applied sciences

MDPI

Article

The Influence of Immersive Virtual Reality Systems on Online Social Application

Zuohao Yan and Zhihan Lv *

School of Data Science and Software Engineering, Qingdao University, Qingdao 266071, China; yanzuohao1994@gmail.com

* Correspondence: lvzhihan@qdu.edu.cn

Received: 25 June 2020; Accepted: 20 July 2020; Published: 23 July 2020



Abstract: This study presents a face-to-face simulation of social interaction based on scene roaming, real-time voice capture, and action capture. This paper aimed to compare the difference between social and traditional plane social communication, analyzing its advantages and shortcomings. In particular, we designed an experiment to study the performance of face-to-face simulation based on virtual reality technology, observing the adaptability of the user to the system and the accuracy of body language recognition. We developed an immersive virtual reality social application (IVRSA), which has a basic social function. As an experimental platform, IVRSA uses Unity3D as its engine, and HTC VIVE as an external input/output device, with voice communication, immersive virtual tour, and head and hand movement simulation functions. We recruited 30 volunteers for the test. The test consisted of two parts. The first part was to provide a news topic for volunteers to freely communicate in IVRSA and WeChat. After communication, we used questionnaires to obtain feedback results of the test and compared the two social applications. In the second part, some volunteers were given a list of actions, which they were asked to describe to the rest of volunteers in the form of body expression, letting them guess the action they were performing. After the end of test, the accuracy rate and the time used were analyzed. Results showed that user's intention expression efficiency was very high in the immersive virtual reality social system. Body tracking devices with more tracking points can provide better body expression effect.

Keywords: immersive virtual reality system; social applications; body expression; body positioning and tracking; 3D scene simulation

1. Introduction

1.1. Subject

As a popular form of social communication in recent years, information exchange of social networking has no limitation in time and space due to the special nature of network service and database technology [1]. However, lack of advanced interaction technology brings some problems in expression of intent. At present, network social communication is limited to the form of text and pictures, which can only simulate forms of information transmission such as letters, bulletin boards, and so on. This leaves users with few options for expressing their intentions [2]. Traditional social software is too simple in its expression of single information in social networking, and the distortion rate is high in the process of channel transmission, which makes the receiver unable to understand the information correctly and accurately.

For example, WeChat, a WIMP GUI-based social software, allows users to use nicknames and avatars as personal identifiers, and can communicate or share information with others via text, pictures, voice, and video, as shown in Figure 1. However, such social applications lack interaction with others

and expression of information via body language. This kind of expression is very different from the way people express themselves in the real world. This interaction design limits the user's intended expression and often requires a long period of training before the user can use the functions provided by these applications.

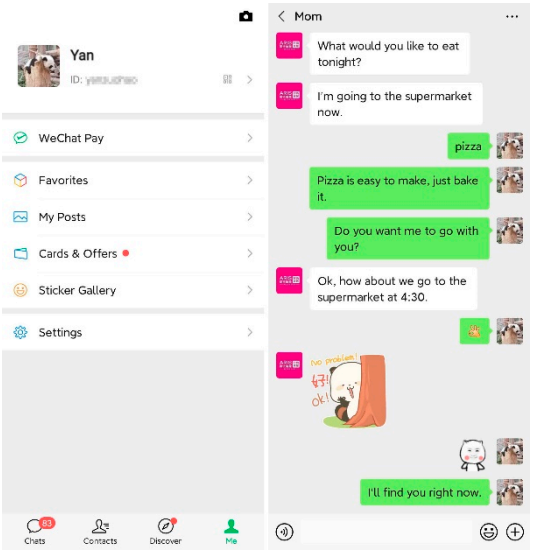


Figure 1. WeChat user ID and text communication (privacy information is blurred).

Is it possible to design new social software that mimics the real world? The appearance of virtual reality technology gives us a feasible scheme.

Virtual reality technology can be considered a kind of advanced UI technology, while the immersive virtual reality system is a more advanced and ideal virtual reality system that tries to fully mobilize the user's perception [3]. It provides a completely immersive experience, giving the operator the feeling of being in a real situation. Head-mounted display (HMD), data glove, and other devices are used to enclose the operator's vision, hearing, and other senses in a designed virtual space [4,5]. The technical advantage of the virtual reality system lies in its use of changes of objects or environments in 3D scenes to output content to users. Traditional desktop UI can only map changes of data generated by information or interaction to two-dimensional representation through metaphor, and what is presented in front of users is changes of pictures or words [6]. Virtual reality technology can directly simulate various responses in real scenes. Using virtual reality in social applications can extend some new modes of operation [7].

There is no doubt that virtual reality is a powerful, high-dimensional, option-rich UI development technology. However, to solve the interaction defects of traditional social applications, the most important thing is to propose a better interaction model.

Development of user interface of human–computer interaction has gone through three periods. The first period, from the early 1950s to the early 1960s, was a user-free phase with punched-card input and line-printer output. The second period, from 1960 to the early 1980s, involved the use of mechanical or teletype typewriters for command entry, which continued into the age of personal microcomputers with command line shells (such as DOS and Unix). The third period, from the 1970s to the present, uses WIMP GUI. WIMP GUI based on Windows, ICONS, menus, and pointing devices has been dominant for more than four decades, thanks to its superior performance in handling common desktop tasks [6]. Professor Andries van Dam of Brown University proposed the next generation of user interface specifications, post-WIMP, in 1997 [8,9]. Such user interfaces do not rely on menus, Windows, or toolbars for metaphors, but instead use methods such as gestures and speech recognition to determine specifications and parameters of operations.

1.2. Overview

Current virtual social platforms can be divided into desktop virtual social platforms and immersive virtual social platforms according to categories of virtual reality systems used and different presentation modes. Desktop virtual social platform is characterized by screen of computer as a window for the operator to observe the virtual environment, and various external devices are generally used to control various objects in a virtual scene. This technology is relatively mature, and currently there are many products on the market. Typical examples are some online games with social functions, such as *World of Warcraft* and *Final Fantasy 14* [10]. Immersive virtual social platform uses a more advanced and ideal immersive virtual reality system. HMD and other devices are used to close the operator's vision, hearing, and other senses in the virtual reality space, and input devices such as position tracker and data glove are used to make the operator feel fully engaged [11]. Development of this technology in the field of social networking is still in its infancy. Currently, there are only a few available platform software, such as Facebook Spaces, Altspace, VRChat, Rec Room, and SteamVR Home [12].

The interaction system of these VR social applications is quite different from that of traditional applications. Facebook launched a virtual social platform called Facebook Spaces at the OC conference in 2016 [13]. Users need to set their own characters according to an official model, and they can communicate and interact with each other in a virtual scene. In Facebook Spaces, players in each room (a separate virtual scene that allows a certain number of players to enter) always surround a virtual round table. Players can freely switch between different round tables or backgrounds. They can also draw pictures, play dice, or take selfies with their friends.

Another typical app is VRChat, which was first released in 2014. VRChat aims at communication and interaction of players, providing a variety of virtual scene activities such as a bonfire party, video and audio entertainment, ball games, and so on. VRChat allows players to create their own personalized characters and spaces. Unlike Facebook Spaces, VRChat not only provides some officially designed characters and scenes, but also allows players to upload their own files, use their favorite models as characters, and design their favorite scenes to play with friends [14]. VRChat also uses "rooms" as a basic unit to divide up different virtual scenes, allowing players to choose their favorite room from a room list. VRChat supports both PC users and HMD users [14]. Users of HMD devices with full-body tracking can perform free body actions and interact with objects in the room at will. According to statistics, the number of VRChat users worldwide has reached 1.5 million, with 20,000 daily active users.

Virtual reality social applications are in their infancy, and there is no unified system framework, interaction design, or data allocation standards. Most of the available virtual reality social applications still rely on text and pictures as the main information carriers, providing users with the features of a virtual reality system as an auxiliary means of communication. Strictly speaking, these apps do not provide full virtual reality social services, but instead provide a hybrid, transitional social service.

2. Related Work

2.1. Research Object

We hope to make a preliminary assessment of immersive virtual reality social applications (IVRSA), analyze how this interaction pattern will affect the online social arena, and compare it with traditional desktop interactions. There are several objective problems in our research. First, the available IVRSA on the market is not entirely dependent on the interaction mode of the post-WIMP user interface and cannot support our research content. Second, there is no standard development and design solution for IVRSA. Therefore, we decided to put forward a development scheme of IVRSA and evaluate the interactive mode of the application.

Before IVRSA development, we must first determine what functions the system should have. In seven IVRSA functions that can be realized (scene roaming and real-time voice, motion capture, scene interaction, user interaction, model transformation, scene/terrain transformation), we chose the most representative of social face-to-face—roaming, real-time voice, and motion capture. It should be

noted that since HTC VIVE is used as the input/output device of the system in this study, the motion capture implemented here has certain limitations and can only capture motion changes on the basis of the three points of head and hand.

At this point, we determined the research objectives of this study—to propose a face-to-face simulation interaction mode based on scene roaming [15], real-time voice, and motion capture, and to design and develop an IVRSA based on this mode. For descriptive purposes, we refer to the system as RVM (roaming, voice communication, motion capture)-IVRSA. RVM-IVRSA was used as the experimental platform to design experiments, and the advantages and disadvantages of this interactive mode were evaluated on the basis of users' actual experience feedback and users' data feedback on the use of body language in virtual scenes.

We designed an experiment wherein 30 volunteers were invited to participate. They were divided into 15 groups, with two people in each group being asked to simulate some social behaviors, and the effects were evaluated. After the experiment, we also prepared a Likert scale questionnaire, through which we analyzed and evaluated this new social mode from the aspects of functional effect and user satisfaction. In addition, we also tested and counted the rate of body language recognition on the experimental platform.

2.2. Technical Support

Unity3D is a cross-platform game development tool that can be developed in a variety of programming languages, such as JAVASCRIPT, C#, PYTHON, etc.; is compatible with Windows, IOS, and other operating systems; and can build PC, mobile, Web, VR, and other applications. Unity3D is a fully functional, highly integrated professional game engine [16,17]. Unity3D uses multiple lenses to observe virtual scenes to achieve 3D visual programming of games, which is convenient for developers to observe and check development content in real time [18].

The peripheral used in this study is HTC VIVE, a VR head-mounted display device jointly developed by HTC and Valve [19]. Valve offers a Steam VR Plugin for developing virtual reality applications on the basis of Steam VR and HTC VIVE. Steam VR Plugin, which is available in Unity3D, provides a script file template to dock with HTC VIVE, which makes it easy to develop sound input of test platform and functions of controller [20].

2.3. Why Improve Interaction

Functions of social software should not affect independent thinking of users in normal social behaviors [21,22]. Users can freely choose which kind of software to use and how to use it. In essence, role of software is to provide a mechanism for users to achieve their goals. Requirements of users are diverse, and thus a comprehensive software should have a certain degree of variability, that is, to provide users with a variety of available solutions. High variability is a kind of respect and consideration that software expresses to user's personal intention. There are many ways to improve variability of software, but interaction mode of desktop social software limits upside potential of variability of software. Immersive virtual reality combines stereoscopic vision and post-WIMP user interaction to create an online social space where users can communicate face-to-face. It does not require users to learn how to use software, nor does it require software to guess users' intention, and thus the potential to improve the function of software is much better than with traditional software.

3. Developing RVM-IVRSA

3.1. Framework

Figure 2 introduces the main functions, network architecture, and underlying data classification of the system from the functional layer, network layer, and data layer of RVM-IVRSA used in the experiment. In addition, we also used some techniques and solutions to optimize the user experience in motion tracking, voice communication, scene synchronization, and visual delay.

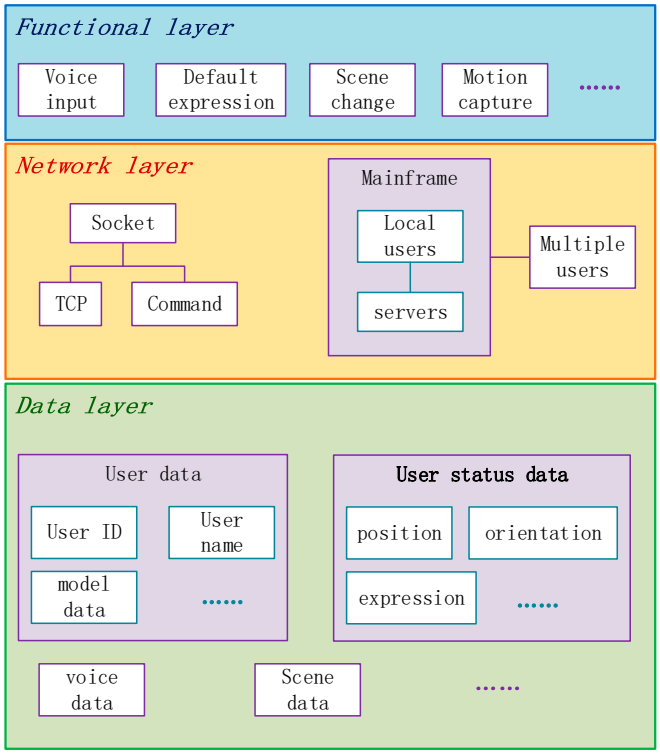


Figure 2. Main functions, network architecture, and underlying data classification of roaming, voice communication, motion capture immersive virtual reality social application (RVM-IVRSA).

3.2. Established Scene

The implementation of RVM-IVRSA functionality is based on a virtual space scenario. We set up a scene in Unity3D and imported a character model and a camera object of SteamVR, as shown in Figure 3. When the scene is running, "Camera" will present the changes of the visual field to the user according to the movement of the HMD. The "Controller (left)" and "Controller (left)" under the "Camera" correspond to simulating the motion of two VIVE controllers. The character model in the scene does not represent the user, but a model used to obtain operational parameters of another terminal to render actions of other users.



Figure 3. A virtual scene built in Unity3D.

3.3. Motion Tracking

A very important function in functional layer is optical positioning of a head display and three points of two controllers, so as to achieve the capture of the user's head and hand movements, which is

the basis of body language expression in a virtual space [23]. The HTC VIVE system includes the following components: VIVE HMD, two Lighthouse laser base stations, and two wireless controllers. The most traditional way to track head position with VR headsets is to use inertial sensors, but inertial sensors can only measure rotation (rotation around the XYZ triaxial axis, called the three degrees of freedom), not movement (along the XYZ triaxial axis, the other three degrees of freedom, collectively called the six degrees of freedom) [24]. In addition, inertia sensor error is relatively large if there is a need for more accurate and free tracking of head movement, or additionally a need for other location-tracking technology. Instead of the usual optical lens and mark point positioning system, the Lighthouse system used by HTC VIVE consists of two laser base stations: an infrared LED array in each base station, and an infrared laser transmitter with two rotating axes perpendicular to each other, with a rotation speed of 10 ms. The base station takes 20 ms as a cycle. At the beginning of a cycle, the infrared LED flashes. The rotating laser on the X-axis within 10 ms sweeps the user's free activity area. In the remaining 10 ms, the rotation laser of the Y-axis sweeps the user's free activity area, and the X-axis does not emit light [25,26]. The Lighthouse base station valve has a number of light-sensitive sensors installed on its HMD, and the controller signal is synchronized after the base station's LED flashes, with light-sensitive sensors being able to measure the time it takes the X-axis laser and Y-axis laser to reach the sensor. This is exactly the time it takes for the X-axis and Y-axis lasers to go to this particular angle, and thus the X-axis and Y-axis angles of the sensor relative to the base station is known. The position of photosensitive sensors distributed on the head display and controller is also known, and thus the position of the head display and motion trajectory can be calculated by the position difference of each sensor. HTC VIVE's positioning system captures and maps spatial position and rotation of HMD and two controllers into virtual space [27]. The schematic diagram is shown in Figure 4.

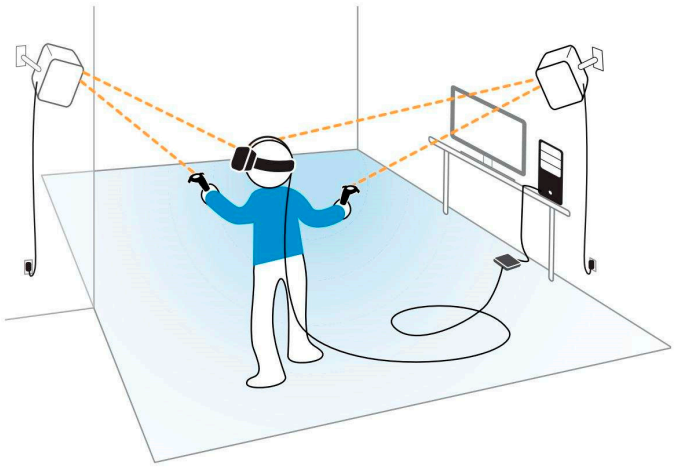


Figure 4. Optical positioning schematic of HTC VIVE.

The screen of user using RVM-IVRSA is shown in Figure 5. What is displayed is the perspective of the user in the system, and the movements of the user's head and hands will be reflected in shape changes of the model's head and hands in another terminal, which is seen by another observer.

In software development, we take a relatively simple path, that is, two terminals send each other user status data obtained by the HTC VIVE's positioning system. The HTC VIVE device was used here without full-body tracking devices, and thus the data captured only included the status of the user's head and hands. We assigned state data of the other user to the head and hand of the model in the scene, so as to complete morphological changes of the head and hand of the user identification (model) in the three-dimensional scene.

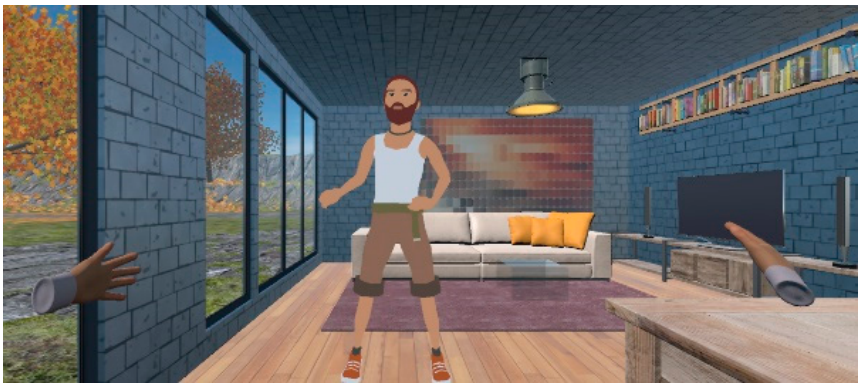


Figure 5. User using RVM-IVRSA.

In the concrete implementation, we added the component "SteamVR_TrackedObject" to the "Controller (left)" and "Controller (left)" of the object "Camera", in which all traceable devices were stored in this "TrackedObject" class. We used the GetComponet method (we needed to set a SteamVR_TrackedObject class variable T using the "T = GetComponent < SteamVR_TrackedObject & gt; ()" statement that realizes the tracking relationship between T and the handle) to obtain the currently traced object and the input to the handle.

The component "SteamVR_Controller" was used to manage input controls for all devices. We set a variable D of class "steamvr_controller.Device" to get the index parameter of T (implemented by the statement "D = steamvr_controller.Input((int)trackedobj.index)"). Using an if statement "if (device.GetPressDown (SteamVR_Controller.ButtonMask.Trigger))" to monitor whether the trigger was pressed, GetPressDown method returned a Boolean value. We designed the system wherein when each trigger was pressed, the "Camera" moved forward 1 m (according to the angle between the "Camera" and the *X*-axis, the system calculates the increment of X in the position parameter when the forward action occurs; the Y increment calculation is the same), thus realizing the user's movement function in the scene.

3.4. Voice Communication

Voice data transfers require use of microphones in HTC VIVE's HMD to collect users' voice data and create a WAVE file to store it. We use DirectX API DirectSound class to complete the task of voice acquisition. The collected voice data are transmitted to other clients through TCP socket. When other clients receive a voice file, they need to write acquired data to buffer and call function, which is "GetVioceData ()" [28–30]. Real-time voice communications allow users to communicate more promptly. At the same time, user's tone, accent, and idiom are also included in the audio signal, which can more comprehensively show the user's mood and personality.

3.5. Scene Synchronization

Different from traditional network data communication, RVM-IVRSA synchronization data are not text, pictures, or files, but are a virtual environment with multiple 3D models. Therefore, scene synchronization should be considered in the design of a test platform.

The test platform adopts ECS (entity component system) architecture [31], whose pattern follows the principle of composition over inheritance. Each basic unit in scene is an entity, and each entity is composed of one or more components. Each component only contains data that represents its characteristics. For example, MoveComponent contains properties such as speed, location, etc. Once an entity owns MoveComponent, it is considered to have the ability to move. System is a tool for dealing with a collection of entities that have one or more of the same components. In this example, moving system is concerned only with moving entities, and it will walk through all entities that have the MoveComponent component and will update the location of entities on the basis of relevant data

(speed, location, orientation, and so on). Entities and components are a one-to-many relationship. What capabilities an entity has depends entirely on what components it has. By dynamically adding or removing components, behavior of entity can be changed at run time [32,33].

On the basis of ECS architecture of the test platform, we adopted a mechanism of deterministic lockstep synchronization to realize scene synchronization [34]. When operation data of a client starts to upload to the server, the server will lock the current frame all the time, and the server will not start to simulate the scene process until all data of the terminal has been uploaded. In the simulation process, the server will process simulation results into the instruction data of the client, which will be forwarded to all user terminals. Finally, user terminals will start their own simulation process on the basis of the forwarding instruction just received. The schematic diagram of the deterministic lockstep synchronization mechanism is shown in Figure 6.

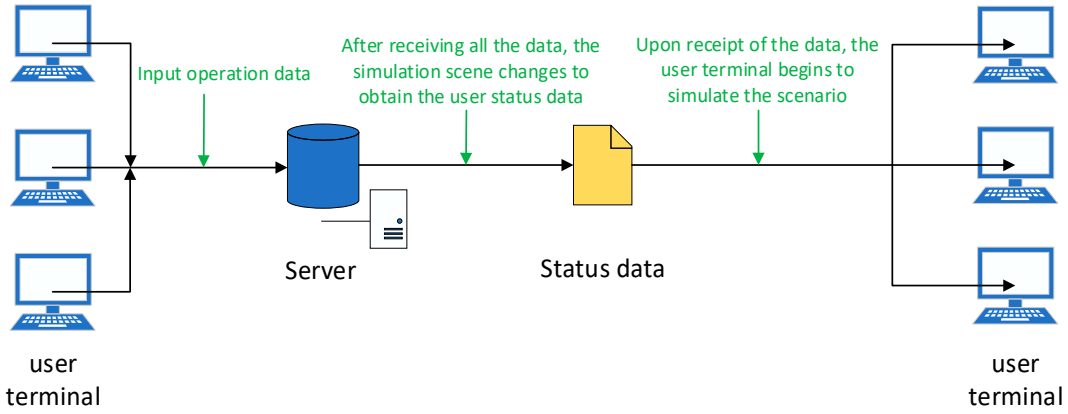


Figure 6. Schematic diagram of deterministic lockstep synchronization.

Here, we used two PCs as terminals along with Alibaba's ECS cloud server. The data transmitted were input data for terminal users, namely, real-time position parameters of HMD, key parameters of handles, and voice files. The values of the operation parameters from the two terminals were first consolidated into two sets of packets, each containing a token value to mark the number of the source terminal (this int variable is called "T number", assigning values 1 and 2 to terminal 1 and 2, respectively). Then, they were packed one at a time and sent to both terminals. The terminal will filter on the basis of the value of "T number", retaining the value of the operation parameter from another terminal to assign the character model B in the scene.

3.6. Compensate Visual Delay

RVM-IVRSA is an immersive virtual reality system with distributed characteristics. When immersive virtual reality system is used, visual delay will occur, which is manifested as HMD conducting angular motion, wherein image generation time in the scene is later than motion time. Humans can sense a visual delay of more than 30 ms, and to maintain a good sense of immersion, a scene needs a frame rate of at least 15 fps and a visual delay of less than 100 ms [35]. There is also some network delay in the RVM-IVRSA in its use of the TCP socket communication system before communication to establish a connection relationship [36,37]. In order to improve delay, some simulation algorithms are needed to estimate motion of HMD in advance.

Dead reckoning (DR) algorithm can compensate for visual delay and network delay [38–40].

$$\hat{P} = P_0 + V_0\tau + \frac{1}{2}A\tau^2 \quad (1)$$

where P_0 is the position vector at time t_0, V_0 is velocity vector at time t_0, A is acceleration vector at time t_0, and $\hat{P}$ is estimated position vector at time $t_0 + \tau$. The calculation error in Formula (1) is

$$\Delta P = P - \hat{P} = \frac{\tau^3}{3!} P^{(3)}(\xi) \qquad \xi \in [t_0, t_0 + \tau] \tag{2}$$

where P is position vector at time $t_0 + \tau$.

All position vectors in Equations (1) and (2) need position parameters (x, y, z) and angle parameters $(\varphi, \theta, \gamma)$ of HMD, that is, $P = (x, y, z, \varphi, \theta, \gamma)^\tau$.

The velocity and acceleration in Equation (1) can be calculated by Equation (3):

$$\begin{aligned} V[t] &= \frac{P[t] - P[t-1]}{\Delta t} \\ A[t] &= \frac{V[t] - V[t-1]}{\Delta t} \end{aligned} \tag{3}$$

In addition to the DR algorithm, we also used the Sklansky model and its prediction algorithm to predict the motion of HMD. The Sklansky model is a kind of basic motion model; the calculation of the algorithm is small, and it is suitable for real-time tracking [41]. Its discrete form is

$$\begin{gathered} X(k+1) = F(k+1|k)X(k) + G\omega(k) \\ F(k+1|k) = \begin{bmatrix} 1 & T & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \; G = \begin{bmatrix} \frac{T^2}{2} \\ T \\ 1 \end{bmatrix}, \; X(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \end{gathered} \tag{4}$$

In Equation (4), T is sampling period. $x_1(k)$ is position of head at time k, $x_2(k)$ is velocity of head at time k, and $x_3(k)$ is acceleration of head at time k. Noise $\omega(k)$ is a gaussian white noise sequence with an average value of 0 ($E\left[\omega(k)\omega^T(j)\right] = Q\delta_{kj}$, δ_{kj} is Kronecker delta).

According to the obtained HMD position information, we can establish the measurement equation of the system as follows:

$$Z(k) = H(k)X(k) + \nu(k) \tag{5}$$

where $H(k) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$, and measuring noise $\nu(k)$ is a gaussian white noise sequence with an average value of 0($E\left[\nu(k)\nu^T(j)\right] = R\delta_{kj}$).

Kalman filter is used in prediction calculation [42], which is a data processing scheme. It can give a new state estimation value from a recursive equation at any time, and thus the amount of calculation and data storage is small [43]. Kalman filtering assumes two variables: position and velocity, which are both random and subject to a gaussian distribution. In this case, position and velocity are related, and the possibility of observing a particular position depends on current velocity. This correlation is represented by $\sum_{ij}$ covariance matrix. In short, each element $\sum_{ij}$ in the matrix represents degree of correlation between ith and jth state variables.

$$\hat{X_k} = \begin{bmatrix} position \\ velocity \end{bmatrix} P_k = \begin{bmatrix} \sum_{pp} & \sum_{pv} \\ \sum_{vp} & \sum_{vv} \end{bmatrix} \tag{6}$$

The equations used for prediction include:

State prediction formula:

$$\hat{X}(k+1|k) = F(k+1|k)\hat{X}(k|k) \tag{7}$$

Variance prediction formula:

$$P(k+1|k) = F(k+1|k)P(k|k)F^T(k+1|k) + G(k)Q(k)G^T(k) \tag{8}$$

Gain formula:

$$K(k) = P(k|k-1)H^T(k)\left[H(k)P(k|k-1)H^T(k) + R(k)\right]^{-1} \tag{9}$$

Filter formula:

$$\hat{X}(k|k) = \hat{X}(k|k-1) + K(k)\left[Z(k) - H(k)\hat{X}(k|k-1)\right] \tag{10}$$

Variance update formula:

$$P(k|k) = [I - K(k)H(k)]P(k|k-1) \tag{11}$$

Prediction residual formula:

$$d(k+1) = Z(k+1) - H(k)\ \hat{X}(k+1|k) \tag{12}$$

Residual assistance formula:

$$S(k+1) = H(k)P(k+1|k)H^T(k) + R(k+1) \tag{13}$$

In the equations above, $(k+1|k)$ means "prediction", $(k|k)$ and $(k|k-1)$ mean "optimal". According to these equations, head position and initial value of predicted variance can be used to predict head motion [44,45].

We added a script component "position adjustment" to the objects in the head and hands of the character model, which corrected the model position on the basis of the DR algorithm and the Sklansky model. It first obtained the position information of the "Camera" and "Controller" at different moments, and then calculated the speed of each moment according to the change of position. According to the change of relative velocity, the system calculated the acceleration at each moment. Finally, the received state parameters were adjusted according to the DR algorithm and Sklansky model, and the position and other parameters of the model were obtained as a result. In this way, the parameter values of the spatial position and direction of the character model were determined, so as to realize the model change synchronization of the two terminals.

4. Experimental Method

4.1. Volunteer

The study involved 30 volunteers from the Chinese city of Qingdao. Of the 30 volunteers, 18 of them were men and 12 were women. We grouped the 30 people by age. Among them, 6 people were over 60 years old (mean age 71.33 years old, standard deviation = 5.28), 6 people were between 45 and 60 years old (mean age 53.5 years old, standard deviation = 3.86 years old), 6 persons were between 30 and 45 years old (mean age 39 years old, standard deviation = 3.32 years old), and 12 people were between 15 to 30 years old (mean age 22.75 years, standard deviation = 3.65).

Before the experiment, the average amount of time in each day spent using social networking software (as measured by volunteers themselves) was calculated for each of the 4 age groups. A total of 12 volunteers aged from 15 to 30 who were all students from Qingdao University used social networking software for an average of 4 h per day (standard deviation = 1.08). Volunteers aged 30 to 45 used social networking software for an average of 3.5 h per day (standard deviation = 1.26). Volunteers aged 45 to 60 used social networking software 2.16 h per day on average (standard deviation = 0.37). All volunteers over 60 were retired and did not use social networking software.

4.2. Experimental Design

This experiment focused on use of real-time voice and body voice in RVM-IVRSA, the two most basic functions for simulating face-to-face communication, and the most advantageous part of RVM-IVRSA compared with traditional social software. The experiment was divided into two parts. In the first part, we divided 30 volunteers into 15 groups (regardless of age or gender, using a free combination) for 10 min of free communication, the users were free to use all functions provided by the RVM-IVRSA. Each set of topics was given by the researchers, all of which were selected from the

top news stories of 2019, such as the 2019 Sino–U.S. trade friction, the 2019 Oscar-winning movie, and so on. The communication activity was carried out twice; the second time, each group had their communication object replaced. During each communication, if one of the volunteers failed to communicate normally, he/she was allowed to interrupt communication at any time, and the duration of activity was recorded. At the end of two communications, the researchers distributed questionnaires to volunteers to measure user satisfaction [46] and some subjective evaluations. In order to compare with traditional desktop social software, we also calculated user comments of desktop social software WeChat according to the same experimental process [47]. Mean and standard deviation of all statistical results and the significant difference between two sets of data are given in the experimental results section.

In the second part, we designed a test for the effect of body language expression in a social scene simulation system [48–50], also dividing 30 volunteers into 15 groups. The test content was that in each group, volunteer A was required to use their head and hands to control the model in RVM-IVRSA and to show action towards another volunteer (volunteer B) in same group. Volunteer B was required to guess what the action was on the basis of the shape change of the model seen in RVM-IVRSA. For 30 s, volunteer B could guess several times until he or she was right. If he or she did not guess correctly after 30 s, the action was considered difficult to understand, and the guessing error was recorded. The test was conducted twice—in the second test, roles of A and B were reversed in each group. Each test required 5 guesses of action content. There are two tests and a total of 10 actions. To obtain a more comprehensive assessment of physical expression in the virtual scene, we asked all "volunteer As" in the first test to demonstrate 5 actions: driving, reading, playing tennis, hugging, and clapping. These were 5 actions that researchers selected after taking into account factors such as difficulty of expression, difficulty of comprehension, and frequency of action. The 5 actions in the second test were freely selected by volunteers, and we hoped to obtain a variety of experimental results in the second test with high randomness.

The 30 volunteers were divided into 15 groups, and we invited volunteers to the laboratory for experiments at 15 different times. Each experiment (consisting of 2 test sections) was used for approximately 2 h. At end of each test, researchers counted the number of correct guesses and time spent using them, and then assessed and analyzed the differences in movement and age.

5. Result

5.1. User Satisfaction

In the questionnaire of the first part of the experiment, we put forward five survey questions about user satisfaction and assessed the satisfaction of volunteers from five aspects. A Likert scale questionnaire was used to score volunteers with a score of 1 to 5. The five questions were

Question 1: Do you think this new mode of interaction is comfortable to use? (high score means comfort)
Question 2: Is the system stable when you use it? (high score means stability)
Question 3: Do you think this kind of social expression is efficient? (high score means high efficiency)
Question 4: Would you recommend this new type of social contact to people around you? (high score means more willing to recommend)
Question 5: Do you think there is much room for improvement in the system? (high score means no improvement is needed)

Through the statistics of the five problems, we could analyze user satisfaction of virtual reality social communication from the five aspects of the system: comfort, stability, use efficiency, extensibility, and improvement demand. We assumed that RVM-IVRSA should be comparable in stability to traditional social software (WeChat), and that it should have an even greater advantage in terms of efficiency. Popularization and improvement on traditional social software should be more mature, with a better result. In terms of comfort, immersive virtual display should be rated lower than

traditional social software because it can cause some physiological problems, such as dizziness. Overall, results should be quite different. Result statistics of the problems are given in Table 1.

Table 1. User satisfaction score.

	RVM-IVRSA Mean/SD	WeChat Mean/SD	*p*-Value
Comfort	3.66/0.59	4.60/0.66	0.02
Stability	4.40/0.32	4.63/0.24	0.39
Efficiency	4.08/0.59	3.77/0.54	0.04
Popularization	3.31/0.65	3.72/0.72	0.04
Improvement	2.45/0.54	3.79/0.31	0.02

In this part of experiment, we proposed the hypothesis that RVM-IVRSA is very different from WeChat in user feedback in terms of the aspects of comfort, efficiency, popularization, and improvement, while RVM-IVRSA's performance in stability was very similar to that of WeChat. The experimental results showed that the user feedback was basically in line with our assumptions. However, combining with the average score data, we found that the average score in efficiency and popularization was relatively close, but the *p*-value was very small, which indicated that users showed obvious personal tendency between IVRSA and WeChat. On the basis of this result, we believe that there are fundamental differences between IVRSA and WeChat as interactive systems.

In terms of stability, a significant difference between two groups of data ($p = 0.39$) indicated that RVM-IVRSA is not different from traditional desktop social applications in terms of stability. The high stability feedback (mean = 4.40, standard deviation = 0.32) indicates that visual delay and network delay of the system are not obvious, and DR algorithm and Sklansky model prediction algorithm have significant effects. RVM-IVRSA's mode of interacting is superior in terms of communication, on the basis of the "usage efficiency" feedback. The statistical results of $p = 0.04$ fully illustrate that this interaction technology is subversive and has a great impact on user experience.

RVM-IVRSA was rated low on comfort. Discomfort caused by using the system was due to vertigo of immersive virtual reality and higher light stimuli. We cannot eliminate the fact that vertigo is always a technical problem of virtual reality that is difficult to conquer. Physical condition and adaptability of VR system experiences determine the degree of vertigo. Although some designs have been adopted in many VR system designs (such as increasing frame rate or canceling physical acceleration) to alleviate vertigo effect to a certain extent, this problem cannot be completely avoided. Generally, continuous wearing time of an immersive VR device should not exceed 30 min.

RVM-IVRSA's "popularization" and "improvement" feedback results were lower than those of WeChat, which revealed shortcomings of immature design of the system. The "popularization" evaluation (mean = 3.31, standard deviation = 0.65) showed that volunteers' evaluation of RVM-IVRSA universality was inconsistent. The main reason was high cost of VR equipment, and some volunteers thought RVM-IVRSA had no advantages in promotion.

5.2. Online Social Experience

In addition to questions about user satisfaction, the questionnaire also included five questions about functional effects and user experience, in order to obtain subjective feedback from volunteers on social functions of the system. The five questions are

Question 6: Is expression of intent limited? (high score means more freedom)
Question 7: Is expression of intent accurate? (high score means more accurate)
Question 8: Are there various ways of expressing intentions? (high score means higher variability)
Question 9: Is social process natural? (high score means more natural)
Question 10: How similar is it to a real face-to-face social situation? (high score means more similar)

In terms of user intent expression, we assumed that RVM-IVRSA should be superior to traditional social software across the board, and differences between the two should be significant, indicating that RVM-IVRSA's social approach is subversive. The results of the above five questions are given in Table 2.

Table 2. Online social experience score.

	RVM-IVRSA Mean/SD	WeChat Mean/SD	*p*-Value
Question 6	4.07/0.35	3.59/0.32	0.01
Question 7	4.00/0.36	3.55/0.26	0.04
Question 8	4.41/0.25	3.75/0.33	0.08
Question 9	4.17/0.36	3.58/0.32	0.02
Question 10	3.69/0.30	2.75/0.47	0.03

In this part of the experiment, we assumed that the feedback results of RVM-IVRSA were better than WeChat in all aspects, and that there was a big gap. The experimental results basically confirmed our hypothesis.

Table 2 shows that the system had obvious functional advantages. Using voice and body language can express users' intention more freely and accurately. Throughout the process, volunteers behaved naturally and communicated efficiently. It was worth mentioning that some volunteers put forward some shortcomings of virtual space scene interaction and modeling quality.

Compared with user feedback from WeChat, RVM-IVRSA features a comprehensive advantage in user's social experience. The significant differences among five sets of data reflected the subversive impact of face-to-face social simulation on the online social field.

5.3. Understanding of Body Language

Table 3 records accuracy rate and mean time data of RVM-IVRSA body language expression recognition by users in our second part of the test. We first recorded recognition accuracy and average time spent (not counting the time used to identify failure) of the five actions provided by researchers (driving, reading, tennis, hugging, and clapping). There were a total of 45 kinds of actions that were independently selected by volunteers, and 75 tests were conducted. It was inconvenient to list too many kinds of test data. The test results of 45 actions in this part were not equal to preset actions in terms of test times and should not be used as comparison data. However, these actions had a strong randomness that was closer to real usage. Therefore, data of this part of the test were integrated and recorded in Table 3 with the label of "Optional". We regard it as reference data that represent real situation. By analyzing it alone, we could better understand the real usage level of body language in this social mode. Combining with preset actions data, we can see the impact of various movements on recognition rate. Figure 7 is drawn from the data in Table 3.

Table 3. Recognition accuracy and mean time of action.

	Accuracy	Mean Time(s)
Drive	1.00	11.6
Read	0.53	18.2
Tennis	0.87	12.75
Hug	0.60	16.5
Handclap	1.00	7.2
Optional	0.55	20.75

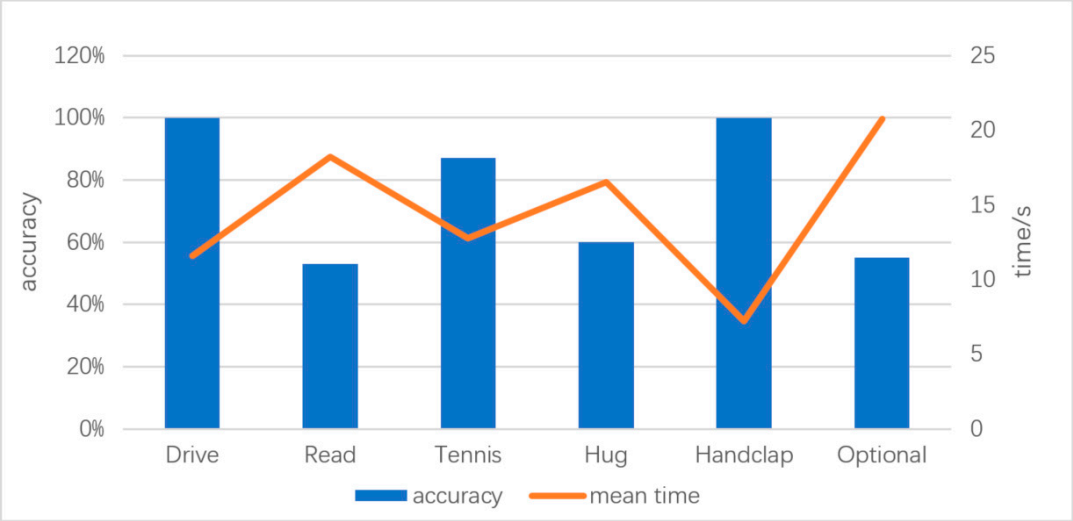


Figure 7. Recognition accuracy and mean time of actions.

Table 3 and Figure 7 show that five pre-prepared actions had a higher recognition accuracy than "Optional" actions of volunteers. The accuracy of "Drive" and "Handclap" reached 100%, "Tennis" reached 87%, and the recognition time of these three movements was within 15 s. The recognition accuracy of "Read" action was only 53%, which we think was because this action not only needed a character model to imitate shape change, but also needed cooperation of other object models (such as books) to be reproduced accurately. When model shape was changed simply, pose estimation could easily be wrong. The "Hug" action was difficult to reproduce accurately using only three points of tracking with the head and hands, requiring a physical change in other parts of body, such as arms. From the data of "Optional", we can infer that in practice, the success rate of body language recognition was not ideal, and the usage time was also a lot. In feedback from volunteers after the experiment, we learned of several reasons behind this. In addition to the points described above, another important reason was that users were not familiar with this type of interaction and could not think of efficient body language movements quickly. Regarding this piece of feedback, we believe that RVM-IVRSA technology development is not mature, and that users need some experience to freely use all its functions; therefore, the use of wizards is very necessary for IVRSA based on motion capture.

We obtained age, gender, and body language recognition rates for 30 users. We attempted to group the data according to age and gender to discuss the effect of users' age and gender on the experimental results in this experiment.

5.4. Influences of User Age

We grouped time data according to age of users. As number of people in each age group was different, we asked the smaller group to carry out several additional tests to ensure that each action of each age group had six test data. The mean time of each group is given in Table 4. Figure 8 is a line graph drawn on the basis of the data in Table 4.

Table 4. Mean action recognition time of each age group.

	15–30	**30–45**	**45–60**	**>60**
Drive	4.16	5.5	7.33	10
Read	16.83	20.5	22.16	24.83
Tennis	10.66	13.3	16.5	20.83
Hug	11.16	16.66	21.5	23.66
Handclap	4.66	5.33	5.33	11
Optional	14.5	19.66	20.66	22.16

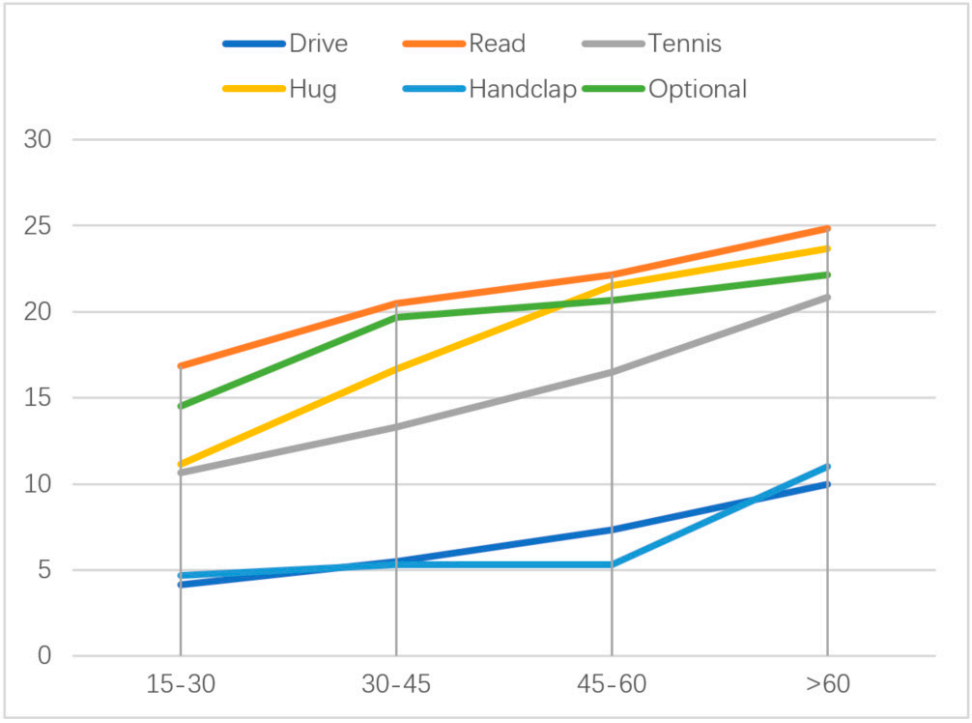


Figure 8. Mean action recognition time of each age group.

Figure 8 shows that the older the user, the more time he or she spends in terms of recognition. Older people use less body language in social interactions and are slower to respond to changes in movement than younger people. In fact, some older people have trouble with body language. Social software developers should consider giving older people more guidance. The time each age group spent in recognizing "Drive" and "Handclap" was the closest, indicating that action with good universality was less affected by user's age.

The time spent on the four actions of "Tennis", "Hug", "Optional", and "Read" increased with the age of users. "Tennis" and "Hug" had the biggest increases in average time, suggesting younger people were using them more than older people.

5.5. Influences of Users' Gender

We also grouped the time data according to the gender of users. We randomly selected 10 groups of male data and 10 groups of female data for comparison. The mean time of the two groups is given in Table 5. Figure 9 is a histogram drawn on the basis of the data in Table 5.

Table 5. Mean action recognition time of males and females.

	Male	**Female**
Drive	6.5	7.7
Read	19.5	21
Tennis	12.5	14.6
Hug	15.3	13.1
Handclap	5	4.8
Optional	17.7	19.6

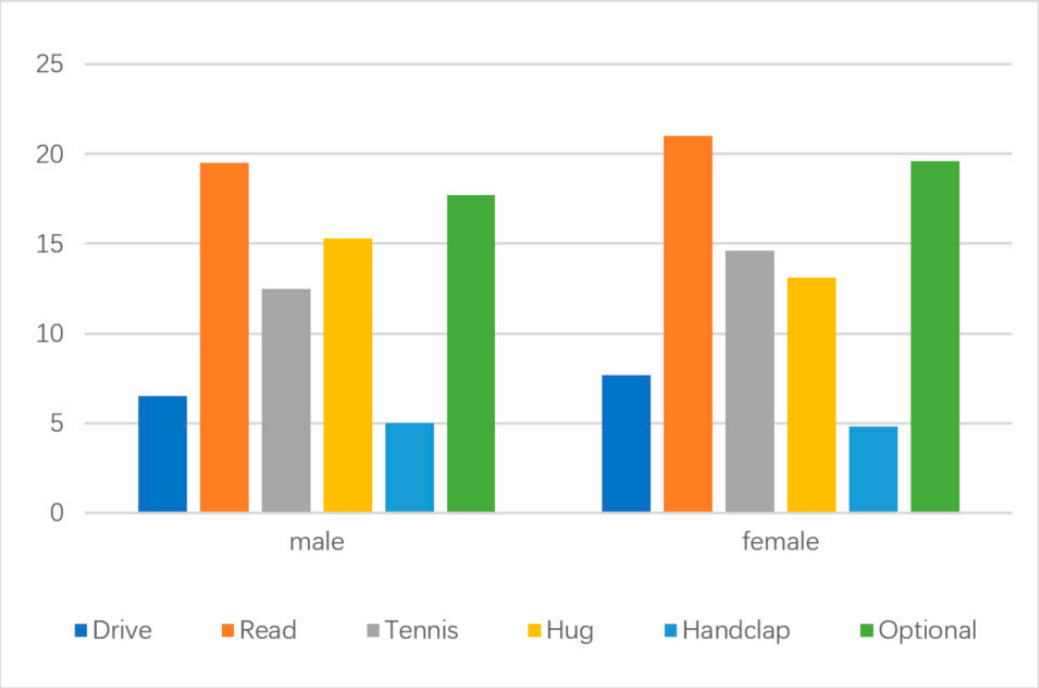


Figure 9. Mean action recognition time of males and females.

From Table 5 and Figure 9, it can be seen that action types have the biggest impact on recognition time. The recognition time of male and female in the same action was not much different. The biggest difference between male and female results was "Tennis", which we think was because males pay more attention to sports than females. This suggests that the influence of gender differences on people's attention is directly reflected in understanding of body language in social behaviors.

6. Conclusions

It should be emphasized that the experimental design and results of this study were based on the functional mode of "scene roaming + real-time voice + motion capture". In the study of face-to-face social simulation, this model was representative, but not comprehensive.

Through the discussion and analysis of experimental results, we have summarized two main conclusions:

Firstly, the method of face-to-face scene simulation using an immersive virtual reality system in social software can effectively improve the efficiency of users' intention expression. The combination of voice communication and body language provides more options for users to express their intentions. Online social applications that simulate face-to-face social situations can create a more natural and realistic social environment for users and improve their social experience. Compared to traditional desktop social applications, volunteers showed a more active and energetic state when using IVRSA.

Secondly, IVRSA does not perform well in body language, but with real-time language communication, it can achieve high recognition accuracy. Social software of VR devices using head and hands positioning is still limited in terms of function of body language expression, and it is difficult to fully reflect shape changes of many complex actions through three-point tracking. High-quality virtual reality social applications require more comprehensive body tracking devices with more tracking points.

In terms of efficiency and ease of use, the post-WIMP interactive system in this mode was obviously better than the traditional desktop interactive system. However, through our experiments, we also found some system defects caused by technical limitations.

First of all, in the user satisfaction survey, we know that users thought that RVM-IVRSA is not convenient and comfortable compared with WeChat application, and that this kind of application

is difficult to popularize at present and needs to be improved. After the experiment, we discussed with the participants and learned that the main cause of these problems was the high requirement of hardware support. Due to the need for real-time graphics computation, the performance of the computer graphics processor directly affects the number of frames of the system. Current immersive VR apps have a small share of the market, and user demand is low. Buying HTC VIVE devices is not a must for most people, and HTC VIVE is not cheap. HTC VIVE's HMD weighs 470 g, which all the participants thought was acceptable, but inevitably uncomfortable.

Secondly, as shown above, the body language recognition experiment shows that the movement reproduction brought by three-point tracking is very rough, which brings great difficulties to the expression of complex movements. In addition, the poor quality of graphic structure and materials has caused some distortion problems. The object collision system in the scene is relatively simple, which also brings about the problem of models penetrating each other. This part of the problem has nothing to do with hardware. It belongs to the shortcoming of the system in function implementation and is the main direction for us to improve it in the future.

We found that when we recorded the rate of body language recognition, we also recorded the user's age and gender. Therefore, in addition to systematic evaluation, we grouped the data according to the age and gender of users, and also discussed the influence of age and gender. We admit that this part of the discussion is not comprehensive, and more rigorous experimental design is needed to further explore this issue.

Users of different ages have different understandings of body language, which was presented in the simulation system. Older users had worse understanding and performance of movements than younger users, and they may have some operational difficulties when using IVRSA. It is necessary for IVRSA to take age into account to improve the manner of operation and guidance. The impact of gender differences on body language understanding is mainly reflected in the different types of hot topics that male and female users pay attention to. For example, in the identification of sports-related body movements, there was a large time gap between male and female users. We believe that in addition to age and gender, cultural, regional, and occupational factors may also lead to differences in users' sensitivity to body movements, and the impact of these factors remains to be studied.

Finally, in a comprehensive evaluation of the system presented in this paper, we believe that the use of RVM-IVRSA is feasible and has many advantages over traditional desktop social applications. However, this system development scheme is not perfect, and there is still a lot of room for improvement. Through the evaluation of RVM-IVRSA, we believe that IVRSA based on other functional patterns is also feasible. In face-to-face social simulation, "RVM" is the most basic functional mode, and other IVRSA functional modes can be regarded as a supplement to "RVM".

Author Contributions: Z.Y. was involved in experimental design, application development, experimental execution, data statistics and paper writing for this study. Z.L. supervised and directed the entire research work and made structural improvements to the final paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China (NSFC) under grant nos. 61902203, Key Research and Development Plan - Major Scientific and Technological Innovation Projects of Shandong Province (2019JZZY020101).

Acknowledgments: First and foremost, I would like to show my deepest gratitude to my supervisor, Zhihan Lv, a respectable, responsible and resourceful scholar, who has provided me with valuable guidance in every stage of the writing of this thesis. Without his enlightening instruction, impressive kindness and patience, I could not have completed my thesis. His keen and vigorous academic observation enlightens me not only in this thesis but also in my future study. Secondly, I would like to thank Qingdao University for its support to my research, which has provided necessary resources for my research. Finally, thank NSDC for its support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Boyd, D.M.; Ellison, N.B. Social Network Sites: Definition, History, and Scholarship. *J. Comput. Med. Commun.* **2007**, *13*, 210–230. [CrossRef]
2. Pempek, T.A.; Yermolayeva, Y.A.; Calvert, S.L. College students' social networking experiences on Facebook. *J. Appl. Dev. Psychol.* **2009**, *30*, 227–238. [CrossRef]
3. Lv, Z.; Yin, T.; Zhang, X.; Song, H.; Chen, G. Virtual reality smart city based on WebVRGIS. *IEEE Internet Things J.* **2016**, *3*, 1015–1024. [CrossRef]
4. Steuer, J. Defining Virtual Reality: Dimensions Determining Telepresence. *J. Commun.* **2000**, *42*, 73–93.
5. Burdea, G.C.; Coiffet, P. *Virtual Reality Technology*, 2nd ed.; John Wiley & Sons: Hoboken, NJ, USA, 2003.
6. Lelli, V.; Blouin, A.; Baudry, B. Classifying and Qualifying GUI Defects. In Proceedings of the 2015 IEEE 8th International Conference on Software Testing, Verification and Validation (ICST), Graz, Austria, 13–17 April 2015; pp. 1–10.
7. Ahlberg, G.; Heikkinen, T.; Iselius, L.; Leijonmarck, C.E.; Rutqvist, J.; Arvidsson, D. Does training in a virtual reality simulator improve surgical performance? *Surg. Endosc. Other Int. Tech.* **2002**, *16*, 126–129. [CrossRef]
8. Van Dam, A. Post-WIMP user interfaces. *Commun. ACM* **1997**, *40*, 63–67. [CrossRef]
9. Van Dam, A. Beyond wimp. *IEEE Comput. Gr. Appl.* **2000**, *20*, 50–51. [CrossRef]
10. Ducheneaut, N.; Moore, R.J. The social side of gaming: A study of interaction patterns in a massively multiplayer online game. In Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work, New York, NY, USA, 6–10 November 2004; pp. 360–369.
11. Biocca, F.; Delaney, B. Immersive virtual reality technology. *Commun. Age Virtual Real.* **1995**, *15*, 32.
12. Lytras, M.D.; Al-Halabi, W.; Zhang, J.X.; Masud, M.; Haraty, R.A. Enabling Technologies and Business Infrastructures for Next Generation Social Media: Big Data, Cloud Computing, Internet of Things and Virtual Reality. *J. UCS* **2015**, *21*, 1379–1384.
13. Neubauer, D.; Paepcke-Hjeltness, V.; Evans, P.; Barnhart, B.; Finseth, T. Experiencing Technology Enabled Empathy Mapping. *Des. J.* **2017**, *20* (Suppl. 1), S4683–S4689.
14. Nugroho, P.W. *Identitas Roleplayer Dalam Game VRChat*; Universitas Airlangga: East Java, Indonesia, 2018.
15. Hui-Juan, Z.H.U. Virtual Roaming System Based on Unity3D. *Comput. Syst. Appl.* **2012**, *10*. [CrossRef]
16. Wang, S.; Mao, Z.; Zeng, C.; Gong, H.; Li, S.; Chen, B. A new method of virtual reality based on Unity3D. In Proceedings of the 2010 18th International Conference on Geoinformatics, Beijing, China, 18–20 June 2010; pp. 1–5.
17. Indraprastha, A.; Shinozaki, M. The investigation on using Unity3D game engine in urban design study. *J. ICT Res. Appl.* **2009**, *3*, 1–18. [CrossRef]
18. Xie, J. Research on key technologies base Unity3D game engine. In Proceedings of the 2012 7th International Conference on Computer SCIENCE & Education (ICCSE), Melbourne, VIC, Australia, 14–17 July 2012; pp. 695–699.
19. Dempsey, P. The teardown: HTC Vive VR headset. *Eng. Technol.* **2016**, *11*, 80–81.
20. Murray, J.W. *Building Virtual Reality with Unity and Steam VR*; CRC Press: Boca Raton, FL, USA, 2017.
21. Von Krogh, G. How does social software change knowledge management? Toward a strategic research agenda. *J. Strateg. Inf. Syst.* **2012**, *21*, 154–164. [CrossRef]
22. Kimmerle, J.; Cress, U.; Held, C.; Moskaliuk, J. Social software and knowledge building: supporting co-evolution of individual and collective knowledge. In Proceedings of the Learning in the Disciplines: International Conference of the Learning Sciences, Chicago, IL, USA, 29 June–2 July 2010.
23. Lubetzky, A.V.; Wang, Z.; Krasovsky, T. Head mounted displays for capturing head kinematics in postural tasks. *J. Biomech.* **2019**, *86*, 175–182. [CrossRef] [PubMed]
24. Barbour, N.; Schmidt, G. Inertial sensor technology trends. *IEEE Sens. J.* **2001**, *1*, 332–339. [CrossRef]
25. Greiff, M.; Robertsson, A.; Berntorp, K. Performance Bounds in Positioning with the VIVE Lighthouse System. In Proceedings of the 2019 22th International Conference on Information Fusion (FUSION), Ottawa, ON, Canada, 2–5 July 2019; pp. 1–8.
26. Bai, H.; Gao, L.; Billinghurst, M. 6DoF input for hololens using vive controller. In *SIGGRAPH Asia 2017 Mobile Graphics & Interactive Applications*; Association for Computing Machinery: New York, NY, USA, 2017.
27. Niehorster, D.C.; Li, L.; Lappe, M. The accuracy and precision of position and orientation tracking in the HTC vive virtual reality system for scientific research. *i-Perception* **2017**, *8*. [CrossRef]

28. Iwami, N.; Matsui, S.; Takahara, K. Voice Communication System and Voice Communication Method. U.S. Patent 5,604,737, 18 February 1997.
29. Schulzrinne, H. *Voice Communication Across the Internet: A Network Voice Terminal*; University of Massachusetts at Amherst, Department of Computer and Information Science: Amherst, MA, USA, 1992.
30. Heredia, R. Voice Communication during a Multi-Player Game. U.S. Patent 6,241,612, 5 June 2001.
31. Wiebusch, D.; Latoschik, M.E. Decoupling the entity-component-system pattern using semantic traits for reusable realtime interactive systems. In Proceedings of the 2015 IEEE 8th Workshop on Software Engineering and Architectures for Realtime Interactive Systems (SEARIS), Arles, France, 24 March 2015; pp. 25–32.
32. Gaspar, C.; Franek, B.; Jacobsson, R.; Jost, B.; Morlini, N.; Neufeld, P.V. An integrated experiment control system, architecture, and benefits: The LHCb approach. *IEEE Trans. Nucl. Sci.* **2004**, *51*, 513–520. [CrossRef]
33. Reilly, C.; Chalmers, K. Game physics analysis and development—a quality-driven approach using the Entity Component Pattern. *Comput. Game J.* **2013**, *2*, 125–153. [CrossRef]
34. Ceze, L.; Godman, P.J.; Oskin, M.H. Enhanced Reliability Using Deterministic Multiprocessing-Based Synchronized Replication. U.S. Patent 8,453,120, 28 May 2013. Available online: https://patents.google.com/patent/US8453120B2/en (accessed on 28 May 2013).
35. Kurita, T.; Lai, S.; Kitawaki, N. Effects of transmission delay in audiovisual communication. *Electron. Commun. Jpn.* **2010**, *77*, 63–74. [CrossRef]
36. Allison, R.S.; Harris, L.R.; Jenkin, M.; Jasiobedzka, U.; Zacher, J.E. Tolerance of temporal delay in virtual environments. In Proceedings of the IEEE Virtual Reality 2001, Yokohama, Japan, 13–17 March 2001; pp. 247–254.
37. Lawton, G. Video streams into the mainstream. *Computer* **2000**, 12–17. [CrossRef]
38. Beauregard, S.; Haas, H. Pedestrian dead reckoning: A basis for personal positioning. In Proceedings of the 3rd Workshop on Positioning, Navigation and Communication, Hannover, Germany, 16 March 2006; pp. 27–35.
39. Randell, C.; Djiallis, C.; Muller, H. Personal position measurement using dead reckoning. In Proceedings of the Seventh IEEE International Symposium on Wearable Computers, White Plains, NY, USA, 21–23 October 2003; pp. 166–173.
40. Steinhoff, U.; Schiele, B. Dead reckoning from the pocket-an experimental study. In Proceedings of the 2010 IEEE International Conference on Pervasive COMPUTING and communications (PerCom), Mannheim, Germany, 29 March–2 April 2010; pp. 162–170.
41. Choi, J. Comparison of Sklansky and Singer Tracking Models Via Kalman Filtering. In Proceedings of the SOUTHEASTCON'83, Orlando, FL, USA, 11–14 April 1983; pp. 335–338.
42. Sun, S.L. Multi-sensor information fusion white noise filter weighted by scalars based on Kalman predictor. *Automatica* **2004**, *40*, 1447–1453. [CrossRef]
43. Delle Monache, L.; Nipen, T.; Liu, Y.; Roux, G.; Stull, R. Kalman filter and analog schemes to postprocess numerical weather predictions. *Month. Weather Rev.* **2011**, *139*, 3554–3570. [CrossRef]
44. Rossi, C.; Abderrahim, M.; Díaz, J.C. Tracking moving optima using Kalman-based predictions. *Evolut. Comput.* **2008**, *16*, 1–30. [CrossRef]
45. Yu, D.; Wei, W.; Zhang, Y. Dynamic Target Tracking with Kalman Filter as Predictor. *Opto-Electron. Eng.* **2009**, *36*, 52–56, 62.
46. Faulkner, X. *Usability Engineering*; Palgrave: London, UK, 2000.
47. Lien, C.H.; Cao, Y. Examining WeChat users' motivations, trust, attitudes, and positive word-of-mouth: Evidence from China. *Comput. Hum. Behave* **2014**, *41*, 104–111. [CrossRef]
48. De Gelder, B.; Hadjikhani, N. Non-conscious recognition of emotional body language. *Neuroreport* **2006**, *17*, 583–586. [CrossRef]
49. Pławiak, P.; Sośnicki, T.; Niedźwiecki, M.; Tabor, Z.; Rzecki, K. Hand body language gesture recognition based on signals from specialized glove and machine learning algorithms. *IEEE Trans. Ind. Inf.* **2016**, *12*, 1104–1113. [CrossRef]
50. Zhao, Y.; Wang, X.; Goubran, M.; Whalen, T.; Petriu, E.M. Human emotion and cognition recognition from body language of the head using soft computing techniques. *J. Ambient Intell. Hum. Comput.* **2013**, *4*, 121–140. [CrossRef]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).