applied sciences

MDPI

Article

Research on Human-Computer Interaction Technology of Large-Scale High-Resolution Display Wall System

Chen Huang [1], Yimin Chen [2,*], Weiqin Tong [1], Tao Feng [3] and Mingxing Deng [3]

[1] School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China
[2] College of Information Technology, Shanghai Jian Qiao University, Shanghai 201306, China
[3] East Sea Information Center, State Oceanic Administration, Shanghai 200136, China
* Correspondence: 21023@gench.edu.cn

Abstract: As an effective solution for data visualization and analysis, the large-scale high-resolution display wall system has been widely used in various scientific research fields. On the basis of investigating existing system cases and research results, this paper introduces the SHU-VAS system (60 screens, 120 megapixels) for data visualization and analysis. In order to improve the efficiency of human-computer interaction in large-scale high-definition display wall systems, we propose an interaction framework based on gesture and double-precision pointing technology. During the interaction process, an adaptive mode switching method based on user action features is used to switch between rough and precise control modes. In the evaluation experiments, we analyzed the characteristics of different interaction methods in movement and navigation interaction tasks, and verified the effectiveness of the gesture-based interaction framework proposed in this paper.

Keywords: display wall system; human-computer interaction framework; double-precision pointing; gesture interaction

check for updates

Citation: Huang, C.; Chen, Y.; Tong, W.; Feng, T.; Deng, M. Research on Human-Computer Interaction Technology of Large-Scale High-Resolution Display Wall System. *Appl. Sci.* **2023**, *13*, 591. https://doi.org/10.3390/app13010591

Academic Editors: Rui Yao and Hancheng Zhu

Received: 1 December 2022
Revised: 29 December 2022
Accepted: 29 December 2022
Published: 1 January 2023

1. Introduction

With the increasing scale and complexity of data, the visual analysis technology of data is receiving more and more attention. A large-scale high-resolution display wall system allows multiple users to observe and analyze a large amount of information at the same time, and can provide enough physical space to realize multi-person collaborative work [1,2]. This type of display wall system has a wide application prospect and can be used for image analysis and processing [3,4], high-resolution image display [5,6], medical research [7], financial transaction data analysis [8], traffic management [9] and other fields. It is an effective tool for visualization and analysis of large and complex datasets and data-intensive tasks [10].

The appearance of the scalable cluster-driven display wall system makes the scale of the display wall system larger and larger. However, it also brings a series of new problems, such as how to achieve the best display effect, how to design appropriate human-computer interaction methods and how to achieve efficient multi-person collaboration [11]. The large-scale virtual reality hybrid system CAVE2 developed by the Electronic Visualization Laboratory of the University of Illinois at Chicago in 2012 is the first display wall system that meets the Hybrid Reality Environment (HRE) standard [12]. The HRE standard has 5 features. The display method: (1). Conforms to the natural visual characteristics of human beings [13]; (2). Supports the stereoscopic display of 2D/3D data sets; (3). Supports natural human-computer interaction; (4). Supports multi-person collaborative workspace; (5). Uses a set of software that can support the effective operation of the system. The HRE standard provides guidance for the design and implementation of similar systems.

In terms of human-computer interaction, traditional display wall systems usually use a mouse-and-keyboard-based interaction method, which limits the operator's interaction range to a certain extent, and the large-scale movement of the mouse reduces the interaction efficiency. In recent years, with the development of sensors and artificial intelligence technology, gesture-based interaction technology has gradually become a hot spot in the field of human-computer interaction research in display wall systems. Although many research works have successfully implemented gesture-based human-computer interaction in display wall systems, work comparing gesture interaction with other interaction methods in different types of interaction tasks is relatively rare. It is therefore difficult to evaluate the superiority of gesture-based interaction methods. To solve these problems, the main work of this paper is:

1. On the basis of fully investigating the characteristics of typical display wall systems in the past, we designed and implemented an interactive high-resolution display wall system SHU-VAS (60 screens, 120 megapixels) for visual analysis of big data.
2. A general human-computer interaction framework for large-scale high-resolution display wall systems is proposed, which uses gesture-based adaptive double-precision interaction technology to achieve efficient human-computer interaction.
3. We designed different types of evaluation experiments to compare the human-computer interaction method proposed in this paper with other interaction methods, and verified the effectiveness of the human-computer interaction framework proposed in this paper.

The organizational structure of the following parts of this paper is as follows: Section 2 introduces the development of the current high- resolution display wall system research and human-computer interaction methods in detail, and summarizes the main field of research on current high-definition display wall systems. Section 3 introduces the system structure of SHU-VAS. Section 4 introduces the human-computer interaction framework proposed in this paper. Section 5 introduces the adaptive double-precision gesture interaction method in the human-computer interaction framework. Section 6 designs and implements two types of evaluation experiments to verify the effectiveness of the human-computer interaction framework in this paper. Section 7 summarizes the work of this paper and presents future work plans.

2. Related Work

2.1. Introduction of Existing Display Wall Systems

Early display wall systems mostly used projection display methods [14], such as the CAVE [15] system. With the development of display equipment, the display screen using LCD panels has become the mainstream display solution for building a display wall system because of its better display effect, higher cost performance ratio and lower maintenance cost. As shown in Table 1, many large-scale high-resolution display wall systems have been used in practical application fields. Although these display wall systems have different shapes and layouts, they are all equipped with powerful computing and rendering systems that can visualize data with extremely high detail and quality. These display wall systems belong to scientific research institutions such as universities, and are mainly used for visualization and analysis of data-intensive scientific research tasks.

The structure of large-scale display wall systems is complex and needs to be composed of various hardware systems and software systems. Therefore, there are many research directions for large-scale display wall systems, including: researching appropriate visualization methods, supporting software for large-scale high-definition display wall systems, and human-computer interaction and collaboration methods. As can be seen from Table 1, most display wall systems are equipped with a tracking system, and the research on human-computer interaction technology for large-scale high-definition display wall systems is a hot spot in this field, and it is also one of the characteristics of display wall systems that are different from traditional display systems.

Table 1. Configuration of large-scale high-resolution display wall systems.

	STALLION [16]	**CAVE2** [10]	**RealityDeck** [17]	**HORNET** [18]	**DO** [19]	**SHU-VAS**
Shape (Top View)						
Layout	Flat 11.2 (L), 2.2 m (H)	Circular 3.65 m (R), 320°	Rectangle 10 m (L), 5.8 m (W)	Curve 7 m (R), 60°	Circular 3 m (R), 313°	Curve 9 m (R), 80°
Screen type (LCD)	30″, 2560 × 1600	46″, 1366 × 736	27″, 2560 × 1440	46″, 1920 × 1080	46″, 1920 × 1080	46″, 1920 × 1080
Number of screens	80 (16 × 5)	72 (18 × 4)	416 (52 × 8)	35 (7 × 5)	64 (16 × 4)	60 (12 × 5)
Number of pixels (megapixels)	328	72	1500	72	132	124
Number of render nodes	23	36	18	3	I: 32 II: 5	I: 24 II: 3 III: 3
CPU	2 × 6-core Intel Sandy Bridge 2667 W	16-core Xeon E5-2690	2 × 6-core Xeon E5645	2 × Intel Xeon E5-2609	I: 6-core Xeon II: 2 × (12/16)-core Xeon	I: 4-core Xeon E5-2623 II: 2 × 12-core Xeon E5-2650 III: 2 × 12-core Xeon E5-2650
GPU	2 × Nvidia Quadro K5000	Nvidia GTX 690	4 × AMD FirePro V9800	3 × Nvidia GTX 780	I: Nvidia Quadro K5000 II: 2 × AMD FirePro W8100	I: Nvidia Quadro M5000 II: 2 × AMD FirePro W9100 III: Nvidia Quadro P5000
Stereo mode	N/A	Passive mode	N/A	N/A	N/A	N/A
Tracking system	N/A	10 × Infrared camera	24 × Infrared camera	7 × Infrared camera	N/A	6 × Depth sensor
Sound system	N/A	20.2	24.4	N/A	16.3	14.2

2.2. Human-Computer Interaction Method

In order to make full use of the advantages of the display wall system, the researchers conducted in-depth research on the key technologies of the display wall system. Human-computer interaction methods constitute one of the important research areas for display wall systems. Although a larger display area is helpful for the visualization of data, it also brings some difficulties to human interaction. With the help of software tools such as Synergy, the cross-machine operation of keyboard and mouse can be realized, but it is difficult to use because the mouse pointer icon is too small and too sensitive. Therefore, the operation efficiency of traditional keyboard and mouse is greatly reduced on large-scale screens [20]. In order to solve this problem, researchers have proposed a variety of solutions, including using additional interactive devices, graphical information and composite interaction methods.

Using additional interactive devices is an effective solution to achieve human-computer interaction, such as using personal smartphones with display wall systems [21,22]. Kim et al. designed a haptic device to assist visually impaired users in target selection during interaction with the display system [23]. However, there is still a lack of unified interfaces and standards for additional interactive devices, making it difficult to achieve cross-system use, which may bring additional learning costs.

Graphical information can provide more auxiliary information for the interactive process at the software layer, and can improve the operation efficiency of users in selecting application windows and targets [24]. For example, the GIAnT system provides a rich set of visual elements to support the study of multi-user interaction behaviors on display wall systems [25]. Similar to the problems of additional interactive devices, there is currently a lack of unified standards for graphical information, so various types of visual information may cause confusion during user interaction.

Combination interaction methods can use multiple interaction methods comprehensively. For example, the TALARIA system combines touch operation with hover operation to help users interact with objects that cannot be directly touched [26]. Combination interaction can make full use of the advantages of various interaction methods, but it also creates higher requirements for the design of combination methods.

Among various human-computer interaction methods, the gesture-based interaction is very suitable for large-scale high-resolution display wall systems. For example, Klein et al. implemented a gesture recognition framework that supports multiple input devices to meet different application requirements and realize the interaction with the display wall system [20]. Mathieu Nance et al. proposed the key technology of designing mid-air interaction [27], and further proposed a solution including an adjustable acceleration function and double-precision technical framework to improve the efficiency of mid-air pointing technology [28]. These interaction methods all need to convert the user's device-based or gesture-based interaction into the movement and operation of the mouse pointer. This research shows that different pointing techniques can effectively accomplish different tasks [29].

Based on these research results, the basic principles for designing human-computer interaction methods in display wall systems can be obtained. First, compared with traditional small-scale displays, when users interact with the display wall system, users tend to use gesture-based interactive operations with larger interactive areas. Then, if the interactive gestures used are familiar to users, it can simplify the process of learning interactive gestures [30]. Finally, the system should provide interaction techniques that do not require a lot of walking, because too much movement will cause physical fatigue and is not suitable for long-term interactive operations [1].

On the other hand, due to the limitations of a single depth sensor in terms of viewing angle and detection distance, in order to realize gesture interaction control on a large-scale display wall system, it is necessary to design a synchronous cooperation and asynchronous alternate control method of multiple depth sensors. To solve this problem, we design and implement a gesture-based human-computer interaction framework called SHU-VAS by referring to the above design principles. The interaction framework uses distributed multiple depth sensors, which can help users realize a wide range of human-computer interaction in the display wall system.

3. System Introduction

The human-computer interaction method of the display wall system is closely related to the structure and shape design of the system. Therefore, before introducing the human-computer interaction framework proposed in this paper, it is necessary to introduce the display wall system used in this paper, including the system structure and layout.

3.1. System Structure

The system structure of SHU-VAS is shown in Figure 1, which is mainly composed of a visual interaction system and a data center. As can be seen from Table 1, the Reality Deck system has reached 1.5 gigapixels. However, the research work of Papadopoulos shows that when the number of pixels in the display wall system exceeds 600 M, the visualization effect of the system will not be further improved, but the search efficiency of visual targets will be significantly reduced [31]. Therefore, SHU-VAS does not pursue a higher number of pixels, but achieves good visualization effects and efficient human-computer interaction under the constraints of limited physical space and funds.

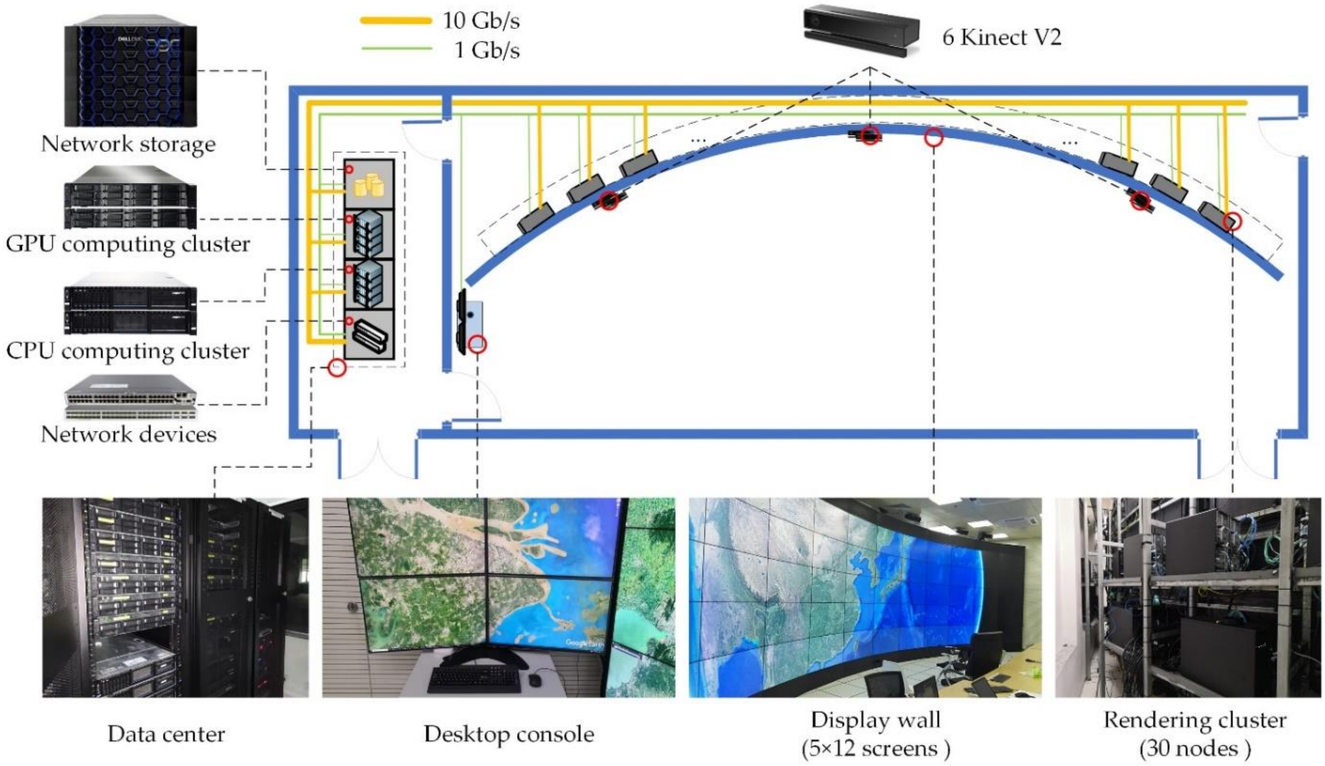


Figure 1. System structure of SHU-VAS.

The visual interaction system mainly includes display wall, rendering cluster and desktop console. The display wall consists of 60 46-inch Full HD (1920 × 1080) splicing screens in a 5 × 12 arc layout, with a total resolution of 23,040 × 5400 and a total of 124 megapixels. The frame thickness between the two screens is less than 3.5 mm, and all screens are connected to the desktop console by daisy chain. The rendering cluster includes 30 high-performance graphics workstations equipped with NVIDIA Quadro series and AMD FirePro series professional graphics cards, which are responsible for driving the entire display array and providing multiple working modes. The desktop console consists of four 4K resolution screens in a 2 × 2 layout. Users can manage the content displayed by the system through the console, and can also control and manage each screen, including switching power supply, switching signal sources, and adjusting display parameters. SHU-VAS includes a gesture interaction system composed of 6 Kinect V2.

With the continuous development of artificial intelligence technology, the scale of various data sets is getting larger, and the complexity of algorithms is also getting higher. In order to better meet the needs of current artificial intelligence algorithms for data set scale and computing power, and to support remote rendering of complex 3D scenes and high-resolution images [32], SHU-VAS includes a data center.

The data center includes CPU computing clusters, GPU computing clusters, network storage and network devices to meet the needs of different application scenarios. Specifically, it includes: 9 CPU computing nodes for high-performance parallel computing; 4 GPU computing nodes equipped with NVIDIA Tesla V100, providing sufficient computing power for AI algorithms; 4 GPU rendering nodes equipped with NVIDIA Quadro P6000, providing the platform with sufficient rendering capacity; 160 TB network storage system, meeting the data storage requirements of various applications. The various components of SHU-VAS are connected by a Gigabit Ethernet network and a 10 Gigabit optical fiber network. The Gigabit Ethernet network is used for the transmission of control signals, and the 10 Gigabit optical fiber network is used for data transmission.

3.2. System Layout

In order to help SHU-VAS have a good display effect, we designed the layout of the screen according to the natural visual characteristics of human beings.

Takashima et al. designed a deformable display wall system, which can change the layout of the screen according to the needs of different application scenarios [33]. Although this design is not yet suitable for large-scale display wall systems, their research work shows that the curved surface structure can meet the needs of most application scenarios. It can also be seen from Table 1 that many display wall systems adopt a curved or circular layout. The flat layout of the STALLION system is not suitable for users to observe all displayed content at medium and close distances. The rectangular layout of Reality Deck makes it difficult to achieve a smooth visual transition at the corners. The circular layout of CAVE2 and DO can provide users with a good sense of immersion, but it also means that users need more turning and walking actions to observe the full picture of the displayed content. In order to enhance the user's observation effect on the overall display content, SHU-VAS adopts a curved layout with a radius of 9 m and an opening angle of 80 degrees.

3.3. System Observation Area

We analyze the observation area of the system according to the two physiological characteristics of human vision: Visual Acuity (VA) and Field of View (FOV).

The visual sensitivity of the human eye limits the Pixel Per Degree (PPD) that the human eye can recognize to 60. That is to say, within the range of 1° viewing angle, the human eye can distinguish up to 60 pixels in one direction. When the PPD is greater than 60, the human eye will not be able to perceive the existence of pixels [34], such as the retina screen used by Apple.

Let p be the distance between two pixels on the display. Let w be the width of the display screen, and h_r be the horizontal resolution of the display screen, then:

$$p = \frac{w}{h_r}, \tag{1}$$

Assuming that the distance between the user and the display screen is d, the PPD can be calculated by the following formula:

$$PPD \approx \frac{d \cdot \tan 1^\circ}{p} = \frac{d \cdot h_r \tan 1^\circ}{w} \geq 60, \tag{2}$$

Then:

$$d \geq \frac{60w}{h_r \cdot \tan 1^\circ}. \tag{3}$$

The parameters of the 46-inch screen used by the system in this paper are w = 1.02 m, h_r = 1920 px. Substituting them into the above formula, $d_{min} \approx 1.8$ m. Therefore, when the distance between the user and the display screen exceeds 1.8 m, the user will not be able to perceive the pixels and achieve the best visual effect. In addition, limited by the size of the room, $d_{max} \approx 4$ m.

When keeping the eyeballs still, the horizontal FOV of human eyes is about 180°, the vertical FOV is about 130°, and the color perception range is about 100° according to the spatial distribution of cone cells [35]. The area with the highest visual accuracy is only 5° from the optical axis. Although the range of visual information that can be recognized with high precision in the visual field is small, the influence of the peripheral visual field on the user's attention should not be ignored. The research work of Sigitov et al. shows that the working area of the display wall system needs to be carefully designed because the peripheral vision may affect the user's attention and reduce work efficiency [36]. Based on the range that the color perception area can cover on the display wall, we analyzed the observation area of the system, as shown in Figure 2.

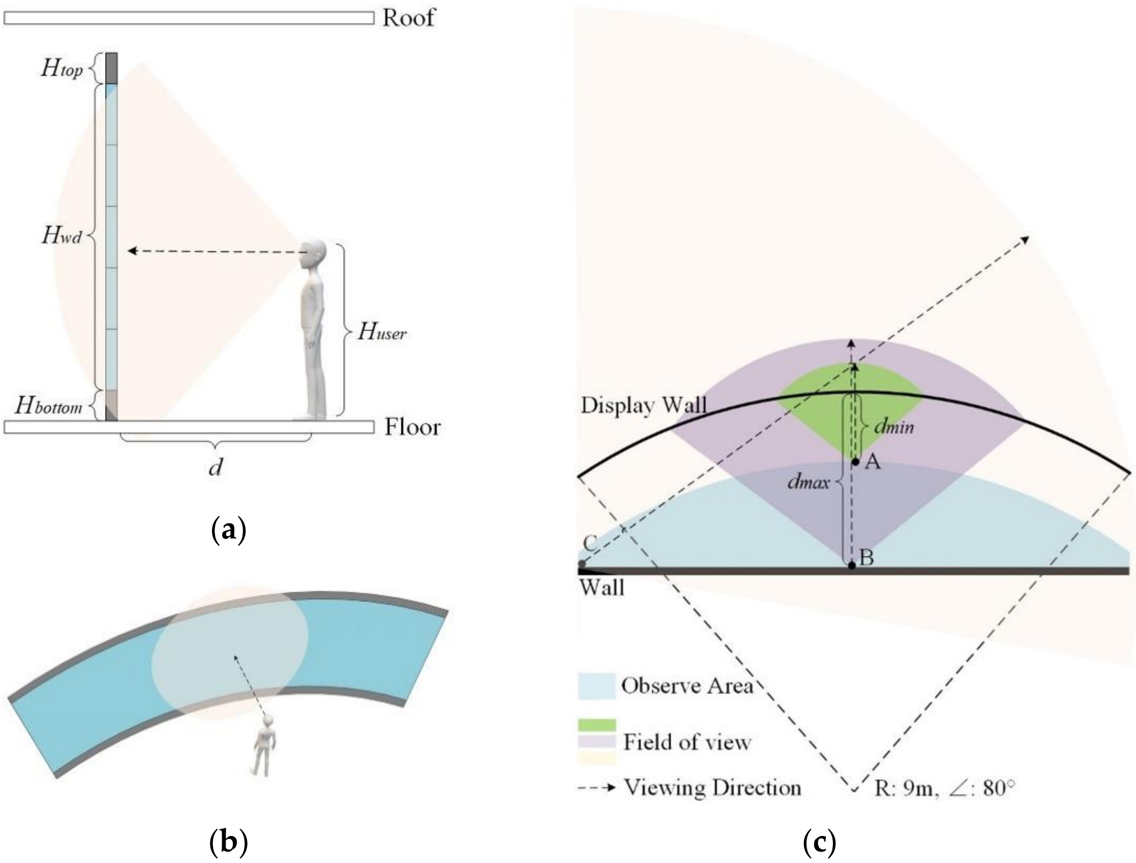


Figure 2. Observation area analysis. (**a**) is a side view of the display wall system; (**b**) indicates that the user's attention is focused on the inside of the display area; (**c**) is the top view of the display wall system.

Figure 2a is a side view of the display wall; the bottom baffle H_{bottom} = 0.3 m, and is used to install 3 Kinect V2 sensors at the bottom. The height of the 46-inch screen is about 0.58 m, so the height of the display area is $H_{wd} = 0.58 \times 5 = 2.9$ m. The height of the top LED strip is H_{top} = 0.3 m. Assuming that the user's height is H_{user} = 1.8 m, the user's horizontal line of sight is in the center area of the display screen, and when $d_{min} \leq d \leq d_{max}$, the display area is all within the color perception range of the human eye in the vertical direction.

As shown in Figure 2b, both the bottom baffle and the top LED light bar use dark color panels, while the roof and the ground use light color panels. By visually setting suggestive boundaries, the user's attention can be focused on the interior of the display area.

Figure 2c is a top view of the display wall. When the user is at point A, the distance from the screen $d = d_{min}$ = 1.8 m, and the user will not be able to distinguish the pixels and achieve good observation effect. However, the user's viewing range is relatively small at this time, and the user needs to turn the head and body in a large range to observe other areas of the display wall. When the user is at point B, the distance from the screen $d = d_{max}$ = 4 m, the user can directly observe most areas of the display wall, and only needs to turn the head in a small range to observe the entire display area. Considering the supplementation of the peripheral vision on both sides, the user will obtain the best observation effect at this time. When the user is at point C, the entire display area can be directly observed, but the observation effect will be affected by the distortion caused by the excessive angle between the line of sight and the display wall.

The effective recognition distance of Microsoft Kinect V2 in the default working mode is 0.8 m to 4 m, and the recommended distance is 1.2 m to 3.5 m. Therefore, when the user is in the system observation area, the Kinect V2 used by the system can better capture the user's skeletal node information. It should be noted that the research work of Hamed et al. [37] shows that when multiple Kinect V2 systems using ToF technology work at the same time, there will be a certain degree of interference, and the median drift reaches 5 mm. In some applications that require high precision, such as high-precision 3D reconstruction, this interference error will have a significant impact on the results. However, in the application based on skeleton node information, the interference error caused by multiple Kinects working together can be ignored. In fact, in order to obtain stable skeletal node information, multiple Kinects with different angles can be used at the same time [38]. In the human-computer interaction framework of the system in this paper, the interference error will also be ignored.

4. Human-Computer Interaction Framework

4.1. Design Principles

Human-computer interaction has always been a research hotspot in the field of large-scale display wall systems. It is meaningful to realize natural and efficient interactive operation with visual objects in a large-scale display wall system. According to the description in Section 2.2 of this paper, we believe that the interaction method for large-scale display wall systems should be simple, direct and reliable. As in the traditional WIMP interaction paradigm, users only need to combine simple operations such as clicking keyboard buttons, clicking mouse buttons and dragging the mouse to realize all interactive operations.

SHU-VAS does not use a display screen that supports touch operation, but we have designed a human-computer interaction framework based on gestures. Compared with previous systems, such as CAVE2, DO, etc., SHU-VAS does not use an infrared camera-based tracking system, but uses 6 Kinects to form a gesture interaction system. The advantage of using Kinect is that users can interact with visual objects using natural gestures without the need for additional markers or equipment. The collaborative use of multiple devices can expand the working area and overcome the occlusion problem to a certain extent.

4.2. Workflow

As shown in Figure 3, in our gesture interaction framework, MSCube SDK is used to collect data from 6 Kinects, and ZeroMQ is used to transmit data to the management server through the network. MSCube is an open source project for processing multi-kinect data, supporting other frameworks such as Cinder, NodeJS, etc. ZeroMQ is a message queue-based multithreaded networking library that uses an asynchronous IO model and provides sockets across multiple transport protocols. Considering the network bandwidth and delay, the system in this paper uses the skeleton node information as the data source of gesture interaction.

After capturing the user's skeletal node information, we use the gesture recognition algorithm we have proposed to recognize the user's interactive gestures [39], then convert the gestures into the operational semantics that the user wants to express, and finally map them to the corresponding system events and call the function interfaces. In order to respond to interactive gestures, applications need to provide customized application interfaces. In order to make the gesture interaction framework of SHU-VAS compatible with more existing applications, we map the user's interaction gesture semantics to the system messages in the traditional WIMP human-computer interaction paradigm. The interaction between the user and the application is realized by converting the user's operational semantics into a series of system messages.

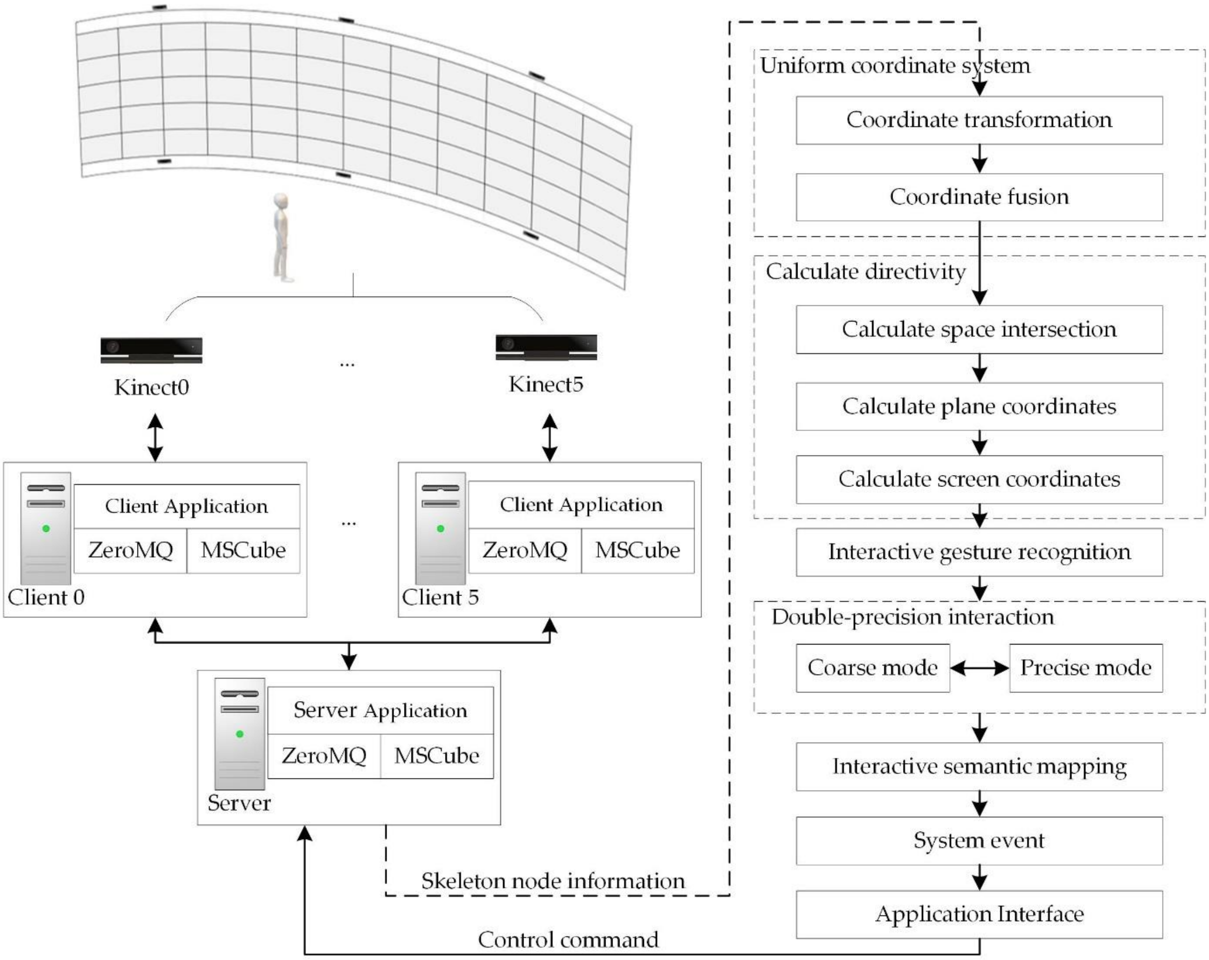


Figure 3. Gesture-based human-computer interaction framework.

4.3. Uniform Coordinate System

In order to facilitate subsequent data processing, we need to make uniform the 3D coordinate information of the skeleton nodes captured by the 6 Kinect V2 into one coordinate system, as shown in Figure 4. To avoid noise interference, all captured skeletal node information is preprocessed using median filtering.

C_i represents the coordinate system centered on the ith Kinect. We take multiple sets of checkerboard calibration board images with different poses in the overlapping area of the two Kinects, and then use stereo calibration method to calculate the transformation matrix $H_{i,j}$ between C_i and C_j.

$$\begin{cases} P_l = R_l \cdot P_w + T_l \\ P_r = R_r \cdot P_w + T_r \end{cases}, \tag{4}$$

where P_w represents the coordinates of a point P on the checkerboard calibration board in the world coordinate system. R_l and T_l represent the rotation matrix and translation matrix of point P relative to the left Kinect position, respectively. R_r and T_r represent the rotation matrix and translation matrix of point P relative to the right Kinect position, respectively. P_l and P_r, respectively, represent the coordinates of the left and right Kinects in the world coordinate system. With P_w, Formula (4) can be rewritten as:

$$(R_l)^{-1} \cdot (P_l - T_l) - (R_r)^{-1} \cdot (P_r - T_r) = 0. \tag{5}$$

Solving Formula (5), we can get:

$$P_r = \left[R_r(R_l)^{-1}\right]P_l + \left[T_r - R_r(R_l)^{-1}T_l\right]. \quad (6)$$

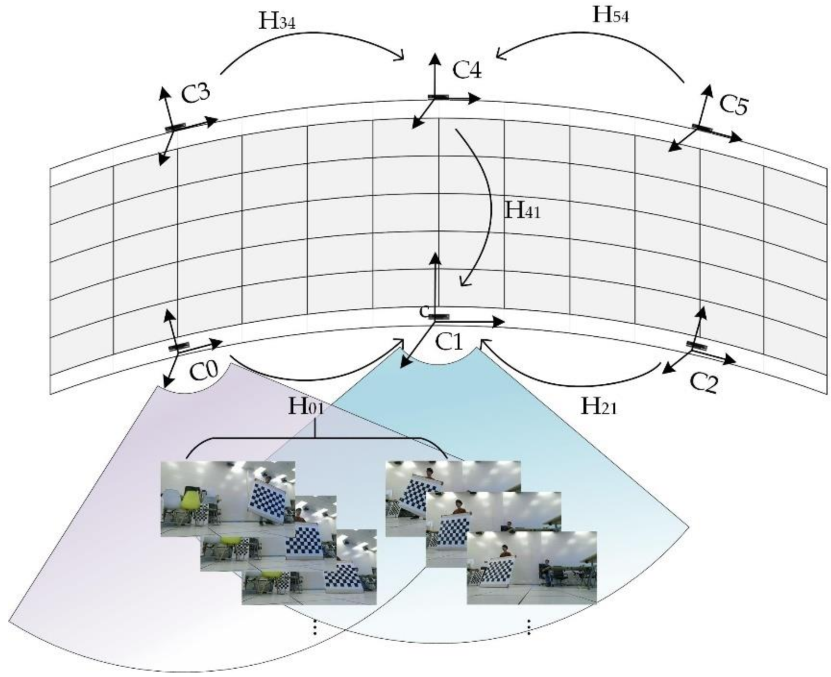


Figure 4. Uniform coordinate system.

The position relationship of the right Kinect relative to the left Kinect can be expressed as:

$$P_r = R \cdot P_l + T. \quad (7)$$

By combining Formulas (6) and (7), we can get:

$$\begin{cases} R = R_r R_l^T \\ T = T_r - RT_l \end{cases}. \quad (8)$$

The transformation matrix H can be expressed as:

$$H = \begin{vmatrix} R & T \\ 0 & 1 \end{vmatrix}. \quad (9)$$

The point $P_{i,k}$ in the coordinate system C_i can obtain the coordinate $P_{j,k}$ of this point in the coordinate system C_j by transforming the matrix $H_{i,j}$, that is $P_{j,k} = H_{i,j}\ P_{i,k}$. We take C_1 as the coordinate system of the interaction area of the whole system. By obtaining the transformation matrix between every two Kinects, we can unify the points in other coordinate systems into the C_1 coordinate system, as follows:

$$\begin{cases} P_{1,0} = H_{0,1}P_{0,0} \\ P_{1,2} = H_{2,1}P_{2,2} \\ P_{1,3} = H_{3,4}H_{4,1}P_{3,3} \\ P_{1,4} = H_{4,1}P_{4,4} \\ P_{1,5} = H_{5,4}H_{4,1}P_{5,5} \end{cases}. \quad (10)$$

The user's skeletal nodes may be captured by n Kinects at the same time, where $n \geq 1$. Then, after the coordinates of the skeletal joint point in each Kinect coordinate system are unified to the C_1 coordinate system by Formula (10), the coordinate of the skeletal node P can be expressed as:

$$P = \frac{1}{n}\sum_{i=1}^{n} P_i, i = 1, 2 \ldots n. \tag{11}$$

The advantage of this processing is that, on the one hand, the stability of the coordinate P can be improved through multi-coordinate system fusion. On the other hand, when the user is performing interactive operations, if some skeletal nodes are blocked relative to a certain Kinect, the spatial coordinates of the node can be obtained with the help of other Kinects, so the robustness of the interaction can be improved to a certain extent.

4.4. Calculating Directivity

After obtaining the stable spatial coordinates of the skeletal nodes, we take the intersection of the spatial straight line determined by the user's right hand elbow coordinates P_{er} and wrist coordinates P_{wr} and the corresponding spatial surface of the display wall as the user's pointing target, as shown in Figure 5. In the WIMP interactive paradigm, this intersection represents the position of the mouse on the screen. The coordinate value of the space intersection point can be calculated by the following formula:

$$\begin{cases} \frac{x - p_{er}(x)}{p_{wr}(x) - p_{er}(x)} = \frac{y - p_{er}(y)}{p_{wr}(y) - p_{er}(y)} = \frac{z - p_{er}(z)}{p_{wr}(z) - p_{er}(z)} \\ x^2 + (z - R)^2 = R^2 \\ 0 \leq y \leq 2.9 \end{cases}, \tag{12}$$

where R represents the radius of the arc surface of the display wall, and R = 9 m.

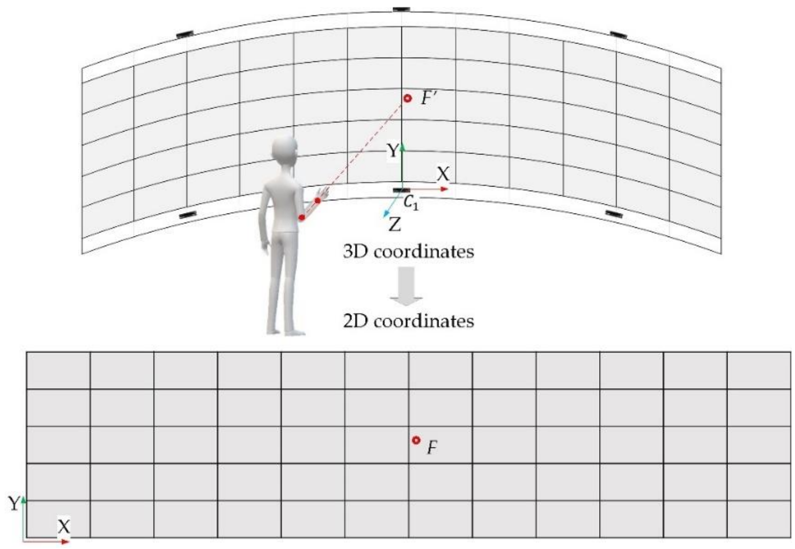


Figure 5. Calculate directivity.

Then, the space coordinates are mapped to plane coordinates for subsequent conversion into screen coordinates:

$$\begin{cases} F(x) = [\sin^{-1}(\frac{F'(x)}{R})]\frac{W}{2} + \frac{W}{2} \\ F(y) = F'(y) \end{cases}, \tag{13}$$

where W is the length of the display wall, and W = 12.57 m.

5. Adaptive Double-Precision Gesture Interaction

5.1. Double-Precision Pointing Technique

Position-based pointing techniques require a transfer function to map input motion captured by the device to cursor motion. The most common transfer function uses Control-to-Display (CD) gain, defined as the ratio between cursor movements, usually measured in terms of linear distance and input change.

The research work of Nancel et al. has theoretically proved that the single-mode remote pointing technology is not suitable for large-scale display wall systems, so a double-precision technical framework supporting the acceleration function was proposed. The double-precision pointing technology divides the pointing task into two stages: (1) the coarse positioning stage, where the user controls the interactive object to quickly approach the target; (2) the precision positioning stage, where the user can precisely control the interactive object to move to the target area.

Both coarse and precise modes can be absolute or relative. Absolute controls are variable speed and easy to use, as the previous cursor position does not affect the input movement, i.e., the cursor can be teleported. They have a fixed gain, however, and thus have limited precision. The pointer acceleration transfer function and wider CD gain range can be applied to the relevant controls, but the offset between the input and cursor position may increase, resulting in variable speed action.

Currently, most research work on double-precision pointing technology uses absolute control for coarse positioning, relative control for precise positioning, and an explicit switching method to switch between coarse and precise modes. We believe that using an explicit switching method will affect the fluency of the user interaction process. In addition, it is necessary to define new interactive gestures or use other devices to realize mode switching, which will increase the user's burden. Therefore, we have designed an adaptive mode switching method based on the moving speed of user interaction gestures.

5.2. Adaptive Mode Switching

By observing the user's interaction process, we found that the user's cursor moves faster in the coarse positioning stage in order to quickly approach the target area. When the interactive object approaches the target area, the user will naturally slow down the moving speed for fine-tuning the position, as shown in Figure 6. According to this operating habit, we switch between coarse and precise modes according to the change of cursor movement speed. After entering precise mode, there is a proportional factor f ($0 < f \leq 1$) between the movement gain of the interactive object and the expected movement gain of the user-controlled object. If user stay in the precise mode for a long time, there will be a significant difference between the position of the interactive object on the display wall and the position pointed by the user, which will bring a sense of strangeness to the user. Therefore, it is necessary to switch the control mode of the position of the interactive object from relative control mode to absolute control mode before making a significant difference.

The moving speed at time t is v_t; then, the average moving speed v within Δt time can be expressed as:

$$\overline{v} = \frac{1}{\Delta t}\int_{t-1}^{t} v_t dt. \tag{14}$$

If the current moving speed v is lower than a certain percentage of the average speed $\overline{v}$ in the previous Δt range, it will automatically enter the precise control mode. In this mode, the movement gain of the interactive object is determined by multiplying the absolute gain of the user's pointing by the scaling factor f_v.

When the user controls the movement distance d of the interactive object in the precise movement mode to be greater than the threshold l, in order to avoid a significant difference between the position of the interactive object and the pointing position of the user, the coarse control mode is restored and the position of the interactive object is corrected to the current position pointed by the user. The switching process between each state is shown in Figure 7.

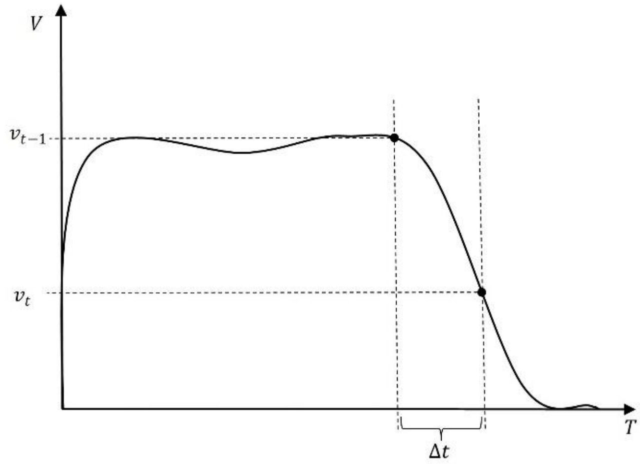


Figure 6. The movement speed of the interactive object.

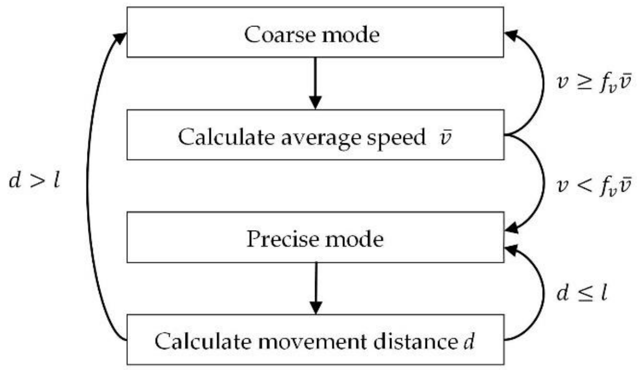


Figure 7. Mode switch state machine.

6. Evaluation Experiments

In order to evaluate the effectiveness of the human-computer interaction framework proposed in this paper, we designed two types of evaluation tasks: movement and navigation. The effectiveness of the interaction method proposed in this paper is evaluated by comparing the time to complete the task, operation accuracy and other metrics with different interaction methods.

6.1. Experiment 1 Movement Task

6.1.1. Experiment Setup

We configure the Sage2 service deployed on the server, and configure the 60 screens of the display wall system into a complete 5×12 unified desktop. The system uses 27 rendering nodes to drive the 60 screens. Among them, there are 23 type I rendering nodes, each driving 2 screens, and 3 type III rendering nodes, each driving 4 screens. Users can interact in the following 3 ways:

1. Desktop console. The user interacts with the Sage2 console using the keyboard and mouse on the desktop console.
2. Tablet PC with touch screen. Users use a Microsoft Surface Pro 4 tablet computer to log in to the Sage2 server and use the touch screen for interactive operations.
3. Gesture interaction. Users use the human-computer interaction framework proposed in this paper to perform interactive operations through gestures.

6.1.2. Operational Flow

The operation process is shown in Figure 8. Randomly select a grid position in the starting area of 5×6 grids on the left to generate an interactive object with a red background, and randomly select a grid position in the target area of 5×6 on the right to generate a dotted frame as the moving target of the interactive object. The timing starts when the user selects the interactive object, and stops when the user moves the interactive object near the target area and releases it.

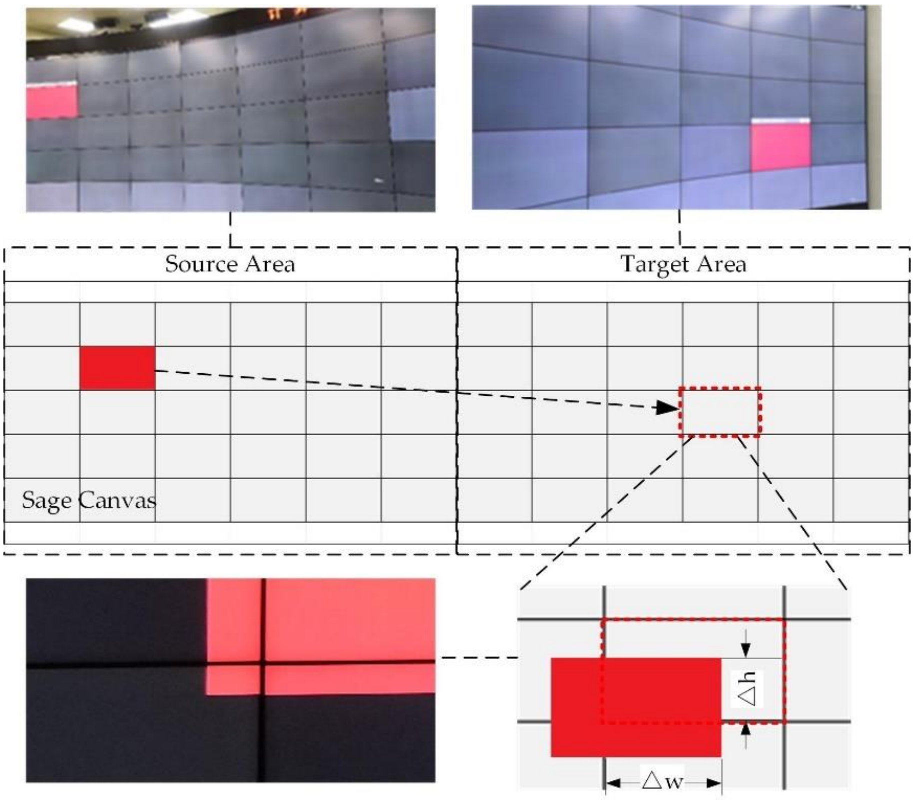


Figure 8. Movement task.

6.1.3. Evaluation Metrics

We define the accuracy rate *acc* of the operation as the ratio of the overlapping part of the moving object and the target area to the area of the target area:

$$acc = \frac{\Delta w \cdot \Delta h}{w \cdot h}, \tag{15}$$

where w and h denote the width and height of the target area, respectively. Δw and Δh represent the width and height of the overlapping part of the interactive object and the target area, respectively.

We believe that the distance moved by the grid and the time used are important factors affecting the score, and the score of the operation is defined as:

$$score = acc \cdot \frac{d}{t}, \tag{16}$$

where d represents the moving distance, and t represents the time used to complete the movement task.

6.1.4. Experiment Results and Analysis

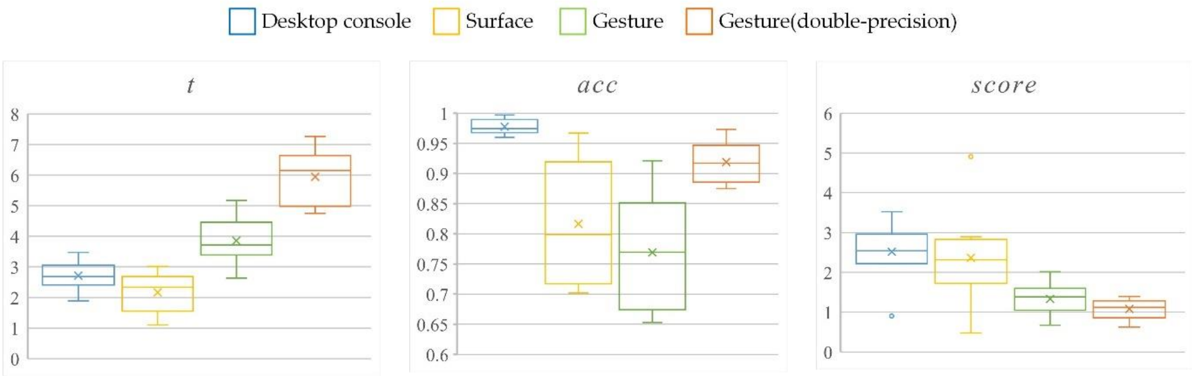
The experimental results on the movement task are shown in Table 2, and the statistical analysis results are shown in Table 3 and Figure 9.

Table 2. Experimental results of movement task.

	Source Area (row,col)	Target Area (row,col)	d	t	Acc	Score
Desktop Console	(1,4)	(8,3)	7.212	2.43	0.997	2.959
	(2,3)	(7,2)	5.201	1.89	0.96	2.642
	(4,1)	(9,4)	5.948	2.61	0.972	2.215
	(3,3)	(9,2)	6.204	2.73	0.977	2.220
	(4,2)	(11,1)	7.212	2.9	0.982	2.442
	(4,5)	(11,3)	7.426	3.05	0.991	2.413
	(2,3)	(11,2)	9.236	3.06	0.968	2.922
	(5,2)	(8,3)	3.226	3.47	0.971	0.903
	(2,3)	(9,2)	7.212	2.35	0.967	2.968
	(1,2)	(10,4)	9.404	2.64	0.989	3.523
Surface	(5,4)	(10,3)	5.201	1.62	0.702	2.254
	(1,5)	(9,4)	8.224	2.52	0.779	2.542
	(4,5)	(12,5)	8.160	1.36	0.818	4.908
	(5,4)	(11,4)	6.120	2.36	0.918	2.381
	(1,3)	(9,4)	8.224	2.75	0.967	2.892
	(4,2)	(11,5)	7.768	3.02	0.722	1.857
	(6,3)	(7,2)	1.442	2.66	0.874	0.474
	(5,1)	(8,4)	4.327	2.31	0.703	1.317
	(4,1)	(8,1)	4.080	1.1	0.757	2.808
	(5,3)	(9,1)	4.562	1.91	0.924	2.207
Gesture	(1,4)	(11,1)	10.649	5.17	0.659	1.357
	(6,2)	(10,4)	4.562	3.69	0.653	0.807
	(4,2)	(7,2)	3.060	3.54	0.773	0.668
	(4,1)	(7,5)	5.100	4.16	0.921	1.129
	(4,5)	(8,1)	5.770	2.94	0.781	1.533
	(5,1)	(12,4)	7.768	3.74	0.679	1.410
	(4,4)	(11,1)	7.768	4.75	0.873	1.428
	(6,5)	(10,5)	4.080	2.63	0.766	1.188
	(4,4)	(12,1)	8.715	3.61	0.74	1.786
	(2,5)	(12,3)	10.402	4.36	0.844	2.014
Gesture (double-precision)	(3,1)	(11,3)	8.411	7.26	0.961	1.113
	(6,1)	(10,3)	4.562	6.72	0.918	0.623
	(2,3)	(9,3)	7.140	5.92	0.915	1.104
	(1,2)	(7,2)	6.120	4.8	0.886	1.130
	(3,5)	(10,5)	7.140	5.04	0.973	1.378
	(4,2)	(12,4)	8.411	6.61	0.885	1.126
	(3,2)	(12,5)	9.677	6.38	0.916	1.389
	(4,1)	(10,3)	6.451	6.48	0.875	0.871
	(1,2)	(8,4)	7.426	5.46	0.919	1.250
	(5,4)	(9,4)	4.080	4.75	0.942	0.809

Table 3. Statistics table of movement task results.

	t			acc			Score		
	Min	**Max**	**Avg**	**Min**	**Max**	**Avg**	**Min**	**Max**	**Avg**
Desktop Console	1.89	3.47	2.713	0.96	0.997	0.977	0.903	3.523	2.521
Surface	1.1	3.02	2.161	0.702	0.967	0.816	0.474	4.908	2.364
Gesture	2.63	5.17	3.859	0.653	0.921	0.769	0.668	2.014	1.332
Gesture (Double-Precision)	4.75	7.26	5.942	0.875	0.973	0.919	0.623	1.389	1.0793



(**a**) (**b**) (**c**)

Figure 9. Boxplot of movement task results. (**a**) The time consumption. (**b**) Accuracy of interaction operation. (**c**) Score of interaction operations.

In terms of interaction time, the keyboard and mouse-based desktop console is closer to the touch-screen-based Surface, while the gesture-based interaction takes longer. Although the user is more familiar with the operation of the traditional keyboard and mouse, the physical distance for the user's finger to move on the Surface touch screen is shorter, so the time consumption of the two interaction methods is relatively close. Since gesture interaction using double precision technology consumes more time in precise control mode, the total time consumption is significantly higher than other methods. We believe that users can improve interaction efficiency after practicing interactive gestures for a period of time.

In terms of accuracy, the accuracy and stability of the desktop console method is significantly better than other methods. The results of Surface and gesture interaction are relatively close. The accuracy rates of these two interaction methods for multiple rounds of operation are quite different, and the stability is low. We believe this is due to the user's low stability of body control. After combining double-precision pointing technology, the accuracy and stability of gesture interaction have been significantly improved.

The score is a combination of interaction time and accuracy. The desktop console and Surface scored higher due to their advantage in interaction time. Affected by the accuracy rate, the score of gesture interaction is significantly lower than other methods. While gesture interaction combined with double-precision pointing technology improved accuracy, the score decreased slightly due to the overall time-consuming increase. This shows that compared with other interaction methods, gesture interaction in movement task does not have advantages in interaction efficiency.

In addition, we counted the time consumption of 10 groups of double-precision gesture interactions, as shown in Figure 10. It can be seen that the time consumption of the precise control mode accounts for about 32.5% of the total time consumption Due to the introduction of the precise control mode, the average time spent on interaction has increased by 54%, and the average accuracy rate has increased by 19.5%, which is close to the accuracy rate of desktop console and better than Surface touch operations. This shows that gesture interaction using double-precision technology is feasible in application scenarios where there is no limit on the interaction time.

6.2. Experiment 2 Navigation Task

On the SHU-VAS platform, we built the Liquid Galaxy [40] system based on the Master-Slaver structure. The Liquid Galaxy system was originally built to run Google Earth to create an immersive visual experience for users [41]. Using the interaction framework of the SHU-VAS, in addition to the traditional keyboard and mouse, 3D mouse, and Leap Motion interaction methods, we have also realized gesture-based human-computer interaction. Users can directly use gestures to interact with the virtual globe, such as zooming, rotating, etc.

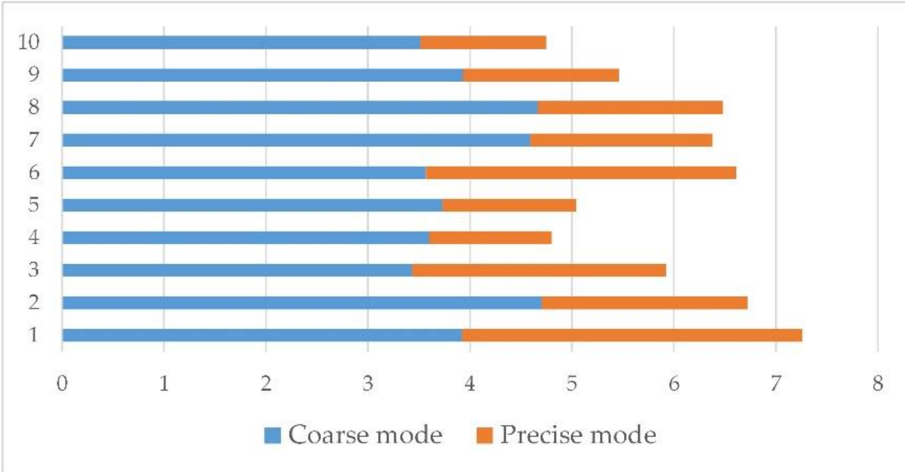


Figure 10. Proportion of each mode of double-precision pointing technology.

6.2.1. Experiment Setup

The structure of the Liquid Galaxy system built based on the system in this paper is shown in Figure 11. The system uses 6 high-performance rendering nodes to drive the 60 screens. Among them, there are 3 type II rendering nodes, each driving 16 screens; 3 type III rendering nodes, each driving 4 screens. Users can use the following four methods for interactive operations:

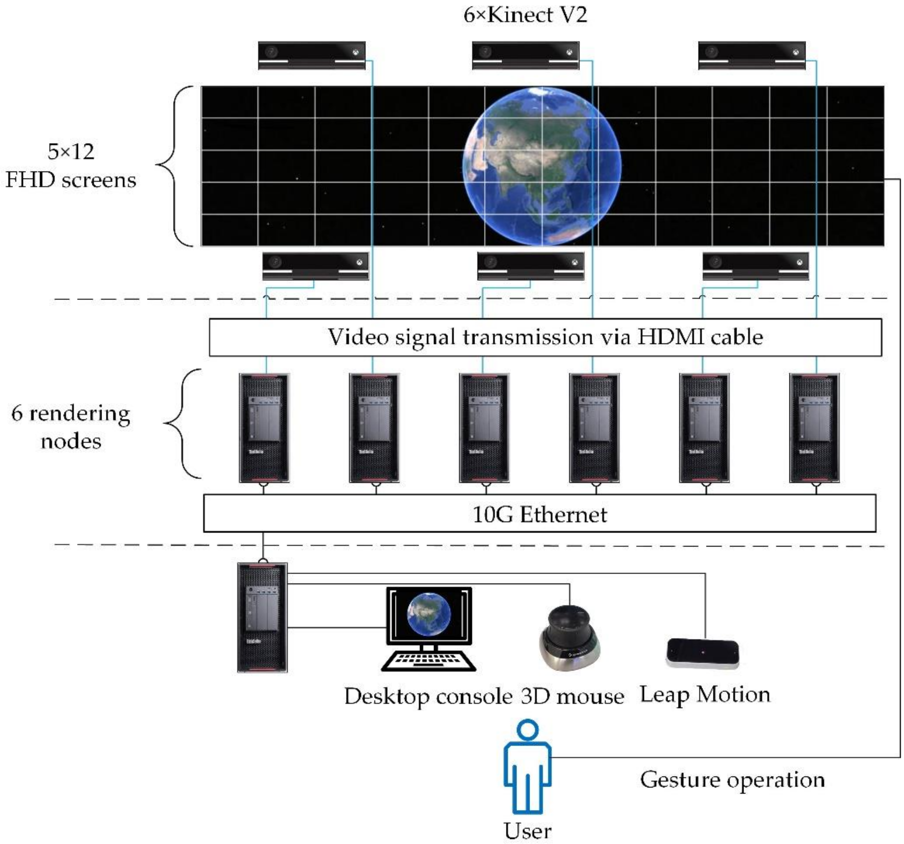
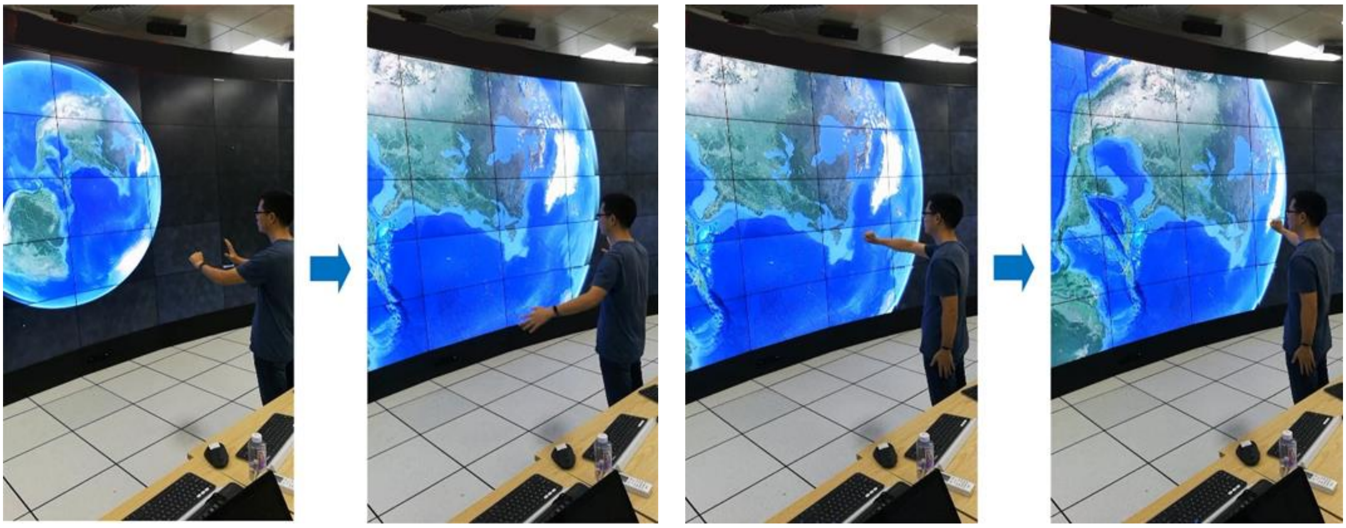


Figure 11. System structure of Liquid Galaxy system.

1. Desktop console. On the console of the display wall system, the user uses the keyboard and mouse to perform interactive operations on the Master side of Google Earth.
2. 3D mouse. On the console of the display wall system, the user uses a three-dimensional mouse to perform interactive operations on the Master side of Google Earth.

3. Leap Motion Controller. The user uses the Leap Motion controller to interact with the Master side of Google Earth on the console of the display wall system.
4. Gesture interaction. Users use the human-computer interaction framework proposed in this paper to perform interactive operations using gestures, as shown in Figure 12.



(a) (b)

Figure 12. Navigation task interactive operation. (**a**) Zoom operation. (**b**) Rotation operation.

6.2.2. Operational Flow

Set the current viewing angle center as the starting position on the Master side of Google Earth: 35°00′00.00″ ± 5″ north latitude, 100°00′00.00″ ± 5″ east longitude, and 10,000 ± 100 km viewing angle altitude. Target position: 31°19′00.00″ ± 5″ north latitude, 121°23′30.00″ ± 5″ east longitude, viewing angle altitude 1000 ± 100 m. When the user navigates the center of view to the target position through various interactive methods, the navigation task is completed and the timing is stopped.

6.2.3. Evaluation Metrics

Navigation operations can generally be divided into two types: moving and zooming. Let N_{move} represent the number of moving operations, and N_{zoom} represent the number of zooming operations. The total number of operations is $N = N_{move} + N_{zoom}$, and the smaller N is, the fewer adjustments need to be made during the navigation process, and the higher the navigation efficiency is.

Define the score of an operation as:

$$score = \frac{C}{N \cdot t}, \tag{17}$$

where C is a constant, and $C = 100$. t represents the time taken to complete the navigation task.

6.2.4. Experiment Results and Analysis

The experimental results of the navigation task are shown in Table 4, and the statistical analysis results are shown in Table 5 and Figure 13. It can be seen that although there is a certain learning cost for using the 3D mouse and Leap Motion for navigation operations, users can still achieve operating efficiency similar to that of traditional mouse operations. We believe that the interactive devices used in these three interaction methods are placed on the console desktop in a relatively stable manner, and the offsets of each user operation are relatively similar, so the average number of operations required to complete the navigation task is similar.

Table 4. Experimental results of navigation tasks.

	Nmove	Nzoom	t	Score
Desktop Console	6	4	7.227	1.384
	6	8	7.815	0.914
	5	5	7.443	1.344
	6	6	7.694	1.083
	6	7	12.084	0.637
	4	7	6.293	1.445
	4	8	6.588	1.265
	4	4	9.961	1.255
	8	4	7.954	1.048
	7	8	12.282	0.543
3D Mouse	4	4	11.446	1.092
	4	6	10.781	0.928
	4	6	11.09	0.902
	6	4	9.838	1.016
	6	4	10.67	0.937
	3	6	9.12	1.218
	6	7	9.313	0.826
	3	7	9.799	1.021
	4	6	11.807	0.847
	6	8	10.813	0.661
Leap Motion	3	5	11.566	1.081
	3	5	7.525	1.661
	6	5	11.403	0.797
	4	6	11.309	0.884
	3	6	12.565	0.884
	6	6	11.696	0.712
	4	5	7.824	1.420
	6	5	8.644	1.052
	6	4	12.878	0.777
	6	5	11.609	0.783
Gesture	7	9	8.476	0.737
	6	10	9.807	0.637
	7	7	10.236	0.698
	7	6	9.493	0.810
	5	7	8.811	0.946
	9	10	9.528	0.552
	8	7	9.67	0.689
	7	7	10.72	0.666
	9	8	9.389	0.627
	8	9	8.804	0.668
Gesture (double-precision)	6	6	10.976	0.759
	8	7	12.738	0.523
	6	10	9.452	0.661
	7	7	12.621	0.566
	9	6	9.29	0.718
	6	8	11.448	0.624
	9	7	13.131	0.476
	5	10	12.841	0.519
	6	9	9.263	0.720
	6	8	10.756	0.664

Gesture operation has a relatively low overall score because the number of interactive operations is significantly more than other methods. The introduction of double-precision gesture operation allows faster reaching of the target in the final fine-tuning stage and reduces the average number of operations. However, due to the increase in total time consumption, the score is reduced. Therefore, for navigation tasks, there is no advantage in gesture operation in terms of interaction efficiency.

Table 5. Statistics table of navigation tasks.

	Nmove + Nzoom			t			Score		
	Min	Max	Avg	Min	Max	Avg	Min	Max	Avg
Desktop Console	8	15	11.7	6.293	12.282	8.534	0.543	1.445	1.092
3D Mouse	8	14	10.4	9.12	11.807	10.468	0.661	1.218	0.945
Leap Motion	8	12	9.9	7.525	12.878	10.702	0.712	1.661	1.005
Gesture	12	19	15.3	8.476	10.72	9.493	0.552	0.946	0.703
Gesture (double-precision)	12	16	14.6	9.29	13.131	11.252	0.476	0.759	0.623

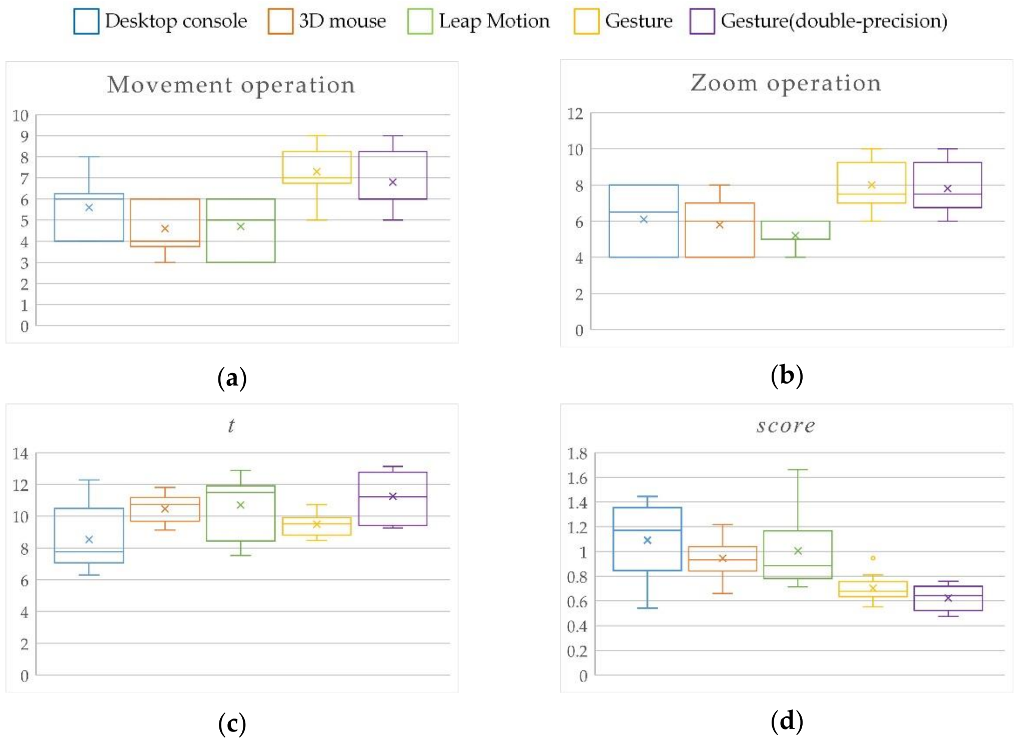


Figure 13. Boxplot of navigation task results. (**a**) Number of movement operations. (**b**) Number of zoom operations. (**c**) The time consumption. (**d**) Score of interaction operations.

7. Conclusions and Outlook

This paper introduces the large-scale high-resolution display wall system SHU-VAS (60 screens, 120 megapixel) for big data visual analysis. This paper designs and implements a human-computer interaction framework for a large-scale high-resolution display wall system. The framework uses gesture-based double-precision interaction technology to achieve efficient human-computer interaction, and uses an adaptive mode switching method to avoid the impact of explicit mode switching on the fluency of the interaction process.

The evaluation experiments of movement task and navigation task show that, compared with other interaction methods, the human-computer interaction framework proposed in this paper is close to the traditional keyboard-mouse interaction method in terms of operation accuracy, and has practical application value.

As large-scale display wall systems are more and more widely used, related software and hardware technologies are also constantly developing. There are still many aspects of the SHU-VAS system that deserve further research and improvement. Since this paper only uses objective evaluation indicators, it lacks the subjective evaluation of users. In the future work plan, on the one hand, we will further improve the evaluation criteria, such as using the SUS (System Usability Scale) evaluation method. On the other hand, we will further expand the human-computer interaction framework to provide more interaction methods, such as using facial tracking to assist gesture interaction and supporting voice interaction.

Author Contributions: Conceptualization, C.H. and W.T.; methodology, C.H. and Y.C.; software, C.H.; validation, C.H.; formal analysis, C.H.; investigation, C.H.; resources, T.F. and M.D.; data curation, T.F.; writing—original draft preparation, C.H.; writing—review and editing, C.H. and Y.C.; visualization, C.H. and M.D.; supervision, Y.C.; project administration, W.T.; funding acquisition, W.T. All authors have read and agreed to the published version of the manuscript.

Funding: This work is partially supported by Shanghai Innovation Action Plan Project (No.16511101200) of Science and Technology Committee of Shanghai Municipality.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article.

Acknowledgments: The authors would like to thank all the teachers and students who have worked hard to build the SHU-VAS system. We are looking for more partners to further explore the capabilities of the SHU-VAS system.

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

References

1. Prouzeau, A.; Bezerianos, A.; Chapuis, O. Evaluating Multi-User Selection for Exploring Graph Topology on Wall-Displays. *IEEE Trans. Vis. Comput. Graph.* **2017**, *23*, 1936–1951. [CrossRef] [PubMed]
2. Jakobsen, M.R.; HornbÆk, K. Up Close and Personal:Collaborative Work on a High-Resolution Multitouch Wall Display. *ACM Trans. Comput. Interact.* **2014**, *21*, 11.
3. Nandal, A.; Bhaskar, V. Enhanced Image Fusion Using Directive Contrast with Higher-Order Approximation. *IET Signal Process* **2018**, *12*, 383–393. [CrossRef]
4. Nandal, A.; Dhaka, A.; Gamboa-Rosales, H.; Marina, N.; Galvan-Tejada, J.I.; Galvan-Tejada, C.E.; Moreno-Baez, A.; Celaya-Padilla, J.M.; Luna-Garcia, H. Sensitivity and Variability Analysis for Image Denoising Using Maximum Likelihood Estimation of Exponential Distribution. *Circuits Syst. Signal Process.* **2018**, *37*, 3903–3926. [CrossRef]
5. Yamaoka, S.; Doerr, K.-U.; Kuester, F. Visualization of High-Resolution Image Collections on Large Tiled Display Walls. *Futur. Gener. Comput. Syst.* **2011**, *27*, 498–505. [CrossRef]
6. Pietriga, E.; del Campo, F.; Ibsen, A.; Primet, R.; Appert, C.; Chapuis, O.; Hempel, M.; Muñoz, R.; Eyheramendy, S.; Jordan, A.; et al. Exploratory Visualization of Astronomical Data on Ultra-High-Resolution Wall Displays. In Proceedings of the Software and Cyberinfrastructure for Astronomy IV, Edinburgh, UK, 26 July 2016.
7. Mitrpanont, J.; Roungsuriyaviboon, J.; Sathapornwatanakul, T.; Sawangphol, W.; Kobayashi, D.; Haga, J.H. Extending MedThaiVis-Thai Medical Research Visualization to SAGE2 Display Walls. In Proceedings of the 2nd International Conference on Information Technology, Nakhon Pathom, Thailand, 2 November 2017.
8. McGinn, D.; Birch, D.; Akroyd, D.; Molina-Solana, M.; Guo, Y.; Knottenbelt, W.J. Visualizing Dynamic Bitcoin Transaction Patterns. *Big Data* **2016**, *4*, 109–119. [CrossRef]
9. Prouzeau, A.; Bezerianos, A.; Chapuis, O. Towards Road Traffic Management with Forecasting on Wall Displays. In Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces, New York, NY, USA, 6 November 2016.
10. Leigh, J.; Johnson, A.; Renambot, L.; Peterka, T.; Jeong, B.; Sandin, D.J.; Talandis, J.; Jagodic, R.; Nam, S.; Hur, H.; et al. Scalable Resolution Display Walls. *Proc. IEEE* **2013**, *101*, 115–129. [CrossRef]
11. Mateescu, M.; Pimmer, C.; Zahn, C.; Klinkhammer, D.; Reiterer, H. Collaboration on Large Interactive Displays: A Systematic Review. *Hum.-Comput. Interact.* **2021**, *36*, 243–277. [CrossRef]
12. Febretti, A.; Nishimoto, A.; Thigpen, T.; Talandis, J.; Long, L.; Pirtle, J.D.; Peterka, T.; Verlo, A.; Brown, M.; Plepys, D.; et al. CAVE2: A Hybrid Reality Environment for Immersive Simulation and Information Analysis. In Proceedings of the Engineering Reality of Virtual Reality 2013, Burlingame, CA, USA, 4 March 2013.
13. Andrews, C.; Endert, A.; Yost, B.; North, C. Information Visualization on Large, High-Resolution Displays: Issues, Challenges, and Opportunities. *Inf. Vis.* **2011**, *10*, 341–355. [CrossRef]
14. Teubl, F.; Kurashima, C.; Cabral, M.; Zuffo, M. FastFusion: A Scalable Multi-Projector System. In Proceedings of the 2012 14th Symposium on Virtual and Augmented Reality, NW Washington, DC, USA, 28 May 2012.
15. Cruz-Neira, C.; Leigh, J.; Papka, M.; Barnes, C.; Cohen, S.M.; Das, S.; Engelmann, R.; Hudson, R.; Roy, T.; Siegel, L.; et al. Scientists in Wonderland: A Report on Visualization Applications in the CAVE Virtual Reality Environment. In Proceedings of the 1993 IEEE Research Properties in Virtual Reality Symposium, San Jose, CA, USA, 25–26 October 1993.

16. University of Texas Stallion Powerwall. Available online: http://www.tacc.utexas.edu/resources/visualization/ (accessed on 2 March 2022).
17. Papadopoulos, C.; Petkov, K.; Kaufman, A.E.; Mueller, K. The Reality Deck-An Immersive Gigapixel Display. *IEEE Comput. Graph. Appl.* **2015**, *35*, 33–45. [CrossRef]
18. Kirschner, K.N.; Reith, D.; Jato, O.; Hinkenjann, A. Visualizing Potential Energy Curves and Conformations on Ultra High-Resolution Display Walls. *J. Mol. Graph. Model.* **2015**, *62*, 174–180. [CrossRef] [PubMed]
19. Wall, M.B.; Birch, D.; Yong, M.Y. Opportunities and Considerations for Visualising Neuroimaging Data on Very Large Displays. *F1000Research* **2016**, *5*, 2157. [CrossRef] [PubMed]
20. Klein, B. A Gesture Control Framework Targeting High-Resolution Video Wall Displays. In Proceedings of the 19th International Conference on Information Visualisation, Barcelona, Spain, 22 July 2015.
21. Kister, U.; Klamka, K.; Tominski, C.; Dachselt, R. GRASP Combining Spatially-aware Mobile Devices and a Display Wall for Graph Visualization and Interaction. *Comput. Graph. Forum* **2017**, *36*, 503–514. [CrossRef]
22. Sharma, G.; Radke, R.J. Multi-Person Spatial Interaction in a Large Immersive Display Using Smartphones as Touchpads. *Adv. Intell. Syst. Comput.* **2021**, *1252*, 285–302.
23. Kim, K.; Ren, X.; Choi, S.; Tan, H.Z. Assisting People with Visual Impairments in Aiming at a Target on a Large Wall-Mounted Display. *Int. J. Hum. Comput. Stud.* **2016**, *86*, 109–120. [CrossRef]
24. Rooney, C.; Ruddle, R. Improving Window Manipulation and Content Interaction on High-Resolution, Wall-Sized Displays. *Int. J. Hum. Comput. Interact.* **2012**, *28*, 423–432. [CrossRef]
25. von Zadow, U.; Dachselt, R. GIAnT: Visualizing group interaction at large wall displays. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, New York, NY, USA, 2 May 2017.
26. Rateau, H.; Rekik, Y.; Grisoni, L.; Jorge, J. Talaria: Continuous Drag & Drop on a Wall Display. In Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces, New York, NY, USA, 6 November 2016.
27. Nancel, M.; Wagner, J.; Pietriga, E.; Chapuis, O.; Mackay, W. Mid-Air Pan-and-Zoom on Wall-Sized Displays. In Proceedings of the 2011 annual conference on Human factors in computing systems—CHI '11, New York, NY, USA, 7 May 2011.
28. Nancel, M.; Pietriga, E.; Chapuis, O.; Beaudouin-Lafon, M. Mid-Air Pointing on Ultra-Walls. *ACM Trans. Comput. Interact.* **2015**, *22*, 21. [CrossRef]
29. Welsford-Ackroyd, F.; Chalmers, A.; Kuffner dos Anjos, R.; Medeiros, D.; Kim, H.; Rhee, T. Spectator View: Enabling Asymmetric Interaction between HMD Wearers and Spectators with a Large Display. *Proc. ACM Hum.-Comput. Interact.* **2021**, *5*, 485. [CrossRef]
30. Wittorf, M.L.; Jakobsen, M.R. Eliciting Mid-Air Gestures for Wall-Display Interaction. In Proceedings of the 9th Nordic Conference on Human-Computer Interaction—NordiCHI '16, New York, NY, USA, 23 October 2016.
31. Papadopoulos, C.; Mirhosseini, S.; Gutenko, I.; Petkov, K.; Kaufman, A.E.; Laha, B. Scalability Limits of Large Immersive High-Resolution Displays. In Proceedings of the 2015 IEEE Virtual Reality (VR), Arles, Camargue, Provence, France, 23 March 2015.
32. Nachbaur, D.; Dumusc, R.; Bilgili, A.; Hernando, J.; Eilemann, S. Remote Parallel Rendering for High-Resolution Tiled Display Walls. In Proceedings of the 2014 IEEE 4th Symposium on Large Data Analysis and Visualization (LDAV), Paris, France, 9 October 2014.
33. Takashima, K.; Oyama, T.; Asari, Y.; Sharlin, E.; Greenberg, S.; Kitamura, Y. Study and Design of a Shape-Shifting Wall Display. In Proceedings of the 2016 ACM Conference on Designing Interactive Systems—DIS '16, Brisbane, QL, Australia, 4 June 2016.
34. Weier, M.; Roth, T.; Slusallek, P. Enhancing Rendering Performance with View-Direction-Based Rendering Techniques for Large, High Resolution Multi-Display-Systems. In Proceedings of the GI VR/AR 2014, Bremen, Germany, 25 September 2014.
35. Hansen, T.; Pracejus, L.; Gegenfurtner, K.R. Color perception in the intermediate periphery of the visual field. *J. Vis.* **2009**, *9*, 26. [CrossRef]
36. Sigitov, A.; Kruijff, E.; Trepkowski, C.; Staadt, O.; Hinkenjann, A. The Effect of Visual Distractors in Peripheral Vision on User Performance in Large Display Wall Systems. In Proceedings of the 2016 ACM on Interactive Surfaces and Spaces—ISS '16, New York, NY, USA, 6 November 2016.
37. Sarbolandi, H.; Lefloch, D.; Kolb, A. Kinect Range Sensing: Structured-Light versus Time-of-Flight Kinect. *Comput. Vis. Image Underst.* **2015**, *139*, 1. [CrossRef]
38. Moon, S.; Park, Y.; Ko, D.W.; Suh, I.H. Multiple Kinect Sensor Fusion for Human Skeleton Tracking Using Kalman Filtering. *Int. J. Adv. Robot. Syst.* **2016**, *13*, 65. [CrossRef]
39. Diao, X.; Li, X.; Huang, C. Multi-Term Attention Networks for Skeleton-Based Action Recognition. *Appl. Sci.* **2020**, *10*, 5326. [CrossRef]
40. Arroyo, I.; Giné, F.; Roig, C.; Granollers, T. Analyzing Google Earth Application in a Heterogeneous Commodity Cluster Display Wall. *Multimed. Tools Appl.* **2016**, *75*, 11391–11416. [CrossRef]
41. Capdevila, M.G.; Silveira, I.F.; Faion, B.F.; Martins, V.F. Usability Study for Liquid Galaxy Platform Using Hand and Voice Controllers. In Proceedings of the 2020 15th Iberian Conference on Information Systems and Technologies, Sevilla, Spain, 24 June 2020.