Scientific Visualization, 2024, volume 16, number 2, pages 97 - 105, DOI: 10.26583/sv.16.2.08

Integration of Physical Reality Objects with Their 3D Models Visualized in Virtual Environment Systems

A.V. Maltsev[1]

Federal State Institution "Scientific Research Institute for System Analysis of the Russian Academy of Sciences" (SRISA RAS), Moscow, Russia

[1] ORCID: 0000-0003-1776-814X, avmaltcev@mail.ru

Abstract

The paper describes methods and approaches for integrating, from the user's point of view, three-dimensional virtual models visible to them in a virtual reality headset, and their physical prototypes. This allows for the addition of some important elements of physical reality to the virtual environment, such as tactile and muscle-motor sensations. Developed solutions are based on the use of wireless HTC Vive trackers, which determine their position and orientation in space and have the ability to be attached to various objects in the real world. An original approach to building a tracking system from several controllers of such type with one anchor element selection is proposed, as well as new methods for real-time integration of physical reality objects with their virtual models by means of functional control schemes with blocks for the Vive trackers. Software modules were created based on developed methods and approaches. Approbation of them was carried out in virtual environment system VirSim and showed adequacy and effectiveness of proposed solutions when using in virtual environment systems and training complexes.

Keywords: visualization, 3D modelling, virtual environment, tracking system, VR headset, controllers, integration.

1. Introduction

At present, methods and technologies of virtual reality (VR) are actively introduced into many areas of human activity. In addition to high-tech hardware, the basic component of VR are three-dimensional virtual scenes creating by means of a computer, which contain models of objects and phenomena, the principles of their dynamics and interaction. Depending on the field of application and solved tasks, virtual model prototypes may have the different nature of the origin. For example, when developing training complexes and other tools to teach qualified specialists [1-6], such prototypes are existing and future objects of physical reality. The gaming and film industries may also use fictional elements.

The degree of perception of synthesized environment by the user depends on the presence and quality of implemented influences on such sensations as vision, hearing, touch, etc. Visual perception plays one of the key roles here. The utmost priority is given to increasing the detail of virtual models and creating methods and algorithms for their realistic visualization in real time, that is, with the rendering frame rate of at least 25 times per second. However, when a person is fully immersed in virtual space, tactile sensations from interactions with objects, the weight of these objects, and response of the muscular skeleton while working with virtual instruments also become important. Therefore, development of new methods and approaches for implementing effects of the sense of touch in virtual reality and, in particular, in virtual environment systems is an important and actual task.

In the field of the touch sense imitation when the user contacts with synthesized virtual environment, active research related to the various aspects of this problem is being conducted. Thus, the paper [7] considers the implementation of tactile sensations during human interaction with virtual objects using the example of pressing buttons. The essence of the approach is to combine visual pseudo-haptic feedback from slowing down the speed of virtual hand movement relative to the real one and tactile pseudo-haptic feedback from effecting on the wrist (squeeze and vibrations) by means of a special bracelet. The article [8] proposes methods and approaches for creating the user sensations from wind and heat in virtual space when he is in the immersive virtual reality environment CAVE [9]. They are based on using large fans and infrared lamps. The authors of paper [10] developed own system to simulate the contact of virtual objects with the person's face, lips and teeth based on a set of ultrasonic emitters placing on VR headset. Ultrasonic pulses can be used to create sensations specific to single and multiple touches, as well as vibrations. The article considers examples when the user in virtual space comes into contact with spider web by the face, drinks water from a small fountain, smokes a cigarette, brushes his teeth, feels gusts of wind.

This paper proposes original methods and approaches for implementation of tactile and muscular-motor sensations of a person immersed in synthesized space by means of VR headset, when interacting with a certain range of objects in virtual environment. The main idea is integration of these objects with their physical prototypes from the user's point of view. The novelty of the developed solutions is such integration implementation in real time based on hardware tracking system including several HTC Vive Trackers with the appointment of "anchor" element, as well as original control schemes with own functional blocks for such trackers.

2. Tracking system based on the Vive devices

Usually the user immersed in virtual environment has some embodiment in it that is so-called avatar or virtual observer [11]. The basic components of such observer are two virtual cameras for the user's eyes and his hand models. This allows seeing the environment and interacting with it. Positions and orientations of cameras are set in accordance with information received from VR headset's sensors such as a gyroscope, accelerometer and magnetometer. Images from cameras are visualized as a stereo pair that is transmitted to displays of the headset. Virtual hands can be controlled by means of modern VR gloves or hand-held VR controllers, such as Oculus Touch or HTC Vive Cosmos. The gloves are more expensive than the controllers, but have significantly greater ergonomics and broader capabilities for implementing human interactions with objects in synthesized environment. In this work, we use Manus Prime II gloves and Oculus Rift CV1 headset.

Fig. 1. The positioning system components: HTC Base Station, HTC Vive Trackers on VR headset and glove, anchor tracker.

To solve the task under consideration of integrating some virtual model and its physical prototype, it is necessary that at each time moment the position and orientation of this model relative to the virtual observer repeat the position and orientation of the prototype relative to the real user. The latter can be determined using HTC Vive Tracker device, which is placed on the physical object. The problem is that the data from different VR devices (the headset, gloves and trackers) comes in different coordinate systems. Conversion of all data to one of them can lead to introducing significant errors in the result due to the difficulty of accurately determining the origin and orientation of each system in physical space for finding the transformation matrix. To avoid transformations between coordinate systems inherent in different type of devices, this paper proposes to create a common positioning mechanism for the user's head and hands, as well as physical objects that need to be integrated with their virtual models.

Proposed tracking system includes at least two HTC Vive Base Stations and five HTC Vive Trackers. Base stations define the working area and allow determining the position and orientation for each tracker in a single right-handed coordinate system, the origin of which is located in one of them set to "*c*" mode. Vive Trackers attach to VR headset, VR gloves, and necessary physical objects. In addition, our tracking system includes a so-called “anchor” tracker, relative to which the positions and orientations of all other used trackers are computed (this will be described in more detail in Section 3). This element is the main link between real and virtual spaces. For this purpose, an exact virtual copy of it is added into three-dimensional scene, relative to which we subsequently place elements of the virtual observer, and virtual objects integrating with their physical prototypes. The local coordinate system for virtual model of the anchor tracker must correspond, in terms of the origin location and orientation of the axes, to Vive Tracker's coordinate system described in the documentation from the device manufacturer.

The specified minimum requirements for positioning system components allows the user to synchronize the movements of the virtual observer with their own and receive tactile and muscular-motor sensations when interacting with one selected virtual object. If it is necessary to increase the number of such objects, the system can easily be scaled by means of additional trackers. Figure 1 demonstrates the installation of Vive Trackers on Oculus Rift CV1 headset and Manus Prime II glove, as well as a view of HTC Vive Base Station and the placement of the anchor tracker on a tripod.

3. Functional schemes and tracker block

In this work, virtual object control is performed by means of functional schemes [12]. They are a set of functional blocks of various types (arithmetic, logical, generators, dynamic, etc.), interconnected by communication lines. The inputs of such schemes receive data from virtual control elements that are in a three-dimensional scene (buttons, toggle switches, joysticks, etc.), as well as from real USB devices. At the outputs, signals are computed, on the basis of which the dynamics of controlled objects and their interaction are simulated.

To ensure the operation of positioning system described in Section 2, this we propose the implementation of a special functional block for HTC Vive Tracker. Its appearance is shown in Figure 2.

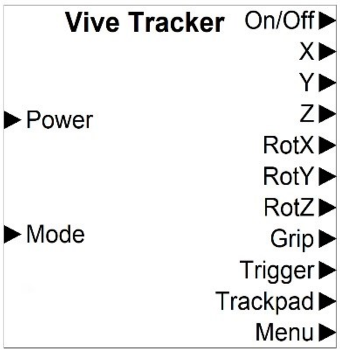

Fig. 2. Functional block for Vive Tracker.

Each instance of such block must be associated with a specific hardware device, which is done by entering the device identifier obtained in the SteamVR application into block settings dialog box. The block's integer inputs specify its on/off (*Power*) and operating mode (*Mode*). The latter determines the role of corresponding tracker in the positioning system, namely, whether it is anchor (value 1) or not (value 0). Note that at any given time only one of all trackers in the system can be the anchor. The block's outputs can be divided into several categories: the state of connection to the device (*On/Off*), position of this device (*X*, *Y*, *Z*), its orientation in the form of Euler angles (*RotX*, *RotY*, *RotZ*), and the state of POGO pins located on the underside of the tracker (*Grip*, *Trigger*, *Trackpad*, *Menu*).

The main task of the proposed block is to receive information from the tracker associated with it and process it in accordance with the selected operating mode. Data requests are performed at least 25 times per second. For this purpose, tools of the OpenVR library are used. Vive Tracker transmits information about its position and orientation relative to HTC Vive Base Station operating in "*c*" mode, in the form of transition matrix M_{t2b} from its own local coordinate system TCS to the BCS coordinate system of this base station (Figure 3). BCS is the world coordinate system for all trackers within the same workspace. In addition to the matrix M_{t2b}, OpenVR allows the block to get the current states of POGO pins. They can be used to implement input buttons for physical object on which the tracker is attached. The state of any such pin is set to one if there is a short circuit between it and the Ground pin, otherwise it is equal to zero.

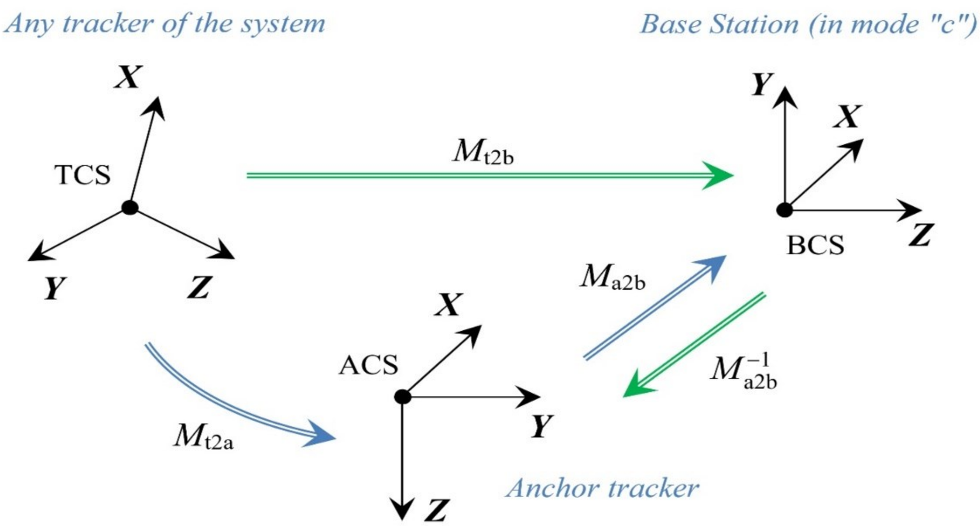

Fig. 3. Transition to the anchor tracker's coordinate system ACS.

To implement an approach to positioning of all trackers relative to one anchor tracker, this work proposes allocating a fragment of the RAM memory that can be accessed by all blocks.

In this case, only the block associated with the anchor tracker can write data to this memory area, and read-only mode is available for all others. The data is the transition matrix M_{b2a} from the world coordinate system BCS to the local coordinate system ACS of the anchor tracker (Figure 3). It can be found by calculating the matrix inverse to M_{a2b}, which is equivalent in meaning to M_{t2b}, that is, it performs a transformation from the local coordinate system of the tracker to the coordinate system of the base station and is accessible by means of functions from OpenVR library. By reading the matrix stored in the shared memory, any tracker of the positioning system under consideration can compute the transition matrix M_{t2a} from its local coordinate system TCS to the coordinate system ACS of the anchor tracker:

$$M_{\text{t2a}} = M_{\text{a2b}}^{-1} M_{\text{t2b}}$$

The desired coordinates P_x, P_y, P_z of the position P of an arbitrary tracker relative to the anchor one will be written in the fourth column of the 4x4 matrix M_{t2a}, and the Euler angles R_x, R_y, R_z of its orientation can be found from a 3x3 rotation matrix contained in the first three columns of M_{t2a}. The paper [13] describes in detail a process of computing the angles. For optimization purposes, it is possible not to compute the position coordinates and orientation angles of the anchor tracker itself, since in its own coordinate system these parameters will always have zero values.

The obtained coordinates P_x, P_y, P_z and rotation angles R_x, R_y, R_z are transmitted by the block to the corresponding outputs *X*, *Y*, *Z* and *RotX*, *RotY*, *RotZ* (Figure 2).

4. Integration of real and virtual objects

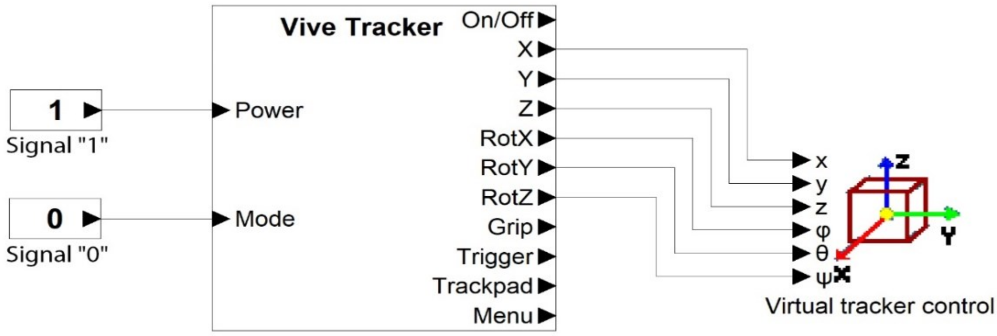

Fig. 4. Functional scheme example with control via tracker block.

To integrate virtual objects from synthesized environment with their physical prototypes in the real world, several conditions must be met. It is first necessary to place the positioning system's anchor tracker in a convenient location in the working area, eliminating the possibility of changing its position and orientation during the system operation. Furthermore, virtual copy of the anchor tracker is added to three-dimensional scene, the local coordinate system VACS of which is identical to the ACS coordinate system of the real device. The height of this model location relative to the surface on which the virtual observer moves must coincide with the height of the anchor tracker location relative to the working area floor. The other trackers of the system are placed on VR headset, VR gloves and necessary physical objects. Exact virtual copies of these trackers are identically set on the corresponding virtual models and then integrated into the scene hierarchy as parent nodes for them. The latter ensures that the model movements are synchronized with its virtual tracker movement controlled on the base of the data from proposed positioning system (see section 2). For the virtual observer, two copies of trackers are placed on hand models and one more is placed at a certain distance from virtual cameras, which is equal to distance between real tracker on VR headset and the user's eyes. It is important to note that at the time of loading three-dimensional scene, the local coordinate system VTCS of every virtual tracker must have the origin coordinates and orientation of its axes the same as those of the anchor tracker model's coordinate system, that is, any VTCS must coincide with the VACS.

The integration process of real and virtual objects is carried out by means of functional scheme. To do this, the above-described tracker blocks are used in an amount equal to the number of Vive Tracker devices in the positioning system considered. Each block is assigned its own physical tracker, and its operating mode is selected by applying the appropriate signal (0 or 1) to the *Mode* input. The *Power* input is set to value 1. Outputs *X*, *Y*, *Z* and *RotX*, *RotY*, *RotZ*, respectively responsible for the position and orientation of real device, are connected by communication lines to the inputs of control element, which ensures a change in the position and orientation of corresponding virtual tracker relative to its initial location in the scene (Figure 4). Since, when loading the scene, the local coordinate system VTCS of each such model coincides with the VACS coordinate system of the anchor tacker model, as agreed, then their new positions and orientations will be established relative to the VACS. Thus, the layout of tracker models in virtual environment at each moment of time repeats the layout of real Vive Trackers, and hierarchical links between tracker models and models of physical objects ensures the required integration of these models with their physical prototypes. Note that displaying the geometry of tracker models in VR headset is optional and may not be performed.

5. Results

Methods and approaches for integration of virtual objects with their physical prototypes proposed in this paper were implemented in virtual environment system VirSim [14] developed at the SRISA RAS. Testing of these solutions was carried out using a scene of the virtual polygon, which is visible to the user through VR headset. The task of developing skills to control a quadcopter using a control panel with LCD screen in various weather conditions was considered. The user has the ability to adjust the direction and speed of flying machine based on image displayed on the control panel's screen and received from quadcopter's camera. In addition, within the visibility range, he has visual control of the apparatus.

Fig. 5. Control panel with HTC Vive Tracker 3.0 set on it.

To solve this task, three-dimensional virtual models of a quadcopter, and its control panel, as well as the virtual observer consisting of two cameras and two hands were placed in the scene of the polygon. The positioning system for user and physical objects described in Section 2 was also used. HTC Vive Tracker 3.0 devices included to this system were placed on Oculus Rift CV1 headset, Manus Prime II gloves (Figure 1) and physical prototype of the control panel (Figure 5). The anchor tracker (HTC Vive Tracker 2.0) was set on a tripod within the working area of the base stations. Virtual analogues of all used trackers were placed on the corresponding models in the scene. In addition, a functional scheme was created that solves two problems. Firstly, this is integration of the control panel and the user's real hands with their virtual models, implemented by means of functional blocks for trackers. Secondly, it provides a control of virtual model of the quadcopter when the user interacts with the control panel's joysticks, toggle switches and buttons.

Fig. 6. Quadcopter model control with integration of real and virtual control panels.

Figure 6 shows the user immersed in virtual environment, who controls the quadcopter model within this environment. In this case, the user not only sees the control panel model, but also feels it with his own hands.

Approbation of proposed methods and approaches in the VirSim software complex showed that they are effective for implementation of tactile and muscular-motor sensations when interacting with a selected range of objects, which are important from the point of view of the problem being solved, in virtual environment systems, training complexes and other applications.

6. Conclusions

This paper presents original methods for implementing integration of virtual objects with their physical prototypes from the user's point of view. It allows user to get tactile and muscular-motor sensations when he is immersed in virtual environment and contacts with this objects. Results obtained in the paper can be used in software development for training complexes and virtual environment systems.

7. Acknowledgements

The publication is made within the state task of Federal State Institution “Scientific Research Institute for System Analysis of the Russian Academy of Sciences” on topic No. FNEF-2024-0002 “Mathematical modeling of multiscale dynamic processes and virtual environment systems”.

References

1. Maltsev A.V., Strashnov E.V., Mikhaylyuk M.V. Methods and technologies of cosmonaut rescue simulation in virtual environment systems // Scientific Visualization, 2021, Vol. 13, No. 4, pp. 52-65.
2. VR trenazher. Virtualnaia realnost v obuchenii. Neftianaia promyshlennost [VR simulator. Virtual reality in learning. Oil industry]. https://www.youtube.com/watch?v=keXlfMKyxsI. Accessed 22 March 2024. [in Russian]
3. Maltsev A.V. Computer simulation and visualization of wheel tracks on solid surfaces in virtual environment // Scientific Visualization, 2023, Vol. 15, No. 2, pp. 80-89.
4. Bruguera M.B., Ilk V., Ruber S., Ewald R. Use of virtual reality for astronaut training in future space missions – spacecraft piloting for the Lunar Orbital Platform – Gateway (LOP-G) // 70th International Astronautics Congress, Washington D.C., 2019.
5. Garcia A.D., Schlueter J., Paddock E. Training astronauts using hardware-in-the-loop simulations and virtual reality // AIAA SciTech Forum, Orlando, FL, 2020.
6. Mikhaylyuk M.V., Timokhin P.Yu. Memory-effective methods and algorithms of shader visualization of digital core material model // Scientific Visualization, 2019, Vol. 11, No. 5, pp. 1-11.
7. Pezent E., Macklin A., Yau J.M., Colonnese N., O'Malley M.K. Multisensory Pseudo-Haptics for Rendering Manual Interactions with Virtual Objects // Advanced Intelligent Systems, 2023, Vol. 5, pp. 1-13.
8. Hülsmann F., Mattar N., Fröhlich J., Wachsmuth I. Simulating Wind and Warmth in Virtual Reality: Conception, Realization and Evaluation for a CAVE Environment // Journal of Virtual Reality and Broadcasting, 2014, Vol. 11, No. 10, pp. 1-21.
9. Kirvan P. CAVE (Cave Automatic Virtual Environment). https://www.techtarget.com/whatis/definition/CAVE-Cave-Automatic-Virtual-Environment. Accessed 22 March 2024.
10. Shen V., Shultz C., Harrison C. Mouth Haptics in VR using a Headset Ultrasound Phased Array // CHI'22: Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems, April 2022, Article No. 275, pp. 1-14.
11. Maltsev A.V. Computer simulation of video surveillance complexes in virtual environment systems // Scientific Visualization, 2022, Vol. 14, No. 2, pp. 88-97.
12. Mikhaylyuk M.V., Torgashev M.A. Vizualnyi redaktor i modul rascheta funktsionalnykh skhem dlia imitatsiono-trenazhernykh kompleksov [The visual editor and calculation module of block diagrams for simulation and training complexes] // Programmnye produkty i sistemy, No. 4, 2014, pp. 10-15 [in Russian].
13. Sablin I.P., Mikhaylyuk M.V., Omelchenko D.V., Kononov D.A., Loginov D.M. Vychislenie uglov Teita-Braiana orientatsii trekera HTC VIVE [Calculation of Tait-Bryan Angles of HTC VIVE Tracker Orientation] // Trudy NIISI RAN, Vol. 13, No. 1-2, 2023, pp. 25-31 [in Russian].
14. Mikhaylyuk M.V., Maltsev A.V., Timokhin P.Ju., Strashnov E.V., Krjuchkov B.I., Usov V.M. Sistema virtual'nogo okruzhenija VirSim dlja imitacionno-trenazhernyh kompleksov podgotovki kosmonavtov [The VirSim virtual environment system for the simulation complexes of cosmonaut training] // Pilotiruemye polety v kosmos, Vol. 4, No. 37, 2020, pp. 72-95 [in Russian].