Visualization and virtual environment technologies in the tasks of cosmonaut training

A.V. Maltsev[1], M.V. Mikhaylyuk[2]

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

[1] ORCID: 0000-0003-1776-814X, avmaltcev@mail.ru
[2] ORCID: 0000-0002-7793-080X, mix@niisi.ras.ru

Abstract

This paper presents original methods and approaches to implement virtual environment system for one of the International Space Station module. Common structure of such system is described, based on using modern technologies of virtual reality (VR), including VR headset Oculus Rift CV1 and Kinect device. To provide human interaction with an environment synthesized by means of computer, a model of virtual observer is introduced into three-dimensional scene of space module interior. A problem of combining such observer with the operator located in real world is considered in detail. Proposed solutions can be used as a part of cosmonaut training complexes in order to orient trainees in the internal space of the module as well as to develop their skills of onboard equipment control.

Keywords: virtual reality, 3D environment, training complex, simulation, visualization, graphics processor, Kinect, tracking, stereo glasses.

1. Introduction

Dynamic direction within computer simulation field is development and adoption of new virtual environment systems and simulation-training complexes built on the base of virtual reality (VR, [1]) technologies. In fact, VR is becoming one of full-fledged tools to train qualified staff and operators of complex equipment. Advisability and efficiency of this approach were studied by researchers [2].

Using VR technologies allows carrying out the most fully immersion of trained operator in a 3D virtual environment. Moreover, several significant benefits are achieved compared to non-VR approaches. First, there is no need to create a real model of an environment for each training complex and system. Implementation and service of such models often require substantial cash investments. Sometimes it's impossible at all. When VR solutions are used, the environment is fully replaced by a virtual model displayed to the operator's eyes through virtual reality helmets and headsets. Graphic designer creates this model by means of 3D modelling system, for example Autodesk 3ds Max, Autodesk Maya and so on. It's much easier to support and update virtual models than real ones. One more advantage from application of VR technologies in virtual environment systems and training complexes is to improve a quality for visual perception by the operator of virtual space. It brings this space more to a real prototype and allows enhancing the effectiveness of qualified staff training by means of such systems.

Technology development for synthesis and real-time visualization of virtual environments as well as a person interaction with them is serious scientific problem which is relevant for training systems in various fields of application. For example, paper [3] describes computer simulation of underwater research vehicle and creation of its virtual environment. In computer animation presented by reference [4], application of the simulator using VR headset

and Oculus Touch devices for staff training in oil industry is demonstrated. Virtual environment systems have a particular relevance in tasks of cosmonaut training. Research in crew training by means of VR was started by NASA laboratories in the second half of the last century [5]. Modern domestic and foreign developments based on virtual environment technologies are devoted to realistic imitation of space ships' exterior and interior [6], training cosmonauts for extravehicular activity [7, 8], teaching them to control complex robotic means [9, 10] and etc.

This paper presents virtual environment system implementation methods and approaches for an interior of the International Space Station's (the ISS) one module. Pirs was chosen as such module. It is an important element of the Russian Orbital Segment, because it includes the docking unit for transportation spacecraft of the Soyuz and Progress type, as well as provides a spacewalk opportunity. Proposed solutions are based on using modern VR technologies. Their novelty lies in addition to synthesized 3D environment a special model of virtual observer having hierarchical structure and using dynamics elements (virtual motors) to interact the operator with virtual interior's objects. Original approach is also application and computation of control schemes when complex technical devices are simulation such as control panel located inside the module. Developed system can be used to examine the interior details, to orient operators in the arrangement of important elements (devices, hatches, etc.) and to train some skills needed on board the ISS. One of these tasks is learning how to use on-board control panel.

2. Virtual environment system

Virtual environment system proposed by us moves the operator from real world to virtual interior of Pirs space module. The working principle of the system is as follows. The user puts on Oculus Rift headset and enters the workspace of Microsoft Kinect device. At the time of initialization, the system loads three-dimensional virtual scene with the interior and a virtual observer. The observer includes eyes (two virtual cameras) and models of hands. After loading the scene, virtual observer is bound with the operator. An image seen by virtual eyes is transmitted to the person's eyes with using VR headset. Thanks to several tracking systems (tracking of the operator's hands and torso is implemented based on the Kinect, head is tracked by the Rift headset), positions and orientations of the observer's elements in virtual environment are synchronized with positions and orientations of the corresponding parts of the operator's body in real space. The operator thus gets an opportunity to move and turn inside virtual interior, turn his head for viewing environment, and interact with interior elements by means of his hands. For example, buttons of virtual control panel inside space module are such elements. Clicking on them causes some actions (turning on or off the lighting, opening or closing the hatches etc.) defined by control scheme. This scheme is developed when creating virtual scene and saved with it.
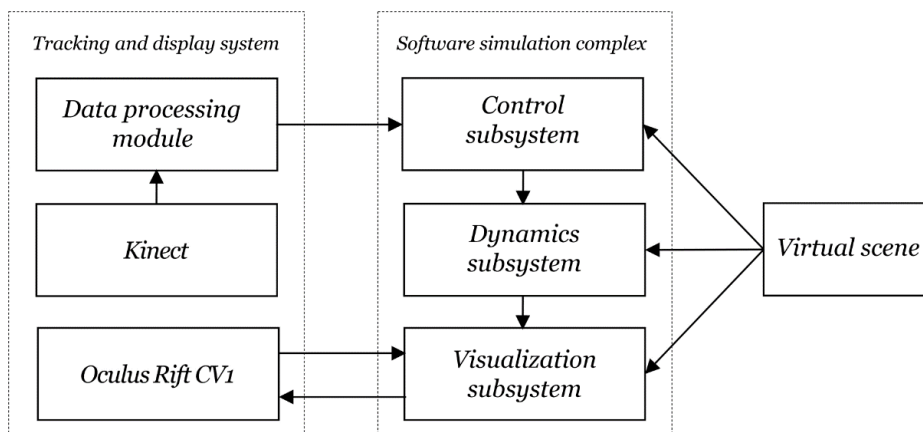

Fig. 1. Virtual environment system structure.

To implement the system described above, we were developed original software complex. It consists of three subsystems (fig. 1), which are responsible for controlling virtual observer and other elements of virtual environment, computing the dynamics of objects, collision detection and response, and also virtual environment visualization with per-pixel calculation of realistic lighting. Each of these three subsystems provides real-time functionality and is original product, designed from the ground up without using third-party software components. To achieve our goal, we also created own virtual scene with model of Pirs space module interior and virtual observer (fig. 2). Except visible elements, it contains many bounding volumes (boxes, spheres, cylinders, etc., fig. 3). They are used for collision detection and response when interactions of the operator with objects are occurred. The scene was made in Autodesk 3ds Max and converted to a format supported by our software complex. In addition, a control scheme for the virtual control panel (fig. 4) was developed by means of our own scheme editor. Implementation of hardware part that is responsible for tracking of the operator's head, body, hands and necessary for rendered image transmission into his eyes is based on the Kinect and Oculus Rift devices.
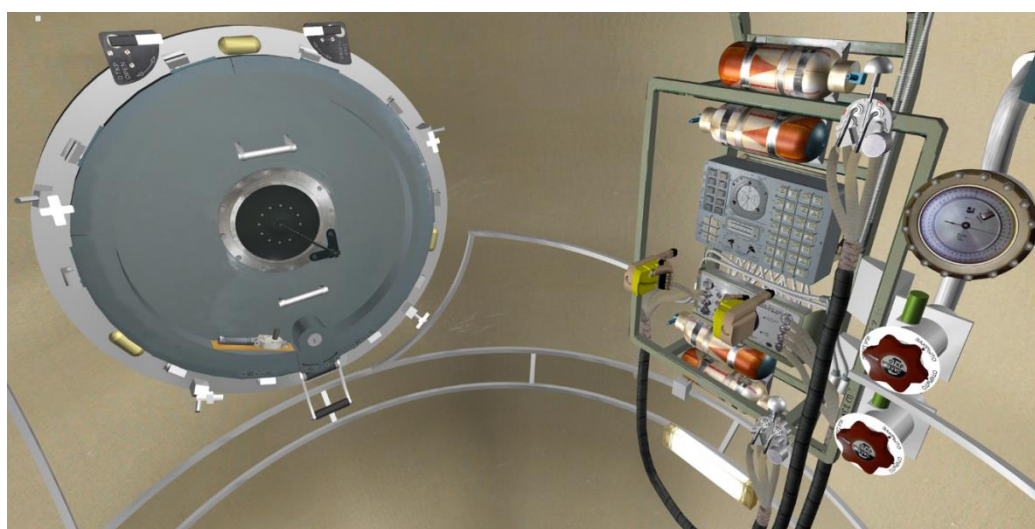

Fig. 2. 3D scene of Pirs module interior with control panel and virtual observer.
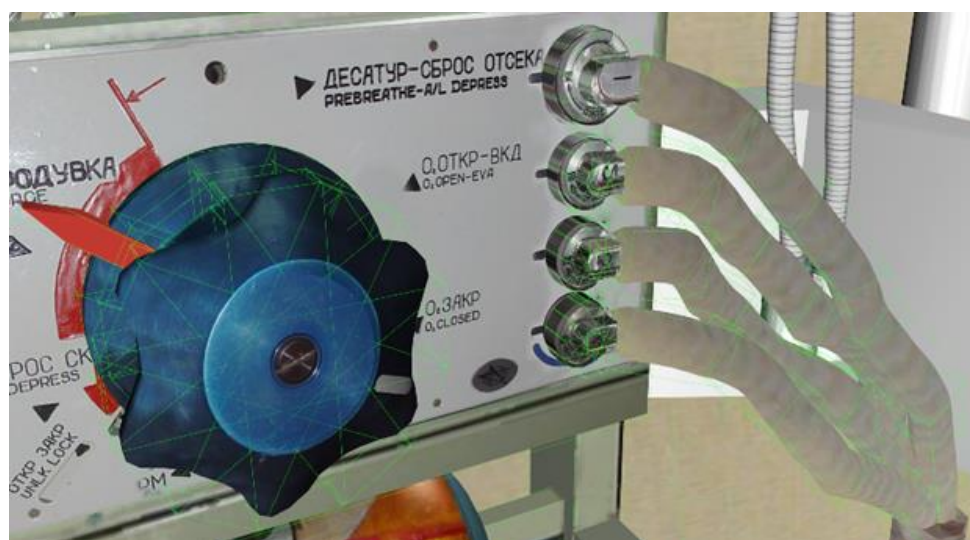

Fig. 3. Bounding volumes (highlighted in green).

Fig. 4. Virtual control panel.

3. Combining virtual observer with the operator

Virtual observer added to three-dimensional scene has hierarchical structure (fig. 5) including several reference points, hand models and two virtual cameras. Since the tasks of current virtual environment system version are visual inspection of Pirs module interior by the operator and learning the basics of control with using the panel buttons, all fingers of virtual hands except the index fingers are bent. Torso point includes three linear and one rotational motors (the last is located below the rest in their hierarchy), which are used to move the observer around the module and rotate it about vertical body axis. In addition, three linear motors are set in reference point of every wrist. They provide movements of virtual hand models parallel to vertical, transverse and sagittal body axes. All motors are controlled by software simulation complex with using own control scheme.

At the time of system initialization, it is necessary to combine virtual observer with real operator. This process involves several basic steps. At first, the operator enters tracking zone and takes initial pose similar to the one set for virtual observer when creating the scene. To effectively use working area of the Kinect device, XY plane of its coordinate system KCS (*the Kinect coordinate system*) must be parallel to frontal body plane of user when he is in this pose. Using the Kinect, virtual environment system gets coordinates of the operator skeleton's reference points (fig. 6) and fixes them as initial. Next, coordinate transformation from real to virtual space is defined. To this effect, it is necessary to form transformation matrix M from KCS to virtual observer coordinate system. To combine the operator with the observer, their coordinate systems BCS (*body coordinate system*) and VBCS (*virtual body coordinate system*) coinciding at any time after initialization are chose equivalent by locations of origins and directions of axes. Let the origin of BCS be at point F of the user's torso, X-axis be parallel to transverse axis of his body and directed towards right hand, Z-axis – vertically up, Y-axis – forward (fig. 7).
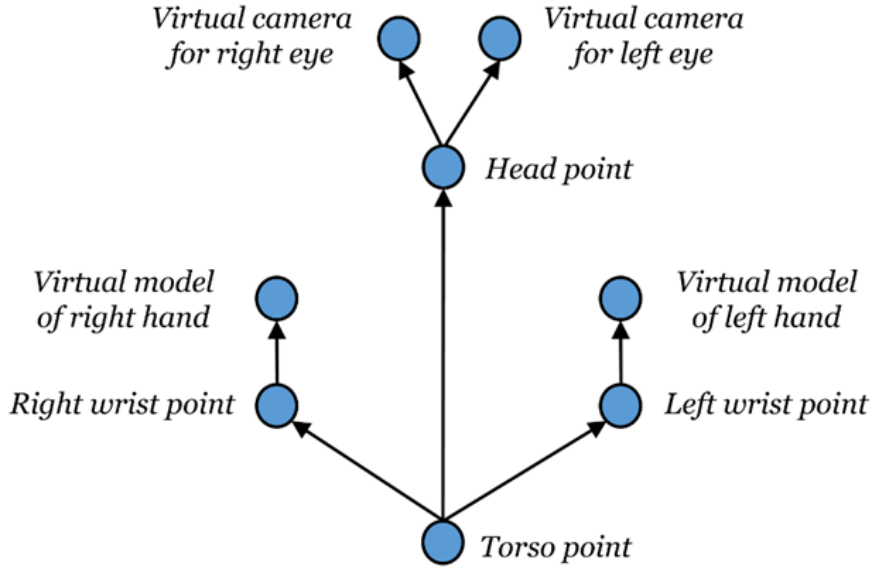
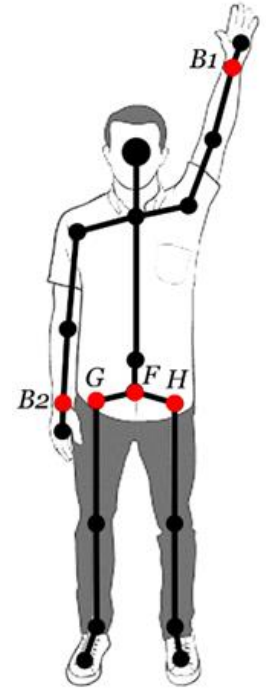Fig. 5. Hierarchical structure of virtual observer.

Fig. 6. Skeleton reference points.

The system VBCS has similar directions of axes, and its origin is located at virtual observer's torso point. Then transformation matrix M from KCS to VBCS (BCS) is

$$M = \begin{pmatrix} 1 & 0 & 0 & -F_{0,x} \\ 0 & 0 & -1 & F_{0,y} \\ 0 & 1 & 0 & -F_{0,z} \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{1}$$

where $(F_{0,x}, F_{0,y}, F_{0,z})$ are coordinates of point F in KCS when the operator is in initial pose.

4. The operator tracking

Tracking of the operator in developed virtual environment system is performed by means of two hardware devices. Oculus Rift CV1 VR headset is used to get rotational angles of head, which are sent to software complex to provide corresponding orientations of virtual cameras for left and right eyes. Tracking of body and hands is implemented on the base of Microsoft Kinect by following algorithm:

1)   initialization of the system with adoption by the operator of given initial pose;
2)   receiving coordinates of the operator skeleton's reference points from the Kinect device by means of the Kinect API;
3)   minimization of sharp change error in points coordinates based on a history about last movements of the operator;
4)   calculating displacement and rotation angle of the operator's body relative to initial pose based on current coordinates of torso points;
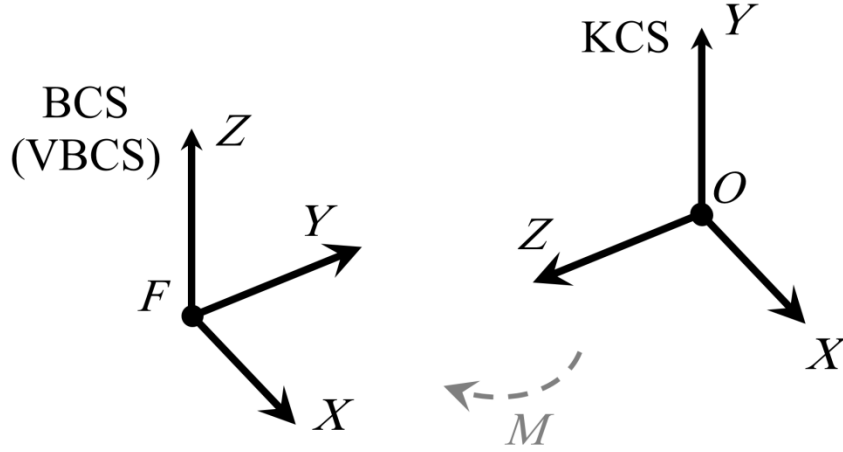5)   calculating displacement of wrist points relative to their initial positions.

Fig. 7. KCS, BCS and VBCS coordinate systems.

First step is performed only once when virtual observer is combined with the operator (see section 3). After the system initialization is complete, own data processing module (fig. 1) real-time receive coordinates of the operator skeleton's reference points from the Kinect, which are computed on the base of information from RGB-camera and IR sensor embedded in the device. In this work, we use points of wrist ($B1$, $B2$) and torso (F, G, H) highlighted in red on figure 6. Due to errors of the camera and sensor, there may be significant changes in coordinates of tracked points from frame to frame even with a static position of the operator. Therefore, before using coordinates received from the Kinect, data processing module smooths wrong fluctuation based on early obtained and accumulated information about reference points. Calculation of the operator body rotation angle α around his vertical axis coinciding with virtual observer rotation angle is performed by the module using current coordinates of skeleton points G and H (fig. 6, 8) in initial system BCS (i.e. when the operator and observer are in initial poses):

$$\alpha = \arccos(Y{\cdot}V) - \pi/2, \qquad V = (H' - G') / |H' - G'|,$$

where $Y = (0,1,0)$ − basis vector of BCS; $H' = M{\cdot}H$, $G' = M{\cdot}H$; M − transformation matrix from KCS to BCS (VBCS) described by the equation (1). Displacement S_F of the operator relative to initial location is equal to current coordinates of point F in BCS: $S_F = M{\cdot}F$.
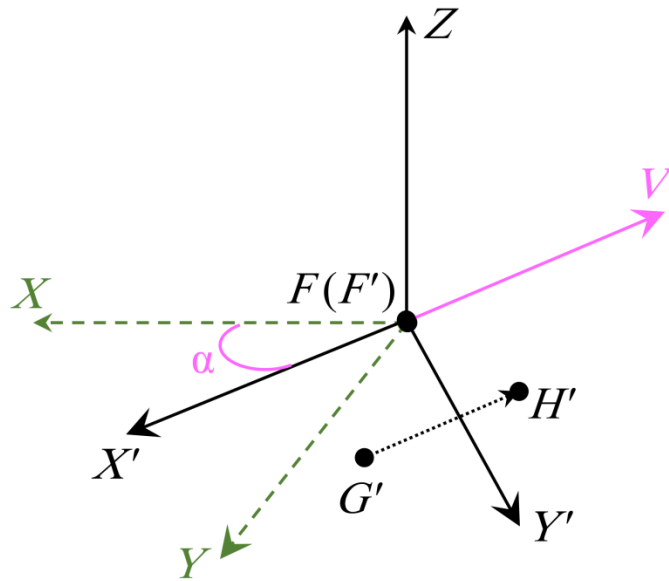

Fig. 8. Calculation of body rotation angle.

Since wrist points of virtual observer have hierarchical link with its torso point, the software complex provide turns and motions of hand models together with the operator's body. To control virtual hands, displacements of wrist points relative to their initial positions $B1'_0 = M{\cdot}B1_0$ and $B2'_0 = M{\cdot}B2_0$ in current local coordinate system of body must also be computed. When the operator moves and rotates, local system BCS (VBCS) transforms to BCS'' (VBCS''), but thanks to link of hands with body initial local coordinates $B1$ and $B2$ don't change, i.e. $B1''_0 = B1'_0$, $B2''_0 = B2'_0$. Current coordinates of these points in BCS'' (VBCS'') are computed as

$$B1'' = M''{\cdot}B1, \quad B2'' = M''{\cdot}B2, \quad M'' = M_R{\cdot}M_T{\cdot}M,$$

where M – the matrix from equation (1), M'' – transformation matrix from BCS to BCS'', M_T and M_R – matrices of system origin translation to current position of point F and system rotation by angle α around Z-axis,

$$M_T = \begin{pmatrix} 1 & 0 & 0 & -S_{F,x} \\ 0 & 1 & 0 & -S_{F,y} \\ 0 & 0 & 1 & -S_{F,z} \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad M_R = \begin{pmatrix} \cos\alpha & \sin\alpha & 0 & 0 \\ -\sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Then required displacements S_{B1}, S_{B2} of points $B1$ and $B2$ relative to their initial positions will be equal to

$$S_{B1} = B1'' - B1''_0 = B1'_0 - M''{\cdot}B1, \quad S_{B2} = B2'' - B2''_0 = B2'_0 - M''{\cdot}B2.$$

Based on obtained values of α, S_F, S_{B1} и S_{B2}, data processing module forms commands for setting new positions and orientations of virtual observer's nodes, which are transmitted to the control subsystem by means of special protocol.

5. Using control schemes

In this work, simulation of virtual control panel as well as movements and turns of the observer are based on using own control schemes. They are stored with scene file and loaded to the control subsystem, which is a part of our software complex. This subsystem takes as its input a data about states of the control panel's buttons, as well as commands for setting new positions and orientations of the operator's torso and hands from data processing module. Based on this information, it computes schemes and synthesizes control signals which are sent to the dynamics subsystem. The last uses this signals for simulating virtual motors of the observer, executing some commands and generating events in virtual environment (turning on/off the lighting of space module by pressing buttons of the control panel etc.).
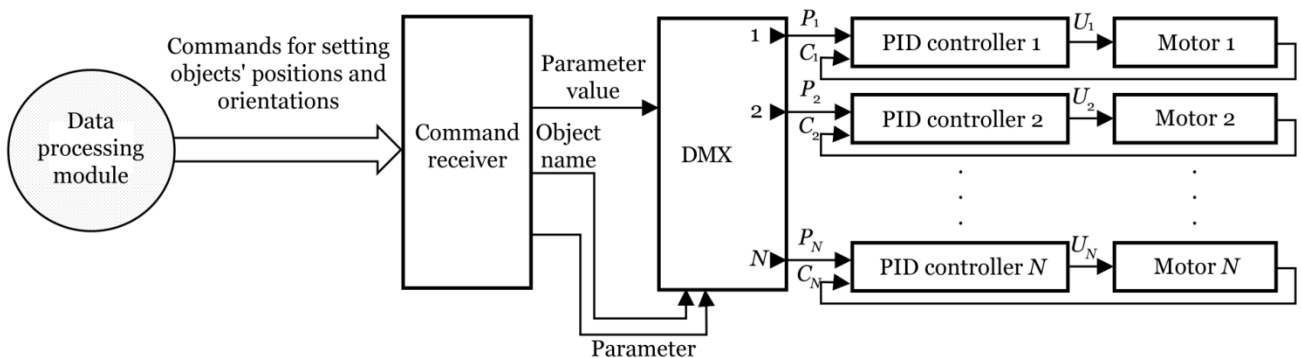

Fig. 9. Control scheme structure for virtual observer.

As an example, consider using control scheme for the observer. Structure of this scheme is illustrated in figure 9. Command receiver parses each packet coming from data processing module and extracts from it name of controlled object, parameter (coordinate or angle) and its new value which must be set for this object. Demultiplexer (DMX) configured by means of text file uses obtained name and parameter to give new value to one of element pairs, which

includes a proportional-integral-derivative (PID) controller and virtual motor. The last implements movement/turn of controlled object along/around X, Y or Z-axis of its local coordinate system. The dynamics subsystem uses PID controller to generate control voltage U, supplied to motor. U is computed based on necessary value P of changed parameter and its current value C, varying over time. Voltage is supplied while $C \neq P$.

6. Results

By using proposed methods and approaches, virtual environment system prototype for Pirs space module interior was made. Hardware platform for implementing this system consisted of high performance computing unit based on Intel Core i7-8700K processor and NVIDIA RTX 2080 graphics card, Oculus Rift CV1 headset and its sensor, Microsoft Kinect v2.0 device, display. Software simulation complex were realized with using object-oriented language C++, the OpenGL 3D graphics library, shader language GLSL 4.3 and the CUDA parallel computing architecture. Considering the above hardware configuration, stereo visualization of high polygonal virtual model of Pirs space module interior is performed with frame rate about 300-400 FPS.

Figure 10 illustrates an example of applying developed system to immerse a person in virtual environment. The operator feels that he is inside Pirs space module, sees its interior and virtual hands. He presses the control panel buttons by means of virtual index fingers, which movements is a copy of his wrists' movements.

Fig. 10. Immersing the operator in virtual interior of Pirs space module.

7. Conclusions

At this paper we propose solutions for creating virtual environment system based on modern virtual reality technologies. Using stereo headset provides high level of immersing the operator in virtual environment. Proposed solution may be adapted for other models of the ISS modules.

However, the Kinect device doesn't provide full interaction of the user's hands with virtual objects. It was used as the initial stage of developing methods and approaches for realization of training systems based on new virtual reality technologies. To next improve created system it is planned to implement full tracking the operator's hands and their interaction with objects of space module model interior by means of specialized devices with sensors.

In this case, optimal way is to use of virtual reality gloves [11, 12]. They provide a control of bending for each finger and allow flexible programming a feedback with possibility to independently effect on separate hand areas.

8. Acknowledgements

References

1. Boletsis C. The New Era of Virtual Reality Locomotion: A Systematic Literature Review of Techniques and a Proposed Typology // Multimodal Technologies and Interaction, 2017, 1(4), 24.
2. Selivanov V.V., Selivanova L.N. Virtualnaia realnost kak metod i sredstvo obucheniia [Virtual reality as method and means of learning] // Educational Technology and Society, 2014, Vol. 17, № 3, pp. 378-391. [in Russian]
3. Bobkov V.A., Morozov M.A., Bagnitsky A.V., Inzartsev A.V., Pavin A.M., Scherbatyuk A.F., Tuphanov I.E. Simulation system for underwater research vehicle // Scientific Visualization, 2013, Vol. 5, № 4, pp. 47-70. [in Russian]
4. VR trenazher. Virtualnaia realnost v obuchenii. Neftianaia promyshlennost [VR simulator. Virtual reality in learning. Oil industry]. https://www.youtube.com/watch?v=keXlfMKyxsI. Accessed 26 June 2020. [in Russian]
5. Fisher S.S., Humphries J., McGreevy M., Robinett W. The Virtual Environment Display System // ACM Workshop on Interactive 3D Graphics, 1986.
6. Masalkin A.I., Torgashev M.A. Experience of Using the Simulation Systems of Visual Environment in the Simulators of Manned Space Vehicles // Piloted Flights in Space, 2015, № 2, pp. 36-42. [in Russian]
7. Stend podgotovki ekipazhei mezhdunarodnoi kosmicheskoi stantsii s ispolzovaniem elementov virtualnoi realnosti [Stand of training the crews of the international space station using virtual reality elements]. http://www.gctc.ru/main.php?id=135. Accessed 26 June 2020. [in Russian]
8. How NASA uses virtual reality to train astronauts. https://spacecenter.org/how-nasa-uses-virtual-reality-to-train-astronauts. Accessed 26 June 2020.
9. Sergeev A.V., Gook M.Yu. Mobile Space Robot Control with Use of Virtual Reality // Piloted Flights in Space, 2018, № 4, pp. 44-52. [in Russian]
10. Kryuchkov B.I., Usov V.M. Developing VR-models to Train Cosmonauts How to Interact with a Robot-Crew Assistant and to Identify Potential Areas of its Application // Proceedings of the International Scientific and Technical Conference "Extreme Robotics", 2013, pp. 230-244. [in Russian].
11. Perret J., Vander Poorten E. Touching Virtual Reality: A Review of Haptic Gloves // Proceedings of 16th International Confernce on New Actuators, 2018, pp. 270-274.
12. MANUS. https://manus-vr.com. Accessed 26 June 2020.