THERAPY OF UPPER LIMBS BY MEANS OF VIRTUAL REALITY TECHNOLOGIES

Branislav SOBOTA*, Štefan KOREČKO*, Sára JAVORKOVÁ*, Marián HUDÁK**
*Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics,
Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic, tel. +421 55 602 6523,
E-mails: branislav.sobota@tuke.sk, stefan.korecko@tuke.sk
**Siemens Healthcare s.r.o., Tr. SNP No. 37, 040 11 Košice, Slovak Republic
E-mail: marian.hudak@siemens-healthineers.sk

ABSTRACT

This paper deals with an approach to upper limbs therapy that uses virtual reality technologies. The previous methods and subsequent improvements of these procedures by means of a skeletal model of the upper limb in a virtual environment are presented here. So, main focus of the paper is on the description of calculation related to the bone rotation system within appropriate skeletal model. The therapist can add either more virtual upper limb objects or more virtual training objects to the virtual environment and thus expand/change the scene or the therapy complexity. The functions used in the limb movement calculations are useful for creating additional animations with various objects. With this system, the patient can be stimulated under the supervision of a therapist to practice certain rehabilitation procedures. Due to the use of collaborative web-based virtual reality, the therapy can be also applied in a remote form. The way in which the underlying idea of rehabilitation process is implemented and it is also described. In the conclusion are the some notes about system testing and evaluation including description of a therapist interface.

Keywords: *upper limbs rehabilitation, inverse kinematics, A-frame, virtual reality*

1. INTRODUCTION

According to [1], rehabilitation represents “all methods that help to restore and regenerate the patient's original state using various methods and techniques. Comprehensive rehabilitation is an interrelated, coordinated and targeted process that has its specialists, diagnostics and established procedures”. Thus, its goal is to improve the course of the disease, reduce the deficit and ensure that the patient returns to the ordinary life in the best condition possible. After rehabilitation under professional supervision, the patient usually continues with domestic rehabilitation. Home rehabilitation should be regular and honest in order to increase the speed and quality of the patient's return to the original state. However, in many cases it is compromised by laziness, disinterest or the fact that rehabilitation exercises are considered boring. What if patients could use virtual reality in home rehabilitation, together with remote professional supervision? It may become more entertaining and in some cases also more effective.

Virtual reality (VR) affects a person's perception and thus his or her nervous system. Development of VR and related technologies, in recent years, allows its application in areas where it was previously difficult or not possible at all, especially in human-machine interaction [2] [3]. One such area is the area of neurorehabilitation. Many people developed impaired motor pathways in the brain due to head injuries or diseases, leading to physical disabilities. One of the most common causes of physical disability for adults is stroke. A stroke involves the reduction of blood flow and insufficient oxygenation of the brain tissue, which can cause failure or loss of function of the affected brain area. Often, this results in an inability to move one or more limbs on one side of the body (hemiparesis). Also, a person may not be able to understand speech, may have memory impairments or a loss of vision to one side. Previous research of the Institute of Measurement Science of Slovak Academy of Sciences (SAS) and the Centre for Cognitive Science Comenius University of Bratislava (CCS). (cooperating organizations within the project solved in the authors' home laboratory LIRKIS) shown that neurorehabilitation helps to achieve beneficial results in the treatment of patients with paresis. Technologies such as Mirror Boxing or BCI-RAS robotic arm (Robo-Arm) [17] system were used in the research together with EEG. Until now, however, virtual reality technologies have not been used in this research phase.

As part of the project, the VR system was developed in the LIRKIS laboratory to support this rehabilitation process (from IT point of view). Its initial versions used fixed animations of the upper limbs movement including the grip of virtual training objects. The change to the use of inverse kinematics in the skeletal structure of the limb (described below) brings much more flexibility to the virtual environment and subsequently to the whole therapeutic process.

2. RELATED WORK

Virtual reality can help patients who recover from serious injuries or operations. According to [4], it can improve rehabilitation at home. It is usually based on performing simple tasks and trainings, thanks to which the patient is able to re-learn various movements, regain new abilities and skills.

In the field of medical applications, the VR and related technologies offers a variety of interaction techniques rendering effective for treatment and rehabilitation of human movement disorders [5] [6]. Upper limb rehabilitation process [7], practiced using head mounted displays with other equipment based on the rehabilitation needs [8] can serve as a commendable example. In a virtual environment, it is possible to simulate different conditions of a virtual hand, visualize different situations for the user and assign them tasks to perform utilizing virtual hands [9],

[10]. Due to the pandemic situation, it is difficult to perform personal rehabilitation and training of patients [11]. Novel science-based treatments are considering remote therapy options, where a therapist can communicate with a patient and oversee the training process [12]. In order to the standard VR applications, web-based VR can provide sharing of virtual content and user collaboration in the real-time [13]. A widely used rehabilitation application is a Kinect based rehabilitation from Microsoft called REMOVIEM [14]. It is created for people with multiple sclerosis who need to practice the movement of the whole body. The application offers several exercises that the patient can practice on their own. The patient needs a Kinect device and a computer with a monitor, which shows a virtual environment for exercise. In addition to the fact that the patient can exercise without the therapist's supervision, the therapist is able to control his or her results and adapt the program plan to the patient. Another example is the IREX (Interactive Rehabilitation and Exercise Systems) project from GestureTek [15]. Unlike the previous one, this project seeks to disconnect people from the reality of lengthy and boring rehabilitation. Each part of rehabilitation is given in the form of simple games (gamification), in which both children and adults have fun. The application contains several games that deal with balancing, rotating, bending or lifting things. There are several games for a given exercise, so the patient can choose the one that suits him. The available games are, for example, football, drumming and volleyball.

Health problems caused by stroke include inability to move one or more limbs on one side of the body (hemiparesis) [16]. Neurorehabilitation is one of the main therapies of stroke treatment. The basic pillar used in neurorehabilitation is neuroplasticity, which is brain property. Due to this property, it is possible that after damage to the brain area, another area that was not originally used for the given function will partially take over the function of this area.

Previous experiments (mainly based on the Mirror Box) of cooperating partners (SAS, CCS) were performed as a series of exercises on healthy volunteers and on patients after a stroke [17]. The experiment consisted of several steps:

- relaxation phase
- motion observation
- performing and observing movement in a mirror box

During the exercises, brain activity was constantly recorded using ten electrodes placed above the sensorimotor area and one electrode above the occipital area of the brain. The test result was changes in the oscillating rhythms in the relaxed state compared to the state in motion. After several exercises with the Mirror-box, the patient was able to switch to the BCI-RAS robotic arm exercises (Fig. 1). Follow-up technique used in the research is rehabilitation using motoric projection (MP). The patient has the idea of the movement of his limb, but the movement itself is not performed. Although the combination of MP and classical rehabilitation is more effective, it also has its disadvantages. The movement of a limb performed by a therapist does not have to correspond with the patient's intention.

From the previous experiment, it was possible to predict when the patient realistically imagines the movement of the hand using the resulting values from the EEG. As in the previous test, the patient was in a state of relaxation when instructed to imagine hand movement. When the oscillating rhythms changed, a robotic arm was started, which moved the wrist up and down. The goal was to aid with patient movement through the robotic arm. Neurorehabilitation consists of the functional replacement of a damaged brain area with another area of the brain that was not originally intended for it. This goal cannot be achieved with classical rehabilitation, where the movement of the affected limb is performed by a therapist. In robot-assisted rehabilitation, the movement of the limb is performed by a robotic arm. The robotic arm individually and optimally adjusts the training and evaluates the progress of the patients based on results from EEG. The goal of the MP-BCI system is to use patient mental activity associated with the movement of the robotic arm and thus return him the lost ability to move the limb. However, it requires a great concentration on the activity. The connection with robotic arm was also used in our system.

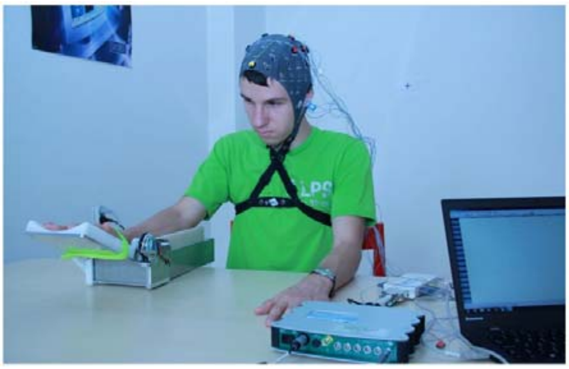


Fig. 1 Robotic arm BCI-RAS [17]

3. SYSTEM DESIGN

A virtual reality system, introduced here, was created to help with neurorehabilitation of a patient with upper limb paresis. It is based on the LIRKIS G-CVE (Global Collaborative Virtual Environments) [18] a development of the authors' home laboratory. The system was divided into a subsystem for the patient and a subsystem for the therapist. An important part of the application is the simulation and visualization of the movement of the patient's upper limb, including gripping of various objects. In principal, animations can be used in two ways. The first one is to use pre-rendered animations, where the corresponding movement or a set of movements is animated in the same way in all cases. The second way is to use 3D objects, which contain data that allow them to be animated in real time according to the current situation. We have chosen the second way and because we need to animate limb movement, where the limbs are used to grab and move objects, the data needed for the animations are about bones of the limbs.

When designing the system, we followed the requirements of collaborators from SAS. The idea of the final product was established, together with an agreement that in addition to software testing, the requirements will be modified if necessary. The project was developed as a part of the LIRKIS G-CVE platform, which utilizes the *A-Frame* and *Three.js* software libraries. Thanks to this, the system can be run on any computing device with a web browser that supports the WebXR web framework. As WebXR is becoming more and more common, these include virtually all current standard web browsers. The system also runs on VR headsets that have a browser, such as MS Hololens or Oculus Quest. For testing and evaluation purposes it is currently hosted at the glitch platform (https://glitch.com/). As it was mentioned before, the system interface consists of two parts:

- *the therapist interface* to set rehabilitation parameters and visualization (Fig. 10) and
- *the patient interface* - for visualization only.

The basic parts of both interfaces have been implemented. As the development progressed, new requirements for animation and test parameters setup, including communication with the BCI-RAS system, have been added. Furthermore, the system is able to perform a sequence of animations, which are aimed at supporting the coordination of the movement of the upper limbs. After entering the parameters and starting the system, a sequence of a certain number of hand animations is performed. Other scenarios should be completed continuously, such as lifting a pen or holding a ball.

3.1. The rehabilitation process implementation design

The rehabilitation is implemented as a sequence of several scenarios with animations (Fig. 2). The therapist is able to enter the number of given scenarios. The time between scenarios is to serve as a time to relax (e.g. just look ahead without any activity). A scenario takes some time to set up. At the beginning of the scenario, an object (e.g. a cup, a key etc.) will appear, which serves as an auxiliary object for animation (for example, when grabbing a cup, the object will be the cup itself). There are two ways to end the scenario:

- scenario timeout or
- the external command (impulse) to perform the animation is recognized (from Robo-arm or interface).

When the command is received and the animation is run, then the scenario time resets. When the animation ends, the system switches to the relax state and the object will be hidden (not displayed). The whole sequence ends if:

- the therapist resets the sequence from his interface or
- all scenarios have been already completed.

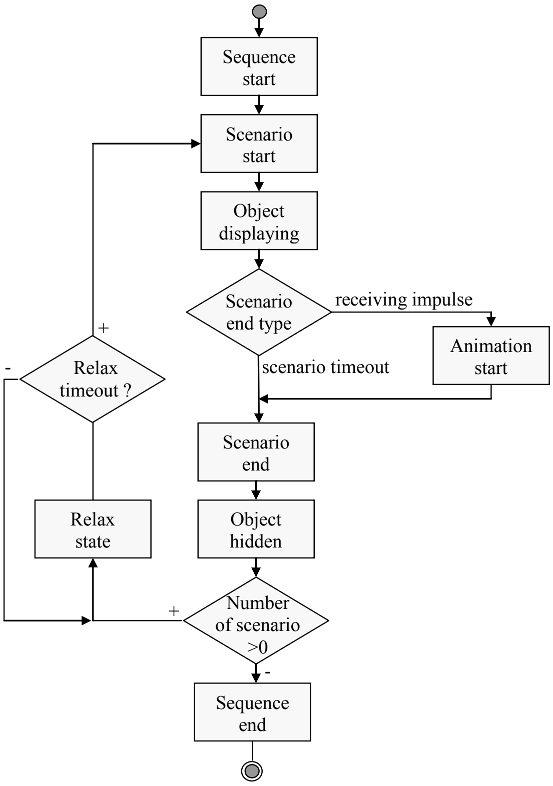


Fig. 2 Rehabilitation process implementation

4. BONE ANGLES CALCULATION

Each animation has its own calculations, although they are very similar. Even the grip of a real hand also differs depending on the object it is about to grab. The core of the application for this part is the processing of the hand movement based on the skeletal model (Fig. 3). *A-frame* can recognize bones and manage them. Every bone has an axis of its own according to the previous one. The position of bones is not recommended to change, because previous bone do not change with it. As in the real world, human needs to change rotation of previous bones to get desirable position of target bone.

Before performing the animation, the final rotations of all parts of the hand are calculated. Subsequently, the application only adjusts the rotation of the bones in the model to get to the end positions. The positions of the bones are most interchangeable. When one of the bones is moved, the whole skeleton moves with it. Therefore, if we, for example, want to change the position of the wrist, it is necessary to change the rotation of the previous joints, as well as in the real world. The calculations of the rotations of the parts of the hand work on the principle of inverse kinematics. A target is determined and algorithms calculate bone angles in the hierarchy to get to the end position with each movement. As the end positions are counted, you can move objects as needed. But if the target objects are not within the reach of the hand, the animation will not perform. In terms of hierarchical joint structure, the end positions of the fingers are understood as leaves. On the rest of this section, we describe the calculation on an example of an animation with a key and a lock. The animation can be described as “a hand picks up a key, inserts it into the keyhole of the lock, rotates it and returns to the starting position”.

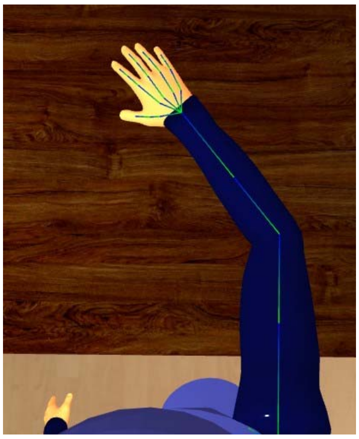


Fig. 3 Used skeletal model of the limb

4.1. Inverse kinematics

The problem, which inverse kinematics is solving, is to find out rotations of every part of object to reach some point in environment. Inverse kinematics is mostly used with robots with more joints to reach some point [19]. It is acceptable for robots to move in each direction, but not with human arm, which has limits. The biggest issue with normal calculation of rotations is that there is a lot of possible solutions for each bone. At the end of the calculations, rotations can be unnatural for a human body.

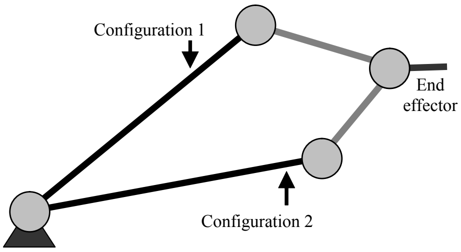


Fig. 4 Inverse kinematics in 2D with two joints [20].

4.2. Angle calculation

A point of environment to reach (the key) and current rotations of every bone in arm are defined. It is sufficient to make calculation with only two joints of arm. The basic joint represents shoulder, which in these animations does not move, only rotates. The next joint represents elbow, which moves because rotation of the previous bone changes and rotates itself to change the position of wrist, the last joint in calculation. In this case, all we must do is to find out the location of the wrist in the environment. Unfortunately, there are infinite possibilities, where it can be. To be more specific, there is a whole circle of potential positions. Therefore, a little different solution was used.

The principle of this solution lies in the division of the axes into separate ones. Then the axis to zero value is reset. Simply, we put all the points into 2D space for each axis, except one. It is the axis that represents the bone rotation itself (in appropriate node) and it has no effect on the position of the next bone.

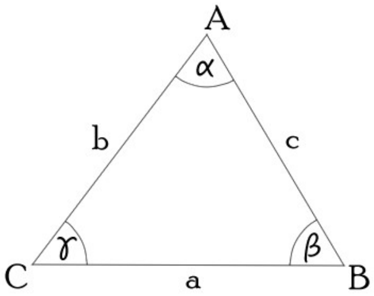


Fig. 5 Calculation of angles in 2D triangle

An ordinary 2D triangle is thus defined (Fig. 5). It is easy to calculate each angle in this triangle using the following formula:

$$\beta \quad = \arccos(\frac{a^2 + c^2 - b^2}{2ac}) \tag{1}$$

where:

- a - distance between the shoulder position (B) and position of the key (C);
- b - distance between the elbow position (A) and the wrist position (C);
- c - distance between the shoulder position (B) and the position of elbow (A).

The distance can be easily obtained using the *Three.js* function *Vector3.distanceTo (Vector3)*, where the vectors are the positions of the selected bones.

When the calculation is done in all three axes, the rotations of the appropriate triangles (meaning: the hand holds the key) are obtained, but the results need to be adapted to the actual rotations of the arm. Therefore, the same calculations are needed, but in a slightly different triangle, to find the differences between the actual and desired rotations. It was only necessary to change one variable: b - distance between elbow’s position (A) and position of key (C);

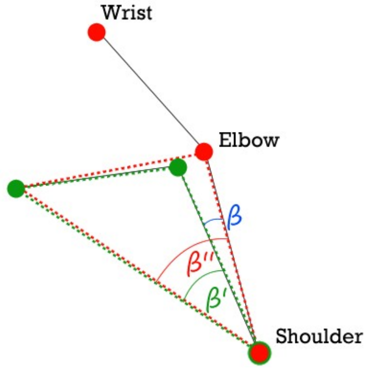


Fig. 6 Calculation of desired rotations for shoulder

As shown in Fig. 6, the green triangle is the desired one, which is already defined by the previous calculation. The red one is representing the actual positions of bones. These two triangles were used to find the angle at which the bone rotates. After inserting the results into the following formula (2), the rotations of the arm were obtained in the actual state:

$$\beta = \beta'' - \beta' \tag{2}$$

For calculations of the elbow rotation, a similar principle was used, with the same desired triangle. Another angle in it was calculated only. As it is shown in Fig. 7, it is needed to reduce the angle from 180° to get the desired angle.

$$\alpha = 180 - \arccos(\frac{a^2 + b^2 - c^2}{2ab}) \tag{3}$$

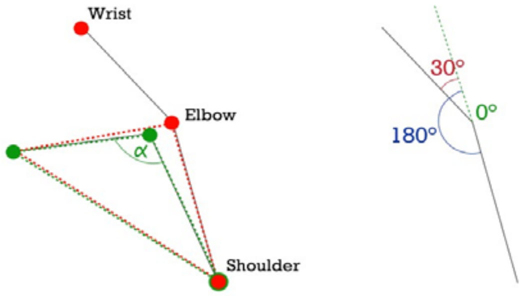


Fig. 7 Calculation of desired angle for elbow

Finally, it should be noted that this calculation calculates the wrist position equal to the key. But that was not our intention. In the real world, when a hand has to pick up a key, the target position of the hand is slightly above the key. To incorporate this, it was only needed to change the target position slightly higher in the *Y*-axis.

4.3. Picking up the target

The rotation of the fingers is set statically to the key lifting position. In this part, we focused mainly on the rotations and positions of the key. The key was added as a child to the thumb. This means that the world given to the descendant depends on the parent. If a parent is moved then a descendant would move with it at the same time. It is not necessary to constantly change position of all the descendants, their position and rotation is calculated from the position of the parent as the base of the object world. There are two ways to assign a descendant to a parent, as it is shown in Fig. 8:

- *Object3D.add(Object3D)* – its local position stays the same, so it moves to the position based on new parent
- *Object3D.attach(Object3D)* – its global position does not change and local position is calculated based on new parent, so it looks like nothing changes on the outside

The first method was implemented, because it was needed to calculate position of the key one way or another.

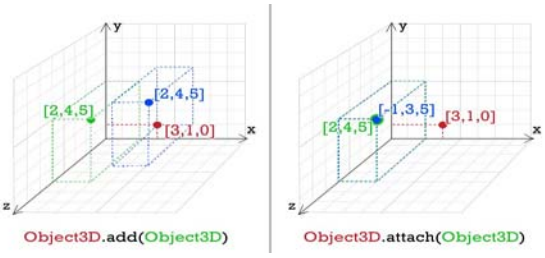


Fig. 8 Difference between *add()* and *attach()* methods

4.4. Movement to the lock

The same principle as in the next section was implemented for the movement to the lock and into the lock, just with different target. A rotation of the key aimed at the center of the lock for more realistic impression of animation was added.

4.5. Rotation around pivot

Fig. 9 shows the principle of rotation around a pivot other than the center of the object, using the following entities (implemented as cubes):

1. the center of gravity of the lock;
2. the center of gravity of the wrist transferred to the local space of the lock with the zero X-axis;
3. target position of the 2nd cube;
4. position of the 2nd cube transferred to the local space of the wrist with zero X-axis;
5. the center of gravity of the wrist before moving the wrist to the 4th cube position.

The 2nd and 3rd cubes are descendants of the 1st cube and the 4th cube is a descendant of the 5th cube.

From the first three cubes, it is possible to calculate the angle by which the hand should turn around the lock. The cubes 1 and 5 will rotate at the same angle simultaneously with the rotation of the wrist on the *Y*-axis. Their descendants will also rotate with them. The position of the wrist is moved to the new position of the 4th cube with each change. Since the key is still a descendant of the wrist, it will rotate with it.

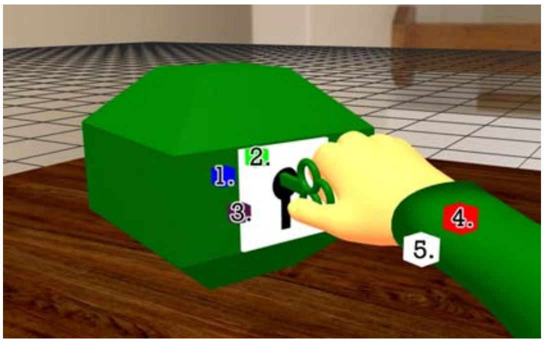


Fig. 9 Rotation according to the specified pivot

5. ANIMATION MANAGEMENT

Rotations at the end of each part of the animation were calculated in previous section. It was needed to get the bones and the objects to the target coordinates. A system of components in *A-Frame*, which contains the function called *tick()*, was used. This function is called in every frame, so the rotation of the bones is continuous in this method (not in method using clock measurements).

For performance optimization, we decided to calculate how many frames each animation part takes. Thus, it was possible to calculate the rotation for one frame. This value is based on frames per second in the currently used device. To detect rotation in one frame, rotation at the beginning and end of the animation part was needed.

$$oneFrameAngle = \frac{startedAngle - endedAngle}{fps} \qquad (4)$$

When one part of animation is done, the next one follows till the end of whole animation. An enumeration type was used in management of the animation part. This contains all parts and one variable that stores the actual one. Almost all parts of animations are calculated on this base, but the rotation in a lock is calculated in real time according to the methods in previous section, because just the position at the end of part is not enough.

6. SYSTEM TESTING AND EVALUATION

The system was tested (Fig. 10) with already completed parts of the application during implementation. Initially, the animations (fixed) were exported from Blender. These animations included all types of the desired movements, as well as at the end of the animation: gripping the cup, moving the cube to another location, and gripping the key and turning it in the lock. Due to the volume complexity, the system was therefore implemented with the support of the inverse kinematics, described above. Despite the reduction in the quality of the object movement, this step facilitates the completion of animations in the future. The animations do not need to be created separately and then imported into the application. The system has implemented functions to automatically complete the movement, i.e. touching/approaching virtual training objects if they are within range of the end effector after animation using the described inverse kinematics. Thus, only minimal changes are required also for individual grips when creating a new type of a grip. In addition, the animations are more versatile. The animation will be performed / completed even after moving the virtual training object, if it is within range of end effector.

The evaluated rehabilitation process, implemented in the virtual reality system for the upper limbs therapy proceeds as follows. After setting up the rehabilitation parameters, the patient is notified that the training will begin. For training purposes, it is essential that the patient sits at the table and perceives the 3D hand model (Fig. 10). After the training starts, several hand animations are performed. If the patient feels uncomfortable, the therapy process is stopped and then continues later. Given that each patient can perceive the 3D hand differently, we decided to extend the system by customized user profiles. These profiles store data on training, hand configuration in space, and its color.

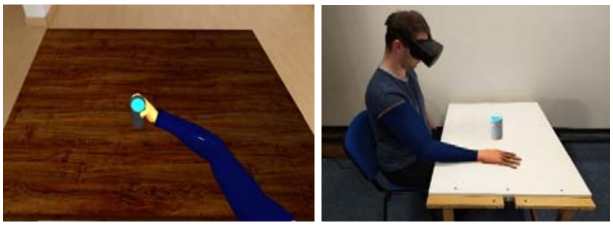
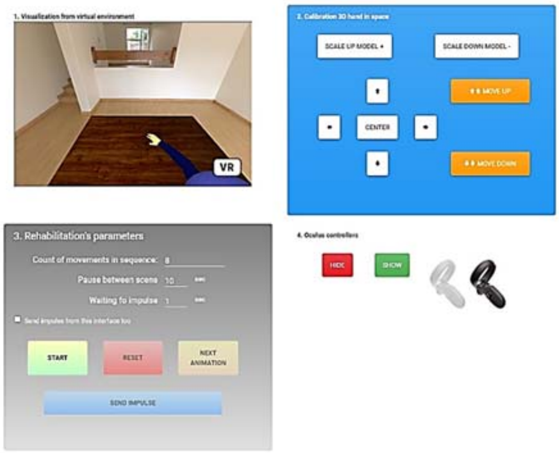


Fig. 10 Real therapy scene with a patient

Due to the pandemic situation, the system was tested only in the initial phase of its implementation. After redesign (the code and the animations), the application was checked only by therapists. The patients, who were involved in the project, were not allowed to leave their home and this experiment requires visits at least twice a week. Although the system is built on a collaborative online platform, the patient needs to have the equipment at home and needs someone to help him or her with it from the beginning. Therefore, it was more advantageous to go to the environment where the experiment was performed.

Along with the addition of the key and lock objects, the option to switch the animation to the next one should be added if the patient has a problem with the current movement difficulty. Another added requirement was for animations with copies of objects so that the patient could see what to imagine. However, the given implementation can confuse the patient and therefore we would like to give the possibility to turn off such animations in the future.
From the beginning, the application was set up in a collaborative environment so that the therapist could see what the patient saw and could communicate with him orf her if they are not in the same room. At the end of the first testing phase, a requirement was added to control the experiment from the therapist's virtual environment, not just at the beginning of the process (Fig. 11). Although the requirement has been met, due to the short implementation time, it is necessary to turn on the environment with parameters controlling the sequence. After alleviating the pandemic situation, it is planned to perform experiments with the given system together with the completion of the collaborative environment and greater possibilities of controlling the therapeutic process from the virtual interface of the therapist.



1. Visualization from virtual environment

2. Calibration 3D hand in space

3. Rehabilitation's parameters

Count of movements in sequence: 8

Pause between scene 10 sec

Waiting fo impulse 1 sec

□ Send impulse from this interface too

4. Oculus controllers

Fig. 11 Therapist interface

6.1. Therapist interface with parameters

The therapist's option is to go to the interface where the parameters of the rehabilitation process are set, which we can see in Fig. 11. The interface is divided into several parts:

- the visualization of the virtual environment in which the patient is located,
- the configuration of the 3D hand position,
- rehabilitation parameters and the necessary sequence control and
- a control to show or hide Oculus helmet controllers.

The therapist can set several parameters before starting the sequence:

- Number of movements in a sequence - determines the total number of animations in one sequence
- Hand display length - determines the time in seconds that the program waits for the impulse that starts the animation
- Pause length between views - between hand views, all objects except the hand are hidden for a given number of seconds
- Confirmation that the impulse can also be sent from the application environment, not only from the Robo-arm

The therapist can interrupt the sequence with the "*RESET*" button and change the settings at any time. The parameters can no longer change during the sequence. The impulse to start the animation is provided by Robo-Arm, but the therapist can also choose to send a command using the "*SEND IMPULSE*" button. The sequence starts after pressing the "*START*" button. After receiving a socket instructing to start the sequence, an animation is randomly selected. The therapist can also control the basic activities in a virtual environment.

7. CONCLUSIONS

As this paper has shown, virtual reality and related technologies affect a person's perception and thus his or her nervous system. It could be possible to positively influence the progress of neurorehabilitation, by providing a suitable virtual environment. This was also the main motivation of the presented work and it creates the preconditions for its development in the future.

The use of VR technologies mainly creates the patient's visual sensations. This improves the mental perception of movement in the brain and thus speeds up the rehabilitation process. The described solution significantly improves the naturalness of the limb movement and simplifies handling the limb model. In addition, this procedure is safe for the patient and, using collaborative VR environments, it allows the therapist's avatar to participate as in reality. Several patients can also take part in one therapy, regardless of their physical location. This also increases the potential socialization of patients and reduces the cost of therapy.

It is possible to use this system not only for the rehabilitation of the upper but also the lower limbs in the future. The advantage of this system is also the independence of technical means only by means of web-browsers. Another advantage is the possibility to perform some rehabilitation procedures for patients in the environment of their home without the intensive need to visit a therapist but under his supervision.

ACKNOWLEDGMENTS

This work has been supported by the APVV grant no. APVV-16-0202 “Enhancing cognition and motor rehabilitation using mixed reality”.

REFERENCES

[1] Clinica orthopedic: What is rehabilitation and why it is important for a patient, cit 04/2021, url: https://clinicaorthopedica.sk/co-je-rehabilitacia-a-preco-je-prepacienta-dolezita.

[2] BEKELE, M. K. – PIERDICCA, R. – FRONTONI, E. – MALINVERNI, E. S. – GAIN, J.: “A survey of augmented, virtual, and mixed reality for cultural heritage, ”Journal on Computing and Cultural Heritage (JOCCH), vol. 11, no. 2,pp. 1–36, 2018.

[3] R´IOS, A. P. – CALLAGHAN, V. – GARDNER, M. – ALHADDAD, M. J.: “Using mixed-reality to develop smart environments,” pp. 182–189, 2014.

[4] KHAN, O. – AHMED, I. – COTTINGHAM, J. – RAHHAL, M. – ARVANITIS, T. N. et al.: Timing and correction of stepping movements with a virtual reality avatar., 2020, PLOS ONE 15(2): e0229641. https://doi.org/10.1371/journal.pone.0229641

[5] SAPKAROSKI, D. – BAIRD, M. – MCINERNEY, J. – DIMMOCK, M. R.: “The implementation of a haptic feedback virtual reality simulation clinic with dynamic patient interaction and communication for medical imaging students, ”Journal of medical radiation sciences, vol. 65, no. 3, pp. 218–225, 2018.

[6] LOPES, D. S. – JORGE, J. A.: “Extending medical interfaces towards virtual reality and augmented reality, ”Annals of Medicine, vol. 51, no. sup1,pp. 29–29, 2019.

[7] NISSLER, C. – NOWAK, M. – CONNAN, M. – B˙UTTNER, S. – VOGEL, J. – KOSSYK, I. – C.M´ARTON, Z. – CASTELLINI, C.: “Vita—an everyday virtual reality setup for prosthetics and upper-limb rehabilitation, ”Journal of neural engineering, vol. 16, no. 2, p. 026039, 2019.

[8] CHEN, D. – LIU, H. – REN, Z.: “Application of wearable device htc vive inupper limb rehabilitation training,” pp. 1460–1464, 2018.

[9] WU, X. – LIU, H. – ZHANG, J. – CHEN, W.: “Virtual reality training system for upper limb rehabilitation,” pp. 1969–1974, 2019.

[10] ELOR, A. – TEODORESCU, M. – KURNIAWAN, S.: “Project star catcher: A novel immersive virtual reality experience for upper limb rehabilitation,” ACM Transactions on Accessible Computing (TACCESS), vol. 11, no. 4, pp. 1–25, 2018.

[11] BECKER, S. – CHAPLE, M. – FREESE, T. – HAGLE, H. – HENRY, M. – KOUTSENOK, I. – KROM, L. – MARTIN, R. – MOLFENTER, T. –

POWELL, K. et al.: “Virtual reality forbehavioral health workforce development in the era of covid-19,”Journalof Substance Abuse Treatment, p. 108157, 2020.

[12] PEDRAM, S. – PALMISANO, S. – PEREZ, P. – MURSIC, R. – FARRELLY, M.: “Examiningthe potential of virtual reality to deliver remote rehabilitation,”Computersin Human Behavior, vol. 105, p. 106223, 2020.

[13] SEO, D. – YOO, B. – KO, H.: “Webizing collaborative interaction space forcross reality with various human interface devices,” pp. 1–8, 2018.

[14] GIL-GÓMEZ, J. A. – LOZANO-QUILIS, J. A. – GIL-GÓMEZ, H.: Virtual Reality System for Multiple Sclerosis Rehabilitation using KINECT. cit.: 05/2021, url: http://rehab-workshop.org/2013/papers/136696042743759.pdf.

[15] GestureTek. “Irex is gesturetek health’s flagship product for the healthcare arena.” Cit.: 06/2021, url: http://www.gesturetekhealth.com/ products/irex.

[16] CIMROVÁ, B. – FARKAŠ, I. – ROSIPAL, R.: "An usage of brain-computer interface during neurorehabilitation: review of actual state." Kognícia a umelý život 14 (2014).

[17] ROSIPAL, R.: Brain-computer interface with robot-assisted training for rehabilitation (BCI-RAS). url: http://aiolos.um.savba.sk/~roman/rrLab/ projects.html.

[18] HUDÁK, M. – SIVÝ, M.: Web-based collaborative virtual environments to support cross-platform access, In: Poster 2019 International student scientific conference, Prague, 2019, pp.178-182.

[19] DEMERS, D. – KREUTZ-DELGADO, K.: “4 - inverse kinematics of dextrous manipulators,” in Neural Systems for Robotics, O. Omidvar and P. van der Smagt, Eds. Boston: Academic Press, 1997, pp. 75–116. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780080925097500087

[20] BOWER, T.: Inverse kinematics. [Online]. Available: http://faculty.salina.k-state.edu/tim/robotics sg/Arm robots/ inverseKin.html

Received July 1, 2021 , accepted November 15, 2021

BIOGRAPHIES

Branislav Sobota was born n 1967. In 1990 he graduated (MSc.) with honors at the Department of Computers and Informatics of the FEEI at Technical University in Košice. He obtained his PhD in 1999 and habilitation thesis in 2008 in the field of virtual reality and computer graphics. He is currently working as an associate professor at the Department of Computers and Informatics at the Technical University of Košice, Slovakia. His scientific research is focusing on computer graphics, parallel computing and especially virtual reality and related technologies.

Štefan Korečko was born in 1978. In 2006, he defended his PhD thesis at the department of Computers and Informatics of the Faculty of Electrical Engineering and Informatics at Technical University (DCI FEEI TU) in Košice. The title of his thesis was “Integration of Petri Nets and B-Method for the mFDT Environment”. Since 2004 he is working as an assistant professor at the DCI FEEI TU in Košice. His scientific research is primarily focused on formal methods, Petri nets and B-Method in particular, their integration and utilization in modelling and simulation. He also deals with virtual and augmented reality technologies.

Sára Javorková was born in 1998. In 2021, she graduated (MSc) at the Department of Computers and Informatics of the Faculty of Electrical Engineering and Informatics at Technical University of Košice. Her work is focused on web technologies, computer graphics and virtual reality.

Marián Hudák was born in 1992. In 2017 he graduated (MSc) and in 2021 he graduated (PhD.) at the Department of Computers and Informatics of the Faculty of Electrical Engineering and Informatics at Technical University of Košice. He defended his doctoral thesis in the field of Informatics. Currently, he is an employee in the Siemens Healthcare s.r.o. His research is focused on web-based virtual reality interfaces, cross-platform XR development, and collaborative immersive technologies