Biomechanics Sensor Node for Virtual Reality: A Wearable Device Applied to Gait Recovery for Neurofunctional Rehabilitation

Alexandre Fonseca Brandão[1,2,3], Diego Roberto Colombo Dias[2,4(✉)], Sávyo Toledo Machado Reis[4], Clovis Magri Cabreira[7], Maria Cecilia Moraes Frade[5], Thomas Beltrame[2,5], Marcelo de Paiva Guimarães[5,6], and Gabriela Castellano[1,2]

[1] Institute of Physics Gleb Wataghin, University of Campinas, Campinas, SP, Brazil
{abrandao,gabriela}@ifi.unicamp.br
[2] Brazilian Institute of Neuroscience and Neurotechnology, University of Campinas, Campinas, SP 13083887, Brazil
beltramethomas@gmail.com
[3] Institute for Bioengineering of Catalonia - IBEC, Barcelona, Spain
[4] Universidade Federal de São João del-Rei, São João del-Rei, MG, Brazil
diegodias@ufsj.edu.br, savyomachado@gmail.com
[5] Universidade Federal de São Paulo, São Paulo, SP, Brazil
mariaceciliafrade@gmail.com, marcelodepaiva@gmail.com
[6] Centro Universitário Campo Limpo Paulista - UNIFACCAMP, Campo Limpo Paulista, Brazil
[7] Center for Research and Development in Telecommunications (CPqD), Campinas, SP, Brazil
cmagri@cpqd.com.br

Abstract. In several segments of the health areas, sensing has become a trend. Sensors allow data quantification for use in decision making or even to predict the clinical evolution of a given treatment, such as in rehabilitation therapies to restore patients' motor and cognitive functions. This paper presents the Biomechanics Sensor Node (BSN), composed of an inertial measurement unit (IMU), developed to infer input information and control virtual environments. We also present a software solution, which integrates the BSN data with Unity Editor, one of the most used game engine nowadays. This asset allows Unity-developed virtual reality applications to use BSN a secure interaction device. Thus, during rehabilitation sessions, the patient receives visual stimuli from the virtual environment, controlled by the BSN device, while the therapist has access to the information about the movements performed in the therapy.

Keywords: Wearable sensors · Stationary gait · Virtual reality · Human computer-interaction · Neurofunctional rehabilitation

© Springer Nature Switzerland AG 2020
O. Gervasi et al. (Eds.): ICCSA 2020, LNCS 12255, pp. 757–770, 2020.
https://doi.org/10.1007/978-3-030-58820-5_54

1 Introduction

With the advancement of technology, virtual reality (VR) has been growing and becoming ever more present. Nowadays, we are using VR in areas as education, health therapies, and entertainment industries. The term "virtual reality" was coined in 1989 by Jaron Lanier. VR technologies aim to recreate the sensation of reality to the subject, leading him/her to experience the interaction with a virtual environment (VE) as a momentary reality [25].

A natural way to interact with this virtual environment is required for a completely immersive experience. There must be a way of interacting with it, and in most cases, a navigation device is responsible for mediating this communication with VE.

Nowadays, devices such as Kinect allow the user to move through a VE [37]. Such devices are still quite limited in terms of user freedom, precision, and complexity of movement control. These restrictions give rise to a scenario favorable to the creation of new solutions that aim to optimize the immersion of the user. Other examples of devices are the wearable sensors fixed in different parts of the user's body, which can capture patterns of movement. A problem with this sort of device is that they present excessive battery consumption, requiring continuous recharging, which can become a negative point in the user experience. Low energy technologies have been developed to mitigate this problem, resulting in so-called low energy devices.

With the virtual environment integrated with wearable devices, it is possible to develop solutions for a wide range of healthcare applications. These include the prevention of muscular atrophy (sarcopenia), neurological recovery from diseases such as stroke, Parkinson's [20], Alzheimer's [8], and respiratory rehabilitation, such as Chronic Obstructive Pulmonary Disease (COPD) [33]. Among vascular diseases, stroke represents the leading cause of long-term disability. The increasing proportion of survivors from this disease is associated with an increase in patients who persist with neurological deficit. Indeed, more than half of the survivors remain with a severe disability affecting functional independence in daily life activities [17,24].

For this reason, rehabilitation programs based on VR [25] have been highlighted as an alternative and complementary therapy for motor recovery [15,18,19,23,30]. VR can stimulate various sensory systems of the human body, including the visual and auditory systems, which facilitate the input and output of information to the brain. Therefore, VR systems can be used in conjunction with other therapeutic interventions to increase the complexity of tasks during the rehabilitation process [1,7,16,27,28,32]. This complement is particularly important when considering the potential increase in the number of patients with stroke and neurodegenerative diseases in the future.

Every VR interface must be composed of immersion and interaction devices. Regarding interaction devices, we can highlight the use of inertial devices as a means of body tracking, which is not new, but still lacks specific solutions for healthcare. Foxlin [11], a few years ago, came up with a walking tracking solution called NavShoe. The solution consisted of an inertial sensor attached to the

shoelace of one of the user's feet, connected to a PDA (wireless). Tregillus and Folmer [29] presented a walking solution in VE using the smartphone's sensors. Wittman et al. [35] studied the feasibility of an unsupervised arm therapy for self-directed rehabilitation therapy in patients' homes using an inertial measurement unit (IMU)-based VR system (ArmeoSenso) in their homes for six weeks. These solutions have been using inertial sensors for tracking virtual environments. However, in most of them, the prototypes are focused on one group of members (upper or lower).

Given these considerations, this paper presents a Biomechanics Sensor Node (BSN), which is a wearable device developed by our group. BSN contains gyro, accelerometer, and compass sensors, which are combined to associate user movement with input commands to control virtual reality applications. We also present a Unity Asset which integrates the BSN data with VR applications. We also present a Unity Asset[1] which integrates the BSN data with VR applications.

This article is organized as follows: Sect. 2 presents the material and methods used to create the BSN device. Section 3 shows our case study results. Section 4 presents the discussions, and lastly, Sect. 5 presents the conclusions.

2 Materials and Methods

For the construction of the BSN device, previous results [9,12], also developed by our group, were used. These are the GestureMaps (non-immersive) application, presented in Sect. 2.1, and the e-Street (immersive) application, presented in Sect. 2.2. Finally, in Sect. 2.3, we present the main contribution of this paper, the actual development of the BSN device.

2.1 GestureMaps

GestureMaps is a mixed VR application [5], where the user can navigate the virtual maps of Google Street View using stationary gait movements. This movement is identified through a gesture recognition sensor, Kinect type, which translates the stationary gait into an input for the VR system, allowing user navigation (output) through the virtual environment. In this case, a hip and knee flexion movement is required from the user. A distance of at least ten centimeters from the foot relative to the floor is required to ensure displacement in the virtual environment. Figure 1 shows an example of the use of GestureMaps [12].

[1] The Asset is an item (e.g., source code, a 3D model, an audio file or an image) that facilitates to create Unity applications. An asset can be used to build diverse applications. Unity is a cross-platform engine that is used to develop games on multiple platforms.

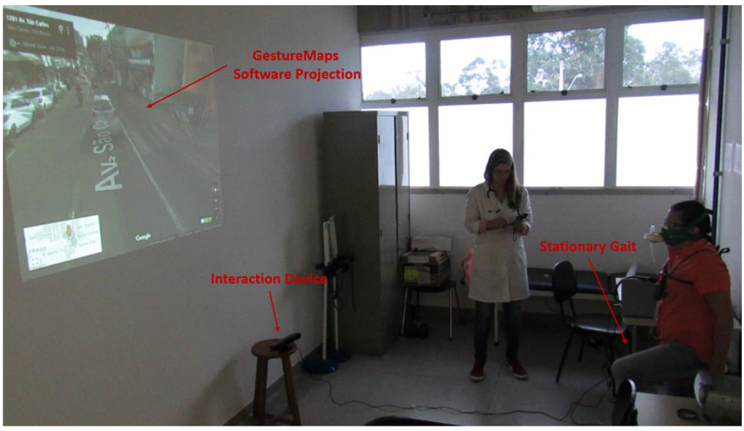


Fig. 1. Example of use of GestureMaps. The subject uses stationary gait to navigate the virtual environment.

2.2 e-Street and Interaction Device: Ultrasound Sensor and Arduino-Based

A virtual city (called e-Street) was created to develop an immersive version of GestureMaps in a Unity 3D environment to simulate navigation, street crossing, and spatial orientation. This virtual urban environment reproduces traffic situations, with autonomous cars that circulate in the e-Street.

An interaction device was constructed from ultrasound sensors and an Arduino to translate the stationary gait movements into input for the e-Street software. From actions that reach displacements of three centimeters (or more) of the ultrasound sensors, it is possible to start navigation in the urban virtual environment by the user [9]. An Arduino (UNO) microcontroller, two sonars (HC-SR04), and one Bluetooth module (HC-05) were used to create the BSN wearable device. Figure 2 represents the e-Street software interface and interaction device.

2.3 Biomechanics Sensor Node - BSN

The BSN device was developed due to the imprecision and clumsiness of the ultrasound sensors used for interaction with the e-Street environment. To interact more precisely with a virtual environment, an ergonomic bracelet that could be attached to the wrist or ankle was designed to be used to track the user's movements.

From inertial sensors, the connection between the bracelet (BSN) and the VR software (e-Street - Unity 3D) is possible. The Unity 3D (motor engine) proved very suitable for the development of the solution (e-Street + BSN). Besides, this game engine also has support for exporting the applications to the mobile environment, which is essential since smartphones can be used with the BSN.



Fig. 2. e-Street interface (upper right corner) and interaction device (lower right corner).

Next, the development of the asset was carried out, bringing together all the functions implemented in the BSN. In this way, the communication module and the BSN sensors were configured correctly, according to the information pertinent to the communication protocol (Table 1).

Table 1. Gestural interaction devices comparative.

Gestural interaction devices				
Device	Sensor	Gesture recognition	Movement required	Controlled software
Kinect	Infrared	Computer vision	Stationary gait	GestureMaps
Arduino-based	Ultrasonic	Spatial displacement on the Y axis	Foot up and down (Y-axis)	e-Street
BSN	IMU	Three-dimensional spatial displacement	Stationary gait	e-Street

The BSN device is also prepared to send the raw data to the Internet of Things - IoT DoJot platform [10], which is a Brazilian open-source platform (developed by CPqD – https://www.cpqd.com.br/). These quantitative data can be compared with qualitative data, relative to the clinical evolution of the

patient, and thus be related to the proposed motor and neurofunctional rehabilitation treatment. Figure 3 presents the BSN connection with the IoT platform. The flowchart shows the communication between the BSN device, the e-Street software (Unity 3D), and the DoJoT platform (IoT).



Fig. 3. Communication between BSN, Android/Unity and Dojot.

3 Results

3.1 BSN Communication and Data Collection Service

The communication occurs through four phases: scanning of devices compatible with BLE technology; BSN identification in the list of found devices and connection requests; sending of calibration command with BSN, and reading the BSN sensor data.

The service responsible for collecting BSN data is named Data Collection Service. In this category of services, it is possible to find the step information and battery status, which is essential for future applications.

Regarding the available services for reading the information that is Notify type, they send information to the recipients only when the data is updated. Thus, it is possible to subscribe to these services to get constant data feedback. This subscription becomes interesting in the process of tracking members since applications can create a history of user movement in space.

For the first step execution, we use a scan function through a Unity script, which verifies the presence of nearby connected devices that support the BLE protocol. The device must be switched on and in standby mode, waiting for new connections. The identification of this Bluetooth connection occurs with the blue color of the light-emitting diode (LED), which uses the RGB (red, green, and blue) system.

In the second step, the identification (ID) of the device of interest was made using its name or media access control (MAC) address. Then your ID is represented on the bracelet by a white LED. Thus, an information exchange with the armband is allowed. If the pairing process occurs without problems, the LED will turn green.

In the third step, the calibration command is sent to the BSN, a BLE package consisting of two elements: service and characteristic. If no shipping problems

occur, the LED turns orange, returning to the green color again after the calibration process. Finally, in the fourth step, the information available in the BSN is read, getting for each data a Service UUID, and a Characteristic UUID.

3.2 BSN Asset

Knowing that the *Asset* to be developed has all the bracelet manipulation features, it includes both functions, the configuration service, and the data collection service. The functions that compose the *Asset* are: FindBSN, ConnectBSN, CalibrateBSN, ReceiveQuaternions, ReceiveRawData, ReceiveEuler, ReceiveRotationMatrix, ReceiveCompass, ReceiveGravityVector, ReceiveBatteryStatus, and ReceiveSteps.

Since the developed solution must allow the connection between a mobile device and BSN, we have proposed some functions for this purpose. FindBSN and ConnectBSN are responsible for this task; they are described by Algorithm 1 and Algorithm 2, respectively. The first one tracks all devices that support BLE technology and identifies the BSN among them. The second makes the connection itself with the armband, where after the function call, the device is available for future instructions.

Algorithm 1: FindBSN function pseudocode

```
   Result: Find BSN
1  begin
2      FoundDeviceList.DeviceAddressList ← List¡DeviceObject¿();
3      initializeBLE();
4      discoveredDevices ← scanForPeripherals();
5      foreach DeviceObject d ∈ discoveredDevices do
6          if d.name = "BSN" then
7              FoundDeviceList.DeviceAddressList.Add(d);
```

Algorithm 2: ConnectBSN function pseudocode

```
   Result: BSN connection established
1  begin
2      foreach DeviceObject d ∈ FoundDeviceList.DeviceAddressList do
3          bsn ← d;
4          connectToPeripheral(bsn);
```

3.3 BSN Calibration

With the connection between the BSN and a mobile device (smartphone), it is possible to perform the calibration function. Thus, through the function CalibrateBSN described in Algorithm 3, the smartphone sends the calibration command to the BSN, a set of bytes using service and characteristic corresponding to the calibration command.

Algorithm 3: CalibrateBSN function pseudocode

```
Result: Starting BSN calibration
1 begin
2     serviceWriteUUID ← "XXXX";
3     characteristicCalibrationUUID ← "XXXX";
4     calibrateBytes ← { bytes };
5     sendBytesBSN(calibrateBytes, bsnDevice.Address, serviceWriteUUID,
      characteristicCalibrationUUID);
```

3.4 BSN Data Reading

BSN data reading occurs by a service containing a standard UUID, which is used by all the read functions present in the Asset. Thus, to consult the information present in the armband, the Characteristic UUID must be equal to the information UUID that one wishes to obtain.

Algorithm 4 presents the ReceiveSteps function. The step data are obtained using the Characteristic UUID equivalent to the step information. The result of the ReceiveSteps function is a set of bytes, converted to the uint16_t format an unsigned integer of 16 bits size. With the bytes duly converted, we have the number of steps counted by the BSN.

Algorithm 4: ReceiveSteps function pseudocode

```
Output: steps
1 begin
2     serviceReadUUID ← "XXXX";
3     characteristicReadStepUUID ← "XXXX";
4     stepBytes ←
      subscribeCharacteristicWithDeviceAddress(bsnDevice.Address,
      serviceReadUUID, characteristicReadStepUUID);
5     steps ← convertToUInt16(stepBytes)
```

Algorithm 5 presents the pseudocode of the ReceiveQuaternions function. The registration function results in a set of bytes that contains the components W, X, Y, and Z of the quaternion vector. Thus, the byte set is converted to an int32_t data pattern to obtain the actual data of the four components. Finally, the function groups all the components and returns Q, the quaternion vector obtained from the BSN.

Algorithm 5: ReceiveQuaternions function pseudocode

```
   Output: Q
1  begin
2      serviceReadUUID ← "XXXX";
3      characteristicReadQuaternionsUUID ← "XXXX";
4      qByte ← subscribeCharacteristicWithDeviceAddress(bsnDevice.Address,
       serviceReadUUID, characteristicReadQuaternionsUUID);
5      qw, qx, qy, qz ← convertToInt32(quaternionsByte);
6      Q ← {qW,qx,qy,qz};
```

We create the ReceiveRawData function, to obtain the raw data of the BSN device. Algorithm 6 presents the use of the Characteristic UUID equivalent to the data information sensor. Thus, we convert the resulting bytes (that contains the X, Y, and Z components of the accelerometer, gyroscope and compass sensors) to the int16_t standard data, the default used by the BSN for this data set. Finally, the function groups the elements of each sensor into an R vector, returning a composite list of the vectors of each sensor.

Algorithm 6: ReceiveRawData function pseudocode

```
   Output: R
1  begin
2      serviceReadUUID ← "XXXX";
3      characteristicReadRawDataUUID ← "XXXX";
4      rawDataByte ←
       subscribeCharacteristicWithDeviceAddress(bsnDevice.Address,
       serviceReadUUID, characteristicReadRawDataUUID);
5      acelX, acelY, acelZ, gyroX, gyroY, gyroZ, compX, compY, compZ ←
       convertToInt16(rawDataByte);
6      accelerometer ← {acelX, acelY, acelZ};
7      gyroscope ← {gyroX, gyroY, gyroZ};
8      compass ← {compX, compY, compZ};
9      R ← {accelerometer, gyroscope, compass };
```

To illustrate the behavior of the Asset developed, Fig. 4, we describe the process of obtaining the steps given by the user, showing all the actions (steps) performed as input.

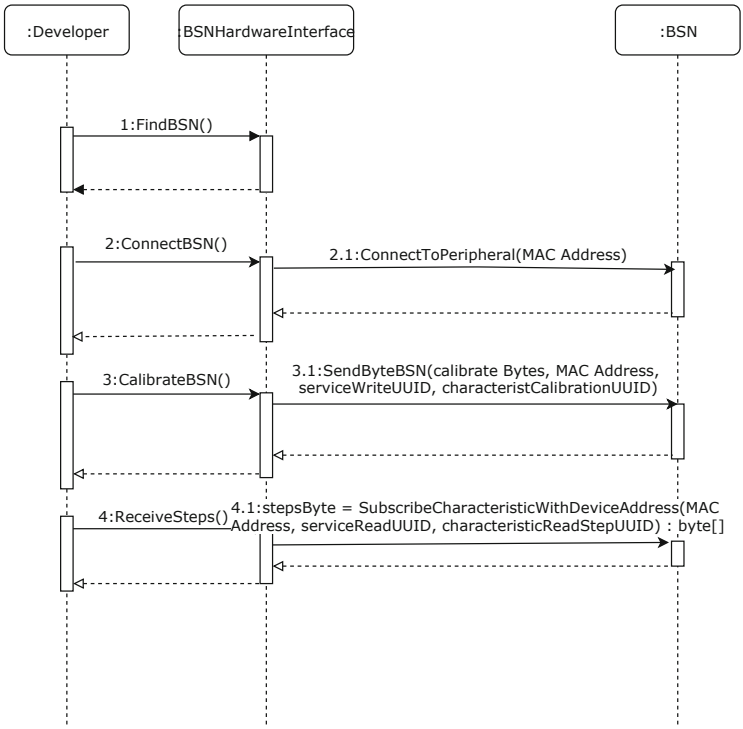


Fig. 4. BSN sequence diagram.

The first command (step 1 of the diagram) is responsible for identifying the BSN among nearby devices that support BLE technology. In this step, the BSN has the LED on blue.

After the second method call (step 2 of the diagram), the bracelet emits a pink light, signaling that there has been a connection request. In this step, the device performs the process of matching and applying filters to the sensors.

Once the pairing process is complete, the BSN LED turns green. At this stage, the bracelet enters in standby mode, waiting for commands from the paired smartphone. In this way, we represent any input sent with the calibration command.

After sending the calibration command (step 3 of the diagram) to the device, the bracelet emits an orange light, which corresponds to the calibration process. At this point, it is necessary to perform three equal movements, with a pause of two seconds between them, so that the bracelet registers the movement correctly. After the calibration process occurs, the LED turns green again.

Finally, the method responsible for receiving the steps (step 4 of the diagram) counts the steps. With the calibrated BSN, we add the value one to the step counter with each step complete.

4 Discussion

Wearables are becoming very popular and accessible due to reduced costs [14], allowing patients to obtain personal biological data daily. The smaller sensor sizes, associated with low power wireless communication, are allowing users/patients to use such devices imperceptibly, with data acquisition performed throughout the day.

The use of this type of technology is finding more and more applications in health sciences, which will lead us to a deeper understanding of disease development through the analysis of the collected data [2]. Biological data obtained from wearables has the potential to allow providing more objective and effective treatments for patients with neurofunctional disorders.

With the advancement in the use of miniature sensors and mobile devices and with the high computational power of data processing, studies with wearable devices in the area of rehabilitation are becoming ever more feasible and have great potential to motivate patients in the process of physical recovery [3,36] and be used in the neuroscience field [6].

Regarding the present study, because the BSN has the form of a bracelet, this device can become part of the patient's dressing room. Thus, it will not cause discomfort and will still collect data in real-time during treatment due to remote BLE communication. Other studies have presented interaction devices, based on Inertial Measurement Unity (IMU), aimed at health applications. Such as the instrumented trail-making task (iTMT) developed for the prevention of fragility [38]; and the ArmeoSenso, which connects with the virtual environment and allows the control of a virtual arm, offering elbow flexion and extension stimuli [35]. These can be used for monitoring rehabilitation patients in hospitals and at home [13].

Our BSN device uses an IMU (such as gesture recognition sensor) to calibrate a specific motion and convert it into input for virtual reality systems. Other commercial IMU-based motion analysis systems, such as Werium [34] and SWORD health [26], serve as inspiration for the next stages of development of our BSN device. The idea is to expand its functionality to gather a range of motion information and integrate it with kinematic analysis software such as RehabGesture [4], also developed by our team.

From the identification of the movements through the BSN device, it was possible to integrate it with the e-Street VR software allowing the user to navigate in the virtual city. The outcome of this integration is expected to be an e-Health solution, with applications in several fields such as rehabilitation, motor learning, and fall prevention.

Due to the motor requirements of the lower limb muscles and the cognitive requirements needed for spatial orientation during software control, this solution finds application in fall prevention therapies in the elderly and the recovery of gait movements after stroke.

However, studies have also pointed out the need to increase the number of multicenter and randomized clinical trials with a representative sample of

the population, to highlight the full potential of VR as a complement to rehabilitation therapy and to consolidate the efficacy of treatment in medium and long-term [21,22,31].

5 Conclusions

In this article, we present the proposal and design of a BSN device by the lack of tools that can track the upper and lower limbs and the results of our initial solutions using Arduino. Our BSN allows up to 8 devices to be scanned at the same time, enabling multiple body parts to be tracked, lower, and upper limbs.

We also presented an Asset, which is a set of functionalities for the Unity game engine. It facilitates the development of virtual reality applications integrated with our BSN device. Using this solution, the developers can quickly perform actions such as connection, calibration, and capture of raw data.

The Unity development solution is also a significant contribution since the interface between developer and device should be simple and easy. With Unity Asset built, you can connect, calibrate, capture raw data, and allow sensor fusion and motion composition. The result of our BSN device is a universal and customizable tool, providing calibration of steady-gait-related movements as well as their interaction with VR environments.

The next step is the use of the BSN device for control of an avatar to simulate Activities of Daily Living (ADLs), through virtual limbs and objects control, for interaction with other individuals/patients in a multiuser virtual environment, or for sports simulation for improving skills and techniques. The limitations of the study refer to the need for testing with neurological patients in rehabilitation clinics and hospitals, expected as future work.

Funding. FAPESP (Sao Paulo Research Foundation, Brazil) grant number 2015/03695-5 (grant related to author: A.F.B.). The EMBRAPII (Brazilian Agency for Research and Industrial Innovation) and SEBRAE made a financial contribution to the development of BSN.

References

1. Aşkın, A., Atar, E., Koçyiğit, H., Tosun, A.: Effects of kinect-based virtual reality game training on upper extremity motor recovery in chronic stroke. Somatosens. Mot. Res. **35**(1), 25–32 (2018)
2. Beltrame, T., Amelard, R., Wong, A., Hughson, R.L.: Extracting aerobic system dynamics during unsupervised activities of daily living using wearable sensor machine learning models. J. Appl. Physiol. **124**(2), 473–481 (2017)
3. Bernhard, F.P., Sartor, J., Bettecken, K., Hobert, M.A., Arnold, C., Weber, Y.G., Poli, S., Margraf, N.G., Schlenstedt, C., Hansen, C., et al.: Wearables for gait and balance assessment in the neurological ward-study design and first results of a prospective cross-sectional feasibility study with 384 inpatients. BMC Neurol. **18**(1), 114 (2018)

4. Brandao, A.F., Dias, D.R., Castellano, G., Parizotto, N.A., Trevelin, L.C.: Rehabgesture: an alternative tool for measuring human movement. Telemedicine e-Health **22**(7), 584–589 (2016)
5. Brandão, A.F., Dias, D.R.C., Guimarães, M.P., Trevelin, L.C., Parizotto, N.A., Castellano, G.: Gesturecollection for motor and cognitive stimuli: virtual reality and e-health prospects. J. Health Inform. **10**(1), 9–16 (2018)
6. Byrom, B., McCarthy, M., Schueler, P., Muehlhausen, W.: Brain monitoring devices in neuroscience clinical research: the potential of remote monitoring using sensors, wearables, and mobile devices. Clin. Pharmacol. Ther. **104**, 59–71 (2018)
7. Cameirao, M.S., i Badia, S.B., Duarte, E., Frisoli, A., Verschure, P.F.: The combined impact of virtual reality neurorehabilitation and its interfaces on upper extremity functional recovery in patients with chronic stroke. Stroke **43**(10), 2720–2728 (2012)
8. Cushman, L.A., Stein, K., Duffy, C.J.: Detecting navigational deficits in cognitive aging and alzheimer disease using virtual reality. Neurology **71**(12), 888–895 (2008)
9. Dias, D.R.C., Alvarenga, I.C., Guimarães, M.P., Trevelin, L.C., Castellano, G., Brandão, A.F.: eStreet: virtual reality and wearable devices applied to rehabilitation. In: Gervasi, O., Murgante, B., Misra, S., Stankova, E., Torre, C.M., Rocha, A.M.A.C., Taniar, D., Apduhan, B.O., Tarantino, E., Ryu, Y. (eds.) ICCSA 2018. LNCS, vol. 10963, pp. 775–789. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-95171-3_60
10. dojot: dojot documentation (2019). https://dojotdocs.readthedocs.io/en/latest/. Accessed Dec 2019
11. Foxlin, E.: Pedestrian tracking with shoe-mounted inertial sensors. IEEE Comput. Graph. Appl. **6**, 38–46 (2005)
12. Frade, M.C., dos Reis, I.M., Basso-Vanelli, R.P., Brandão, A.F., Jamami, M.: Reproducibility and validity of the 6-minute stationary walk test associated with virtual reality in subjects with COPD. Respiratory care, pp. respcare-06237 (2019). https://doi.org/10.4187/respcare.06237
13. Hadjidj, A., Bouabdallah, A., Challal, Y.: Rehabilitation supervision using wireless sensor networks. In: 2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), pp. 1–3. IEEE (2011)
14. Hsu, Y.L., Yang, S.C., Chang, H.C., Lai, H.C.: Human daily and sport activity recognition using a wearable inertial sensor network. IEEE Access **6**, 31715–31728 (2018)
15. Laver, K.E., George, S., Thomas, S., Deutsch, J.E., Crotty, M.: Virtual reality for stroke rehabilitation. Cochrane Database Syst. Rev. (2) (2015)
16. Lloréns, R., Noé, E., Colomer, C., Alcañiz, M.: Effectiveness, usability, and cost-benefit of a virtual reality-based telerehabilitation program for balance recovery after stroke: a randomized controlled trial. Arch. Phys. Med. Rehabil. **96**(3), 418–425 (2015)
17. Nichols-Larsen, D.S., Clark, P., Zeringue, A., Greenspan, A., Blanton, S.: Factors influencing stroke survivors' quality of life during subacute recovery. Stroke **36**(7), 1480–1484 (2005)
18. Piron, L., et al.: Exercises for paretic upper limb after stroke: a combined virtual-reality and telemedicine approach. J. Rehabil. Med. **41**(12), 1016–1020 (2009)
19. Rose, F.D., Brooks, B.M., Rizzo, A.A.: Virtual reality in brain damage rehabilitation. Cyberpsychol. Behav. **8**(3), 241–262 (2005)

20. dos Santos Mendes, F.A., et al.: Motor learning, retention and transfer after virtual-reality-based training in parkinson's disease-effect of motor and cognitive demands of games: a longitudinal, controlled clinical study. Physiotherapy **98**(3), 217–223 (2012)
21. Saposnik, G., et al.: Efficacy and safety of non-immersive virtual reality exercising in stroke rehabilitation (EVREST): a randomised, multicentre, single-blind, controlled trial. Lancet Neurol. **15**(10), 1019–1027 (2016)
22. Schuster-Amft, C., et al.: Effect of a four-week virtual reality-based training versus conventional therapy on upper limb motor function after stroke: a multicenter parallel group randomized trial. PLoS ONE **13**(10), e0204455 (2018)
23. Shin, J.H., Ryu, H., Jang, S.H.: A task-specific interactive game-based virtual reality rehabilitation system for patients with stroke: a usability test and two clinical experiments. J. Neuroeng. Rehab. **11**(1), 32 (2014)
24. Statements, A.S.: Heart disease and stroke statistics 2013 update. Circulation **127**(1), e6 (2013)
25. Steuer, J.: Defining virtual reality: dimensions determining telepresence. J. Commun. **42**(4), 73–93 (1992)
26. SwordHealth: Reinventing Physical Threapy (2019). https://swordhealth.com/. Accessed Feb 2019
27. Tieri, G., Morone, G., Paolucci, S., Iosa, M.: Virtual reality in cognitive and motor rehabilitation: facts, fiction and fallacies. Expert Rev. Med. Dev. **15**(2), 107–117 (2018)
28. Tobler-Ammann, B.C., et al.: Exergames encouraging exploration of hemineglected space in stroke patients with visuospatial neglect: a feasibility study. JMIR Serious Games **5**(3), e17 (2017)
29. Tregillus, S., Folmer, E.: VR-step: walking-in-place using inertial sensing for hands free navigation in mobile VR environments. In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, pp. 1250–1255. ACM (2016)
30. Turolla, A., et al.: Virtual reality for the rehabilitation of the upper limb motor function after stroke: a prospective controlled trial. J. Neuroeng. Rehab. **10**(1), 85 (2013)
31. Vanbellingen, T., Filius, S.J., Nyffeler, T., van Wegen, E.E.: Usability of videogame-based dexterity training in the early rehabilitation phase of stroke patients: a pilot study. Front. Neurol. **8**, 654 (2017)
32. Wade, E., Winstein, C.J.: Virtual reality and robotics for stroke rehabilitation: where do we go from here? Top. Stroke Rehab. **18**(6), 685–700 (2011)
33. Wardini, R., et al.: Using a virtual game system to innovate pulmonary rehabilitation: safety, adherence and enjoyment in severe chronic obstructive pulmonary disease. Can. Respir. J. **20**(5), 357–361 (2013)
34. Werium: INICIO - Werium Solutions (2019). http://www.weriumsolutions.com/. Accessed Feb 2019
35. Wittmann, F., et al.: Self-directed arm therapy at home after stroke with a sensor-based virtual reality training system. J. Neuroeng. Rehab. **13**(1), 75 (2016)
36. Zago, M., et al.: Gait evaluation using inertial measurement units in subjects with parkinson's disease. J. Electromyogr. Kinesiol. (2018)
37. Zhang, Z.: Microsoft kinect sensor and its effect. IEEE Multimed. **19**(2), 4–10 (2012)
38. Zhou, H., Razjouyan, J., Halder, D., Naik, A.D., Kunik, M.E., Najafi, B.: Instrumented trail-making task: application of wearable sensor to determine physical frailty phenotypes. Gerontology **65**, 186–197 (2018)