Chapter 7
Virtual Reality Gaming with Immersive User Interfaces

Emmanouela Vogiatzaki and Artur Krukowski

Abstract Stroke is one of most common diseases of our modern societies with high socio-economic impact. Hence, rehabilitation approach involving patients in their rehabilitation process while lowering costly involvement of specialised human personnel is needed. This chapter describes a novel approach, offering an integrated rehabilitation training for stroke patients using a serious gaming approach based on a Unity3D virtual reality engine combined with a range of advanced technologies and immersive user interfaces. It puts patients and caretakers in control of the rehabilitation protocols, while leading physicians are enabled to supervise the progress of the rehabilitation via Personal Health Record. Possibility to perform training in a familiar home environment directly improves the effectiveness and compliance in performing personal rehabilitation.

7.1 Introduction

Stroke affects about 2 Million [1] people every year in Europe. For these people the effect of stroke is that they lose certain physical and cognitive abilities at least for a certain period. More than one-third of these patients i.e. more than 670,000 people return to their home with some level of permanent disability leading to a significant reduction of quality of life, which affects not only the patients themselves but also their relatives. This also increases costs of the healthcare services associated with hospitalisation, home services and rehabilitation. Therefore, there is a strong need to improve ambulant care model, in particular, at the home settings, involving the patients into the care pathway, for achieving maximal outcome in terms of clinical as well as quality of life.

E. Vogiatzaki (✉)
Research for Science, Art and Technology (RFSAT) Ltd, Sheffield, UK
e-mail: emmanouela@rfsat.com

A. Krukowski
Intracom S. A. Telecom Solutions, Peania, Greece

E. Vogiatzaki, A. Krukowski (eds.), *Modern Stroke Rehabilitation through e-Health-based Entertainment*, DOI 10.1007/978-3-319-21293-7_7

7.2 The Concept

The StrokeBack project addresses both of the indicated problem areas. The goal of the project is the development of a telemedicine system to support ambulant rehabilitation at home settings for the stroke patients with minimal human intervention. With StrokeBack, the patients would be able to perform rehabilitation in their own home where they feel psychologically better than in care centres. In addition, the contact hours with a physiotherapist could be reduced thus leading to a direct reduction of healthcare cost. By ensuring proper execution of physiotherapy trainings in an automated guided way modulated by appropriate clinical knowledge and in supervised way only when necessary, StrokeBack aims to empower and stimulates patients to exercise more while achieving better quality and effectiveness than it would be possible today. This way StrokeBack system is expected to improve rehabilitation speed, while ensuring high quality of life for patients by enabling them to continue rehabilitation in their familiar home environments instead of subjecting them to alien and stressful hospital settings. This offers also means of reducing indirect healthcare cost as well.

The concept of StrokeBack is complemented by a Patient Health Record (PHR) system in which training measurements and vital physiological and personal patient data are stored. Thus, PHR provides all the necessary medical and personal information for the patient that rehabilitation experts might need in order to evaluate the effectiveness and success of the rehabilitation, e.g. to deduce relations between selected exercises and rehabilitation speed of different patients as well as to assess the overall healthiness of the patient. In addition, the PHR can be used to provide the patient with mid-term feedback, e.g. her/his, rehabilitation speed compared to average as well as improvements over last day/weeks, in order to keep patient motivation high.

The StrokeBack project aims at increasing the rehabilitation speed of stroke patients while patients are in their own home. The benefit we expect from our approach is twofold. Most patients feel psychologically better in their own environment than in hospital and rehabilitation speed is improved. Furthermore, we focus on increasing patients' motivation when exercising with tools similar to a gaming console.

The StrokeBack concept puts the patient into the centre of the rehabilitation process. It aims at exploiting the fact the patients feel better at home, that it has been shown that patients train more if the training is combined with attractive training environments [2, 3]. First, the patients learn physical rehabilitation exercises from a therapist at the care centre or in a therapists' practice. Then the patients can exercise at home with the StrokeBack system monitoring their execution and providing a real-time feedback on whether the execution was correct or not. In addition, it records the training results and vital parameters of the patient. This data can be subsequently analysed by the medical experts for assessment of the patient recovery. Furthermore, the patient may also receive midterm feedback on her/his personal recovery process. In order to ensure proper guidance of the patient, the therapist also gets information from the PHR to assess the recovery process enabling him to decide whether other training sequences should be used, which are then introduced to the patient in the practice again.

7.3 Game-Based Rehabilitation

The use of virtual, augmented or mixed-reality environments for training and rehabilitation of post-stroke patients opens an attractive avenue in improving various negative effects occurring because of brain traumas. Those include helping in the recovery of the motor skills, limb-eye coordination, orientation in space, everyday tasks, etc. Training may range from simple goal-directed limb movements aimed at achieving a given goal (e.g. putting a coffee cup on a table), improving lost motor skills (e.g. virtual driving), and others. In order to increase the efficiency of the exercises advanced haptic interfaces are developed, allowing direct body stimulation and use of physical objects within virtual settings, supplementing the visual stimulation.

Immersive environments have quickly been found attractive for remote home-based rehabilitation giving raise to both individual and monitored by therapists remotely. Depending on the type of a physical interface, different types of exercises are possible. Interfaces like Cyber Glove [4] or Rutgers RMII Master [5] allow the transfer of patient’s limb movement into the virtual gaming environment. They employ a set of pressure-sensing servos, one per finger, combined with motion sensing. This allows therapists to perform, e.g. range of motion, speed, fractionation (e.g. moving individual fingers) and strength (via pressure-sensing) tasks. Games include two categories: physical exercises (e.g. DigiKey, Power Putty) and functional rehabilitation (e.g. Peg Board or Ball Game) ones. They use computer monitors for visual feedback. Cyber Glove has been used by Rehabilitation Institute of Chicago [2] also for assessing the pattern of finger movements during grasp and movement space determination for diverse stroke conditions. Virtual environments are increasingly used for functional training and simulation of natural environments, e.g. home, work, outdoor. Exercises may range from simple goal-directed movements [6] to learn/train for execution of everyday tasks.

Current generation of post-stroke rehabilitation systems, although exploiting latest immersive technologies tend to proprietary approaches concentrating on a closed range of exercise types, lacking thoroughly addressing the complete set of disabilities and offering a comprehensive set of rehabilitation scenarios. The use of technologies is also very selective and varies from one system to another. Although there are cases of using avatars for more intuitive feedback to the patient, the use of complicated wearable devices makes it tiresome and decreases the effectiveness of the exercise [3]. In our approach we have been exploring novel technologies for body tracing that exploit the rich information gathered by combining wearable sensors with visual feedback systems that are already commercially available such as

Microsoft Kinect [7] or Leap Motion [8] user interfaces and 3D virtual, augmented and mixed-reality visualisation.

The environment we develop aims to provide a full 3D physical and visual feedback through Mixed-Reality interaction and visualisation technologies placing the user inside of the training environment. Considering that detecting muscle activity cannot be done without wearable device support, our partner in the project, IHP GmbH, has been developing a customizable lightweight embedded sensor device allowing short-range wireless transmission of most common parameters including apart from EMG, also other critical medical signs like ECG, Blood Pressure, heart rate, etc. This way the training exercises become much more intuitive in their approach by using exercise templates with feedback showing correctness of performed exercises. Therapists are then able to prescribe a set of the rehabilitation exercises as treatment through the EHR/PHR platform(s) thus offering means of correlating them with changes of patient's condition, thus improving effectiveness of patients' recovery.

7.4 Body Sensing and User Interfaces

In order to enable the tracking the correctness of performed exercises automatically without the constant assistance of the physicians, an automated means of tracking and comparing patient's body movement against correct ones (templates) has to be developed. This is an ongoing part of the work due to the changing requirements from our physicians. Although many methods are in existence, most of them employ elaborate sets of wearable sensors and/or costly visual observations. In our approach we initially intended to employ a proprietary approach using visual-light scanning, but the recent availability of new Kinect, Prime Sense and Leap Motion sensors made us change our approach and use existing IR-LED solutions.

When better accuracy is required that offered by 3D scanners then additional microembedded sensor nodes are employed, e.g. gyros (tilt and position calibration) and inertial/accelerometers (speed changes). Such are readily available for us in both EPOC EEG U/I from Emotiv (used currently as a U/I, though intended to be used in the future for seizure risk alerting) and on Shimmer EMG sensor platform that we use for detecting muscle activity during the exercises. Considering very small sizes of such sensors (less than 5×5 mm each) a development of lightweight wireless energy-autonomous (employing energy harvesting) may be possible.

Muscle activity poses problems for measurement since it has been well known for many years [9] that the EMG reflects effort rather than output and so becomes an unreliable indicator of muscle force as the muscle becomes fatigued. Consequently measurement of force, in addition to the EMG activity, would be a considerable step forward in assessing the effectiveness of rehabilitation strategies and could not only indicate that fatigue is occurring, but also whether the mechanism is central or peripheral in origin [10]. Similarly, conventional surface EMG measurement requires accurate placement of the sensor over the target muscle, which would be inappropriate for a sensor system integrated within a garment for home use. Electrode arrays are, however, now being developed for EMG measurement and signal processing is used to optimise the signal obtained. Several different solutions have been investigated to offer sufficiently reliable, but also economic muscle activity monitoring. Finally, we concluded on using EMG sensors on the 2R sensor platform from Shimmer for system development purposes, while a dedicated solution made by IHP GmbH.

However, EMG is not the only sensor that is needed for home hospitalisation of patients suffering from chronic diseases like stroke. This requires novel approaches to combining building blocks in a body sensor network. Existing commercial systems provide basic information about activity such as speed and direction of movement and postures. Providing precise information about performance, for example relating movement to muscle activity in a given task and detecting deviations from normal, expected patterns or subtle changes associated with recovery, requires a much higher level of sophistication of data acquisition and processing and interpretation. The challenge is therefore to design and develop an integrated multimodal system along with high-level signal processing techniques and optimisation of the data extracted. The Kinect system has potential for use in haptic interfacing [11] and has already been used in some software projects, and Open Source software libraries are available for browsers like Chrome [12] and demonstrations of interfaces to Windows 7 [3] systems have been shown.

The existing techniques for taking measurements on the human body are generally considered to be adequate for the purpose but are often bulky in nature and cumbersome to mount, e.g. electro-goniometers, and they can also be expensive to implement, e.g. VIACON camera system. Their ability to be used in a home environment is therefore very limited. In this context, we have decided to address those deficiencies by extending the state-of-the-art in the areas of:

- Extending the application of existing sensor technologies: For example, we tend to use commercially available MEMS accelerometers with integrated wireless modules to measure joint angles on the upper and lower limbs in order to allow wire-free, low-cost sensor nodes that are optimised in terms of their information content and spatial location.
- Novel sensing methodologies to reduce the number of sensors worn on the human body, while maintaining good information quality. For example, many homes now have at least one games console (e.g. Xbox, Nintendo Wii, etc.) as part of a typical family home entertainment system. With the advent of the Xbox Kinect system, the position and movement of a human will be possible to be monitored using a low-cost camera mounted on the TV set.
- Easy system installation and calibration by non-experts for use in a non-clinical environment, thus making this solution suitable for use at home for the first timer and with support or untrained caretakers and family.
- Transparent verification of correct execution of exercises by patients may be based on data recorded by Body Area Networks (BAN), correlation of prescribed therapies with medical condition thus allows to determine their effectiveness on patient’s condition, either it is positive or negative.

7.5 Prototyping

The project has reached 2 years of its lifetime already and the prototyping as well as integration of various technologies have already started. This refers to physiological monitoring with Shimmer sensors, gaming user interfaces as well as the games themselves, focussing on the Unity3D engine. A sub-unit assembly diagram of the ‘Patients home training place’ is depicted in Fig. 7.1 and shown placed on a patient in Fig. 7.2. The blue and grey rectangles designate respective elements, while green ones are the user interfaces. The PTZ-camera features pan-tilt-zoom. Arrows show the data flow. The description of the user interfaces shown in this diagram follows.

Since home-based rehabilitation may increase the risk of stroke re-occurrence we have decided to include EEG sensor, a 2 × 7-node Emotiv EPOC EEG, which we use for monitoring of brain signals, looking for ‘flashing’ activity between the two brain spheres, indicated by participating physiotherapists as a sign of a likely pre-event condition (Fig. 7.3).

This device offers an additional benefit for being used as a supplementary gaming interface, thus shifting patient’s perception from its use as a preventive device to enjoying and using for controlling games with the ‘power of the mind’. It is not without a merit that Emotiv offers also a Unity3D support for its device, not to mention ongoing development of an even more powerful INSIGHT [13] sensor version.

Currently apart from searching for clues indicating pre-stroke risks and as ‘mouse’-like user interface, we use EPOC for establishing a correlation between

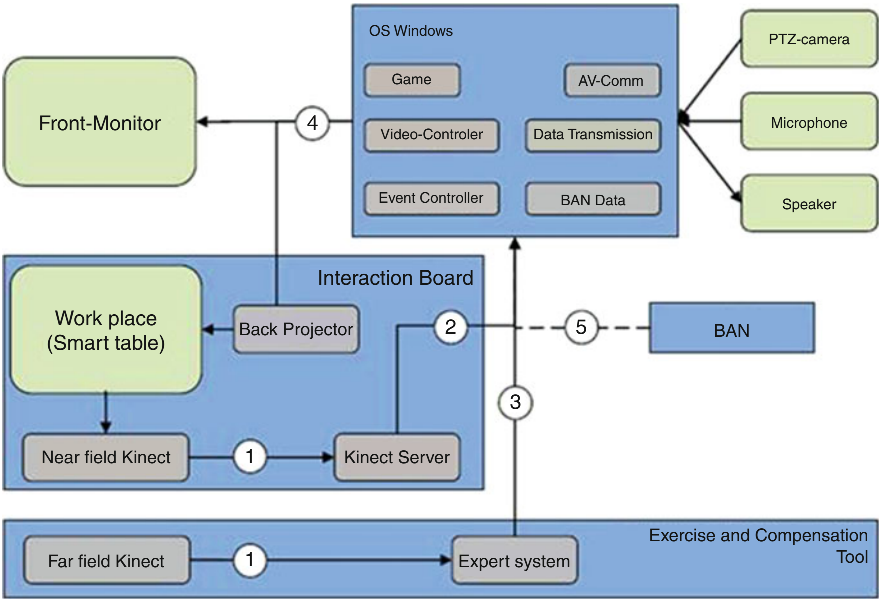


Fig. 7.1 Integration of the overall ‘home’ system

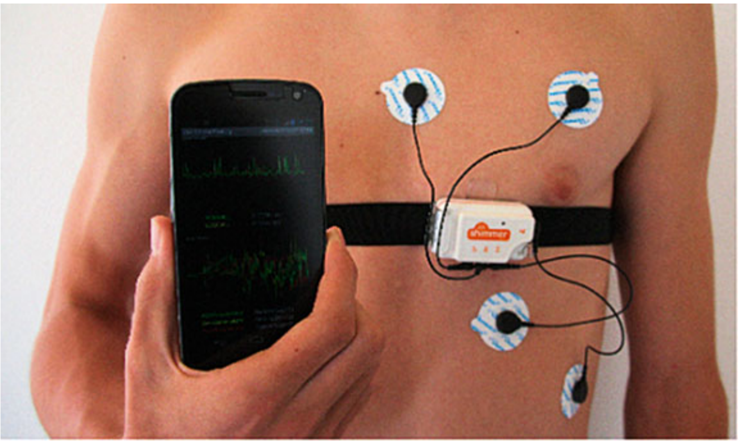


Fig. 7.2 ECG sensing with Shimmer2R

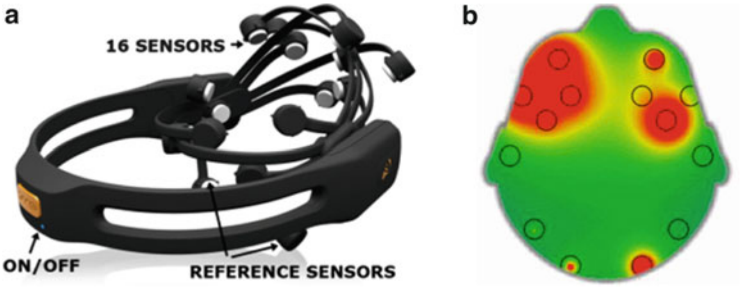


Fig. 7.3 Emotiv EEG (**a**), sample brain activity (**b**)

the mental intention to move a limb and the physical action. By combining with data from EMG sensors we aim to detect cases when patient's brain correctly issues a signal to, e.g. to move an arm, but the patient cannot do it, e.g. due to a broken nerve connection.

7.5.1 'Kinect Server' Implementation

The principal user interface used to control games has been Microsoft Kinect, the Xbox version at first and then the Windows version when it has been first released in early 2012. Its combination of distance sensing with the RGB camera proved perfectly suitable for both full-body exercises (exploring its embedded skeleton recognition) as well as for near-field exercises of upper limbs. However since Kinect has not been designed for short-range scanning of partial bodies, the skeleton tracking could not be used and hence we had to develop our own algorithms that would be able to recognise arms, palms and fingers and distinguish them from the background objects. This has led to the development of the 'Kinect server' based on open source algorithms. The first implementation has used Open NI drivers, closed in April 2014 following the acquisition of PrimeSense by Apple [14], offering the opportunity for our software to be built for both MS Windows and Linux platforms.

The 'Kinect Server' has been a custom development from RFSAT Ltd in the StrokeBack project to allow remote connectivity to the Microsoft Kinect sensor and subsequently the use of the data from the sensor on a variety of devices, not normally supported natively by the respective SDK from Microsoft. Initial implementation of the Kinect Server has been based on Open NI drivers for Xbox version of the device, which has been later translated to a more general drivers supplied by Microsoft in their Kinect SDK version 1.7 and the subsequent versions.

The principle of its operation is that it is comprised of two components:

- *Server-side*—operating on a software platform supported by the selected Kinect drivers, having a role of obtaining relevant data from the Kinect sensor and making it available in a suitable form via network to connected clients.
- *Client-side*—operating on any platform where WEB browsers with embedded Java Scripts are supported. This means almost any networked device, including tablets and a variety of smartphones, even Smart TVs, and other devices.

The types of information made available to by the server to clients include both RGB and depth map as images as well as the list of detected objects. Customisations include for example such features as the limitations of the visibility (detection) area, thus allowing to reduce clutter from nearby objects, focus on the selected object (e.g. the central or the closest one) and other ones.

Two modes of operation have been anticipated in order to enable its use for rehabilitation training in the project (refer to Fig. 7.4). In case that the persistent network connectivity can be maintained, the server part could operate on the home gateway, the client on a game console while the game server (game repository and management of game results for each user) remotely on the same server that hosts the PHR platform. In such an approach home client would not need to bother about updating games to latest versions or managing results. However, in case that network connectivity may or may not be secured, then the game server would need to be hosted locally on a home gateway, alongside the Kinect Server.

Since server has been implemented as a generic enabler, a supplementary implementation of gesture recognition geared to the Kinect Server has been implemented. In order to achieve wide interoperability on a number of devices and operating systems, an approach using Python script 'palm_controls' has been selected with a purpose of detecting specific gestures and mapping them to a custom keystrokes and mouse actions. The list of features and calling syntax is shown below:

palm_controls <server IP> <server PORT> options

Sever IP—network address where Kinect Server is hosted

Server PORT—network port on which the server listens for connections

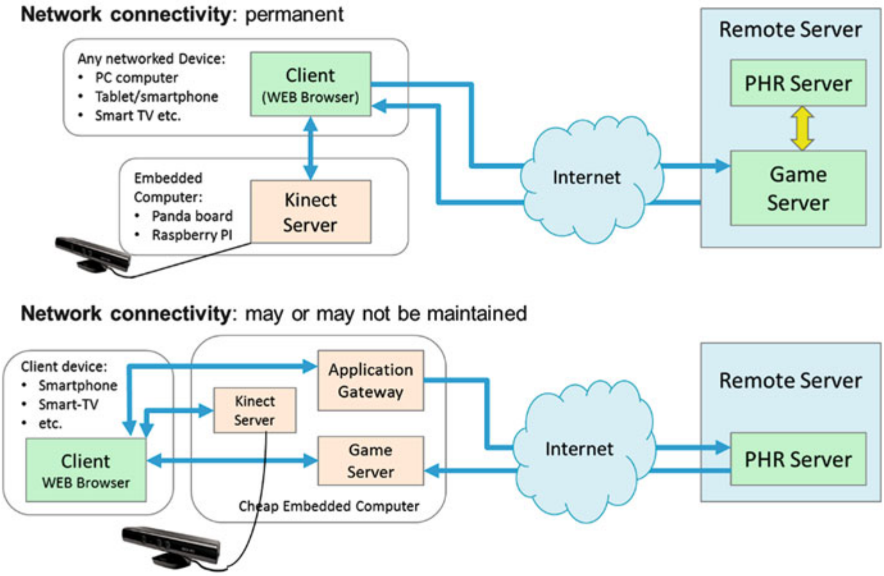


Fig. 7.4 Network operational modes of the 'Kinect Server'

Options:

- '-lHH –rHH –uHH –dHH -sHH'

  Defines keys pressed for left, right, up, down and click actions
  HH is a hexadecimal number from the Microsoft character table:
  http://msdn.microsoft.com/en-us/library/windows/desktop/dd375731(v=vs.85).aspx
  Choosing a zero (0) for a given key disables mapping of the given gesture

- '-x??? –X??? –y??? –Y??? –z??? –Z???'

  This allows the 3D space where objects are detected.
  Parameters define *Xmin*, *Xmax*, *Ymin*, *Ymax*, *Zmin* and *Zmax*, as given below:
  Xmin & Xmax are given in pixels in the range 0–320
  Ymin & Ymax are given in pixels in the range 0–240
  Zmin & Zmax are given in millimeters, refer to a distance from the sensor

7.5.2 Embedded Kinect Server

The limitations of the Kinect in terms of the compatibility with certain Operating Systems, diversity of often-incompatible drivers and restrictiveness to high-end computing platforms have pushed us to investigating alternative ways of interacting

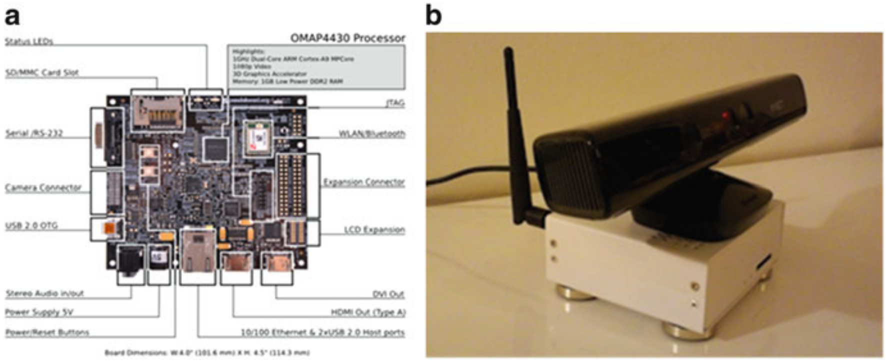


Fig. 7.5 Embedded Kinect server deployment: Panda board (**a**) and physical prototype integrated with the MS Kinect for Windows sensor (**b**)

with Kinect devices. This has led to the attempt to develop an 'Embedded Kinect Server' or EKS. Our idea was to use a microembedded computer like Raspberry PI [15] or similar and allow the client device that was running the game to access data from the EKS via local wireless (or wired) network. Such an approach would allow us to remove the physical connectivity restriction of the Kinect and allow 3D scanning capability from any device as long as it was connected to a network.

Various embedded platforms were investigated: Raspberry PI, eBox 3350 [16], Panda board [17] (Fig. 7.5) and many other ones. Tests have revealed an inherent problem with Kinect physical design that is shared between the Xbox and the subsequent Windows version that is the need to draw high current from USB ports in order to power sensors despite separate power supply still required.

Hardware modifications of the Raspberry PI aimed to increase the current supplied to its USB ports, use of powered external USB hub and other work-around proved all unsuccessful. To date only the Panda Board proved to be the only embedded computer able to maintain the Kinect connectivity and running our EKS. In our tests we have managed to run the Mario Bros game on an Android smartphone and use Kinect wirelessly to control a game with patient's wrists.

7.5.3 *Rehabilitation Games Using the 'Kinect Server'*

The main features of our implementation offer the capabilities of restricting the visibility window, filtering the background beyond prescribed distance, distinguishing between separate objects, etc. This way we were able to implement the Kinect-based interface where following the requirements of our physiotherapists we replaced the standard keyboard arrows with gestures of the palm (up, down, left, right and open/close to make a click). Such an interface allowed for the first game-based rehabilitation of stroke patients suffering from limited hand control. The tests

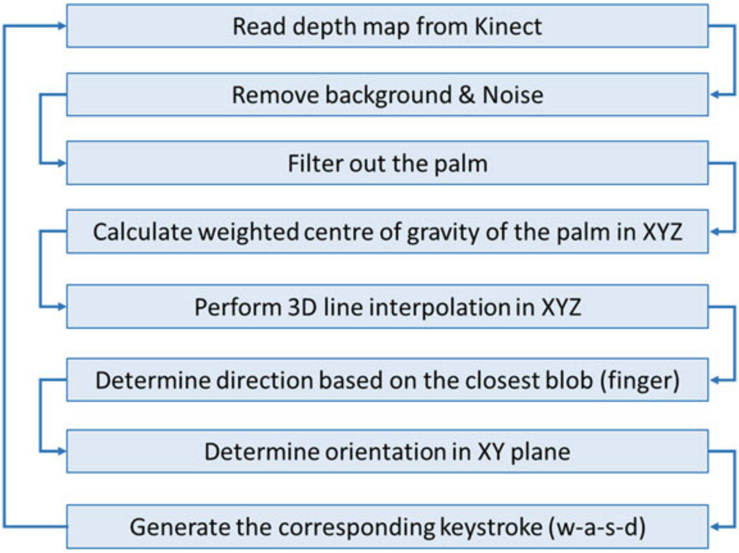

Read depth map from Kinect
Remove background & Noise
Filter out the palm
Calculate weighted centre of gravity of the palm in XYZ
Perform 3D line interpolation in XYZ
Determine direction based on the closest blob (finger)
Determine orientation in XY plane
Generate the corresponding keystroke (w-a-s-d)

Fig. 7.6 Wrist position detection algorithm

were first made with Mario Bros game where all controls were achieved purely with movements of a single palm. The algorithm for analysing wrist position and generating respective keyboard clicks has been developed initially in Matlab and then ported to PERL for deployment along with the Kinect server on an embedded hardware.

The algorithm is shown in Fig. 7.6. It is based on the idea that assuming that the wrist is placed steadily on a support (requirements from physiotherapists), the patients palm would always have fingers closer to the Kinect than the rest of a hand, this allowing easily to determine the palm position and direction the fingers point. Under such condition, we did not have to pay much attention to Kinect calibration and could avoid fixing the relative position of the hand support with respect to the Kinect device. This allowed us to remove the background simply by disregarding anything more distant than the average palm length centred on the centre of gravity (i.e. centre of the palm itself).

The line was then interpolated through remaining points in 3D, whereby the closest point detected was indicating the tip of the closest finger. The direction of the line was equivalent to the movement of a hand in a given direction allowing us to generate the correct keystroke combination (w-a-s-d for N-W-S-E). Since the accuracy was, better than 1/8 of the circle it allowed us to determine also diagonal movements (double keystrokes, i.e. wd-wa-sd-sa for NW-NE-SE-SW). A predefined time delay was applied corresponding to control detection 'speed'.

Since classical rehabilitation required the use of physical objects, like cubes of glasses we have implemented subsequently a 'cube stacking' game, where patient had to use the physical cubes and place them carefully onto the placeholders

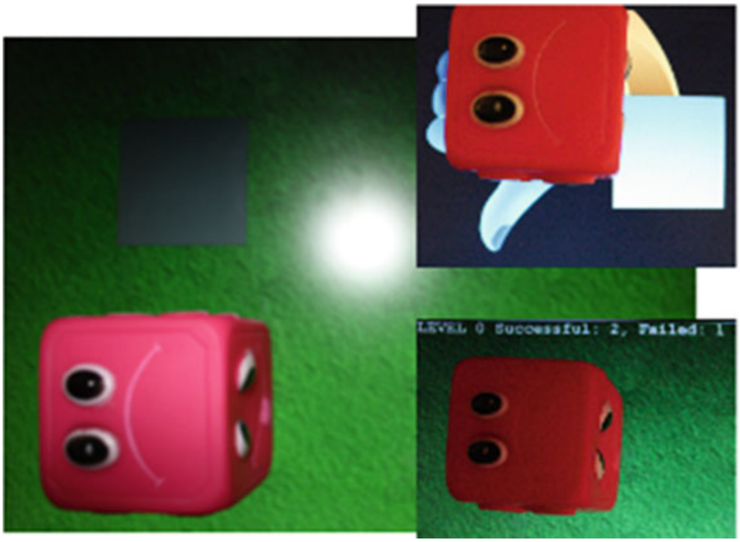


Fig. 7.7 Game using real cubes on a virtual board

displayed on the computer screen positioned flat on the table (later replaced with overhead projection) as shown in Fig. 7.7 where red cubes and placeholders (grey squares) are visible. Here Kinect sensor is placed to scan horizontally at table level, thus allowing to detect XYZ coordinates of the physical cubes. By matching projections with Kinect scan regions allowing detection of correct placement of cubes.

The first level starts with one cube and placeholder parallel to screen edge, then as the game progresses more cubes are used and their requested position could be in any direction. At the end, a score was calculated taking into consideration both the time to place the cubes and the accuracy of placing them over the placeholder. The score was reported in the PHR allowing the physician to track the progress of the patient from one exercise to another. Another variant of the game introduced the possibility of stacking cubes one on top of the other.

An alternative gaming approach to mixing virtual and real objects was a game where patients were requested to throw a paper ball at the virtual circles displayed on the screen as shown in Fig. 7.8. The Kinect sensor synchronised with location of projected object detects physical ball reaching the distance of the wall. Combined with its XY coordinated, this allows to detect the collision.

Such a game allows patients to exercise the whole arm, not just the wrist. Hitting the circle that represented a virtual balloon was rewarded with an animated explosion of the balloon and a respective sound. Such a game proved to be very enjoyable for the patients letting them concentrate on perfecting their movements while forgetting about their motor disabilities, increasing effectiveness of their training.

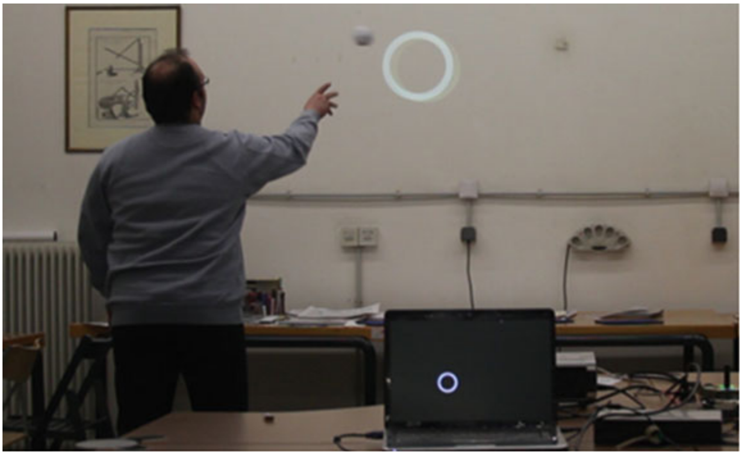


Fig. 7.8 Throwing real paper ball at virtual targets

7.5.4 *Full-Body Games with Avateering*

Subsequently we have investigated more advanced class of games for stroke patients for full-body exercises. In such a case we have chosen to build such games using 3D engine and employ avateering approach, that is patient's body motion capture and its projection onto a virtual avatar. When we have started our first implementation, the MS Kinect SDK was not available yet and hence we have explored various 'hacks' built by the Kinect developer community. The most applicable to our needs appeared to be ZigFu [18], which was compatible with Open NI drivers and easy to use under Unity3D [19] editor.

It proved easier than using commercial products, e.g. Brekel [20] or Autodesk Motion Builder [21]. A prototype system uses environments ranging from familiar home spaces in photorealistic quality [22] (Fig. 7.9) to generic hospital environments (Fig. 7.10).

Scenes with one and two avatars were implemented. The first one was intended as a base for self-training exercises where instruction would be overlaid over the avatar to indicate the movements that the patient would need to perform in order to pass the exercise. A two-avatar scenario was aimed to offer the side-by-side exercise together with a virtual rehabilitator where patient would need to follow the movements of the 'physician' seeing him/herself at the same time. In both cases, the score would be corresponding to the accuracy of following the expected movements. The two scenarios are being subject to assessment by physiotherapists and the decision as to which one will be used for the final system implementation will be depended on evaluation results.

An important advantage of Unity3D over other 3D gaming engines like Cry Engine 3 or Unreal Engine is the possibility to compile games to run either as stand-alone or under from inside a WEB page. The latter approach makes it easier for integrating

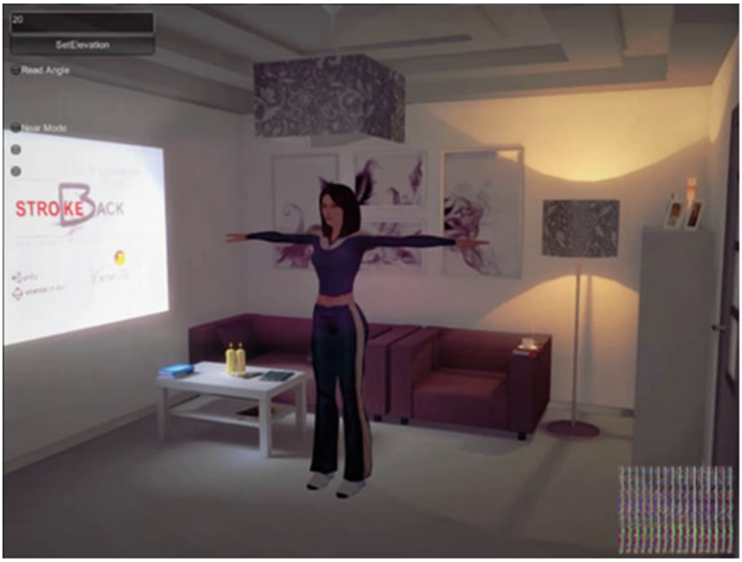


Fig. 7.9 Avateering in a ‘home’ like environment

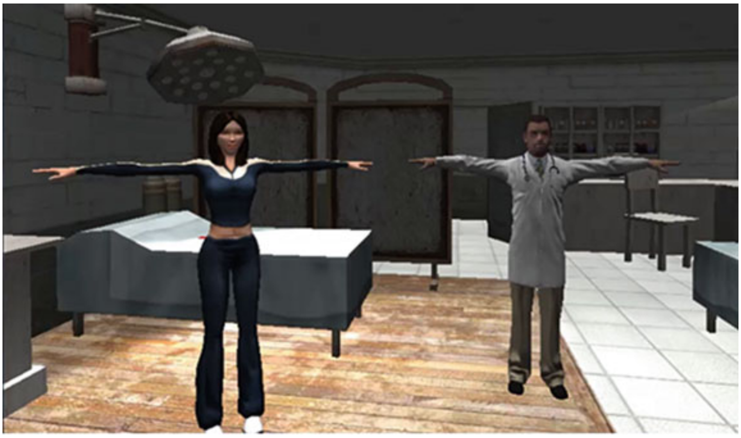


Fig. 7.10 Avateering aimed to repeat movements of an instructor in a ‘hospital’ like virtual environment

games as therapies within the PHR system accessible and controllable via WEB browser. A use of this feature for exercises with a real patient is shown in Fig. 7.11.

The Kinect Server and wrist controls have been integrated into a number of games under Unity3D in order to evaluate the adopted concepts with end users, such as into an ‘Infinite Runner’ [23], shown in Fig. 7.12. It is an incentive-based game combining rehabilitation with entertainment. High scores (coins collected) correspond to improvement in recovering hand movement capabilities.

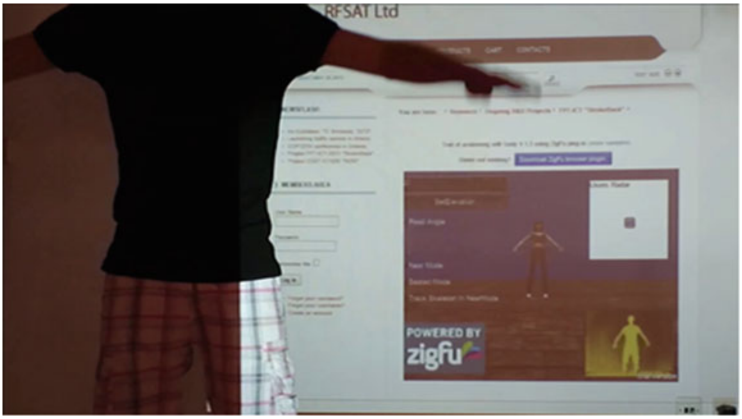


Fig. 7.11 Patient playing online Unity3D game using standard WEB browser

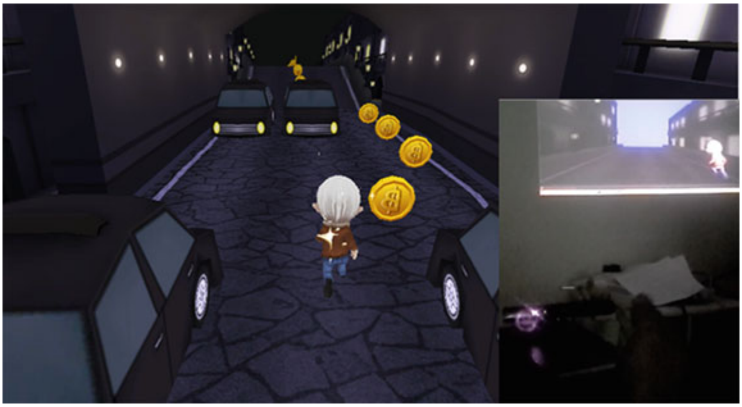


Fig. 7.12 Kinect server used in a wrist-controlled ‘Infinite Runner’ game

The game uses embedded implementation of the Unity3D plug-in for Kinect Server, thus avoiding intermediary use of the Python gesture-interpretation scripts. The game allows exercising lower left and right limb, implementing movements like left and right swipe, up and down swipes, fist and fingers spread, being translated to respective movements of the character, as well as jump and duck actions.

7.5.5 3D Stereoscopic Visualisation

In order to enhance the realism of the games developed with Unity3D engine a stereoscopic projection was implemented offering a sense of depth on supported 3D displays. The current approach has been based on the ‘camera-to-texture’ projection feature available in a PRO version of Unity3D.

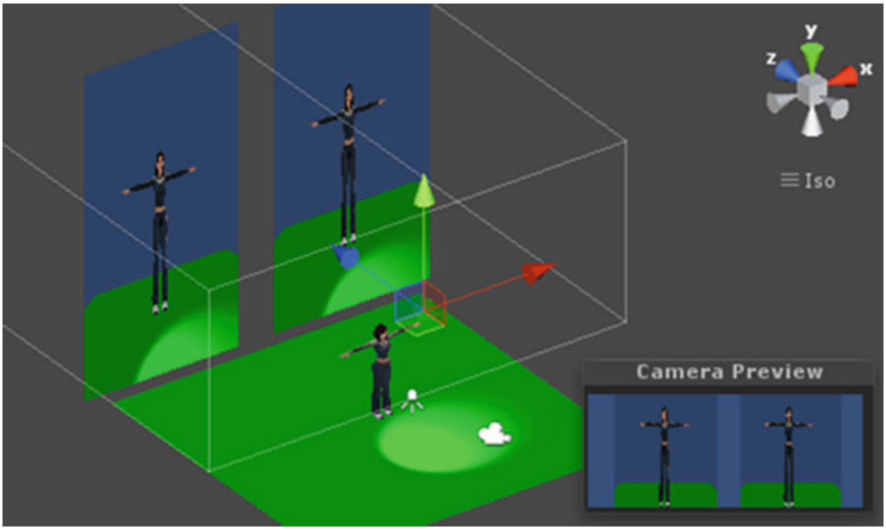


Fig. 7.13 Stereoscopic projection using Unity3D camera-to-texture [24]

It has been based on various published experiments [24, 25]. A dual virtual cameras placed near each other have their images projected onto virtual screens, in turn captured by a single output camera thus creating a side-by-side display (Fig. 7.13). Such an image is then suitable for driving common 3D displays such as of the 3D Smart TV (e.g. Samsung UE46F6400) or 3D projector (e.g. Epson EH-TW5910) with frame-switching 3D glasses. Both of those have been successfully tested with our system. In the future tests we will evaluate also panoramic virtual 3D visors (e.g. Oculus RIFT) and 3D augmented glasses (e.g. VUZIX Star 1200XLD) for virtual and mixed-reality games respectively.

7.5.6 Integration of Leap-Motion User Interface

The latest addition to the portfolio of our user interfaces has been the Leap Motion device. It has proven invaluable for rehabilitation of upper limbs thanks to its superior ability to detect close range distances over the Kinect. Our developments have led us to use features already offered by the Leap Motion SDK and the community applications, e.g. the Touchless for Windows allowing controlling mouse-like control of the Windows applications. This has led us to experiments with standard games used for the rehabilitation of stroke patients like memory board game (as in Fig. 7.14a) and experimenting with operating virtual cubes composing a Stroke word in a virtual 3D space using only your fingers (as in Fig. 7.14b). Both of those have proven very enjoyable for our patients.

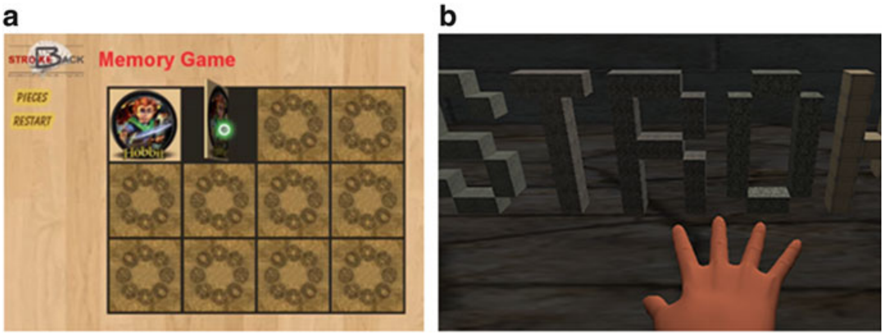
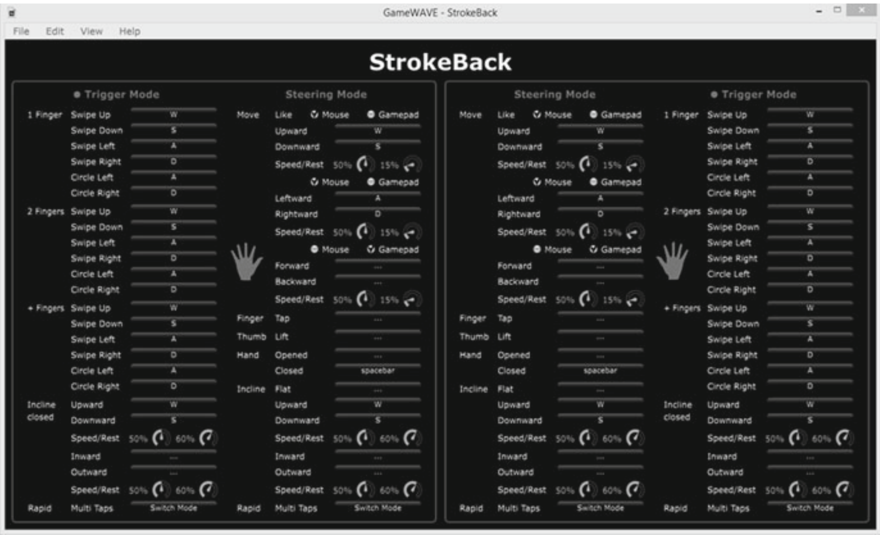


Fig. 7.14 Board games using Leap Motion, Memory (**a**) and Grasping (**b**)



StrokeBack

● Trigger Mode

1 Finger	Swipe Up	W
	Swipe Down	S
	Swipe Left	A
	Swipe Right	D
	Circle Left	A
	Circle Right	D
2 Fingers	Swipe Up	W
	Swipe Down	S
	Swipe Left	A
	Swipe Right	D
	Circle Left	A
	Circle Right	D
+ Fingers	Swipe Up	W
	Swipe Down	S
	Swipe Left	A
	Swipe Right	D
	Circle Left	A
	Circle Right	D
Incline closed	Upward	W
	Downward	S
	Speed/Rest	50% 60%
	Inward	---
	Outward	---
	Speed/Rest	50% 60%
Rapid	Multi Taps	Switch Mode

Steering Mode

Move	Like	○ Mouse ● Gamepad
	Upward	W
	Downward	S
	Speed/Rest	50% 15%
		○ Mouse ● Gamepad
	Leftward	A
	Rightward	D
	Speed/Rest	50% 15%
		● Mouse ○ Gamepad
	Forward	---
	Backward	---
	Speed/Rest	50% 15%
Finger	Tap	---
Thumb	Lift	---
Hand	Opened	---
	Closed	spacebar
Incline	Flat	---
	Upward	W
	Downward	S
	Speed/Rest	50% 60%
	Inward	---
	Outward	---
	Speed/Rest	50% 60%
Rapid	Multi Taps	Switch Mode

Steering Mode

Move	Like	○ Mouse ● Gamepad
	Upward	W
	Downward	S
	Speed/Rest	50% 15%
		○ Mouse ● Gamepad
	Leftward	A
	Rightward	D
	Speed/Rest	50% 15%
		● Mouse ○ Gamepad
	Forward	---
	Backward	---
	Speed/Rest	50% 15%
Finger	Tap	---
Thumb	Lift	---
Hand	Opened	---
	Closed	spacebar
Incline	Flat	---
	Upward	W
	Downward	S
	Speed/Rest	50% 60%
	Inward	---
	Outward	---
	Speed/Rest	50% 60%
Rapid	Multi Taps	Switch Mode

● Trigger Mode

1 Finger	Swipe Up	W
	Swipe Down	S
	Swipe Left	A
	Swipe Right	D
	Circle Left	A
	Circle Right	D
2 Fingers	Swipe Up	W
	Swipe Down	S
	Swipe Left	A
	Swipe Right	D
	Circle Left	A
	Circle Right	D
+ Fingers	Swipe Up	W
	Swipe Down	S
	Swipe Left	A
	Swipe Right	D
	Circle Left	A
	Circle Right	D
Incline closed	Upward	W
	Downward	S
	Speed/Rest	50% 60%
	Inward	---
	Outward	---
	Speed/Rest	50% 60%
Rapid	Multi Taps	Switch Mode

Fig. 7.15 Game WAVE application for Leap Motion [26]

Initial implementation of the games in Fig. 7.15 have used third-party gesture recognition applications, such as Game WAVE for Leap Motion [26], which allowed mapping of pre-defined hand movements to specific keystrokes and mouse movements, thus allowing easy control of any application, including our games. In more recent versions the Unity3D plug-ins from leap Motion have been used, supplemented with custom add-ons aimed to improve the detection of the specific gestures that were important for the selected range of rehabilitation exercises.

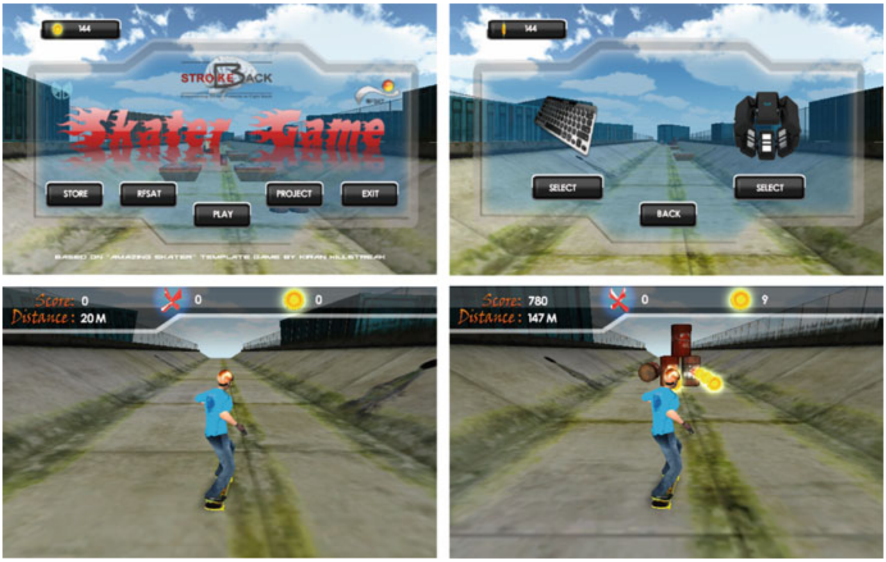


Fig. 7.16 The 'Skater Game' adapted from a template by Ace Games [28]

7.5.7 Integration of Myo Gesture Control Armband

The latest of the sensor that are directly applicable for rehabilitation training of stroke suffering users is an electromyographic (EMG) sensor from Thalmic Labs called 'Myo' [27], launched in early 2015. Such a sensor allows the detection of electrical potentials of the muscles on the affected limbs. In this sense it gives two types of benefits. From one side clinicians can have a direct indication whether signals are correctly sent from the brain to the muscles. On the other hand it offers a possibility of being used as a user interface for controlling rehabilitation games. The manufacturer has released a range of support software, including and SD, Unity3D plug-in and an application manager allowing custom mapping of supported gestures to required keystrokes and mouse actions. A direct access to each of the eight (8) EMG sensors is also readily available by the SDK in order to allow developers to view raw signals from muscles and devising own gesture recognition algorithms.

Our rehabilitation system could not refrain from taking advantage of such a useful interface and a game has been built, adapted from the 'Amazing Skater' Unity3D game template from Ace Games [28], presented in Fig. 7.16.

The game can be played using either a keyboard or a Myo sensor. The latter is supported through a Unity3D plug-in, though it can be also operated using custom StrokeBack application profile via Myo Application Manager.

7.6 Summary and Observations

In the frame of the project, most of the technologies have been implemented, including integrated Smart Table as described in Chapter 9, integration with PHR systems allowing management of rehabilitation by physicians as well as a range of user interfaces, not only based on a Kinect sensor. The initial technical validation tests have proven the viability of the design approach adopted.

The suitability of Leap Motion for 'Touch-Screen'-like applications and game development under Unity3D has been confirmed. Following the success of the technical system tests, the clinical trials with real patients are being conducted since September 2014 into early 2015. Primarily the focus is to be made on the motion capture and recording of the real person (therapist) for subsequent use for demonstration of correct exercises by animating his/her avatar as shown earlier in Fig. 7.10.

Furthermore a 3D hand model needs to be developed, rigged and animated in order to allow its use in Unity3D games. Subsequently the overall integration of the gaming system will be performed whereby selection of games and the necessary data exchange mechanism with the PHR system will be developed. The most difficult work will be related to the real-time comparison of avatar movements for providing an accurate scoring of the correctness of exercises, to be achieved in liaison with the physiotherapists.

References

1. P. Kirchhof, et al, How can we avoid a stroke crisis? 2009, ISBN 978-1-903539-09-5
2. Rehabilitation Institute of Chicago [cited: 2014], http://www.ric.org/conditions/stroke/
3. A. Krukowski, E. Vogiatzaki, J.M. Rodríguez, Patient health record (PHR) system, in *Next-generation remote healthcare: a practical system design perspective*, ed. by K. Maharatna et al. (Springer Science and Business Media, New York, 2013). Chapter 6
4. Virtual Technologies Inc. [cited: 2014], http://www.cyberglovesystems.com/all-products
5. M. Bouzit, G. Burdea, G. Popescu, R. Boian, The rutgers master II—new design force-feedback glove. IEEE/ASME Trans. Mechatron. **7**(2), 256–263 (2002)
6. M. McNeill, et al Immersive virtual reality for upper limb rehabilitation following stroke, in *Systems, IEEE International Conference on Man and Cybernetics*, 2004. ISBN: 0 7803 8566 7
7. Microsoft Kinect [cited: 2014 Aug 17], http://kinectforwindows.org
8. Leap Motion [cited: 2014], https://www.leapmotion.com
9. R. Edwards, O. Lippold, The relation between force and integrated electrical activity in fatigued muscle. J. Physiol. **28**, 677–681 (1956)
10. R. Enkona, D. Stuart, The contribution of neuroscience to exercise studies. Fed. Proc. **44**, 2279–2285 (1985)
11. J. Giles, Inside the race to hack the Kinect. New Sci. **208**(2789), 22–23 (2010). ISSN 0262–4079
12. MIT media lab hacks the Kinect for browser navigation with gestures [cited: 201z4], http://www.readwriteweb.com/2010/11/24/kinect_browser_navigation
13. Emotiv INSIGHT [cited: 2014], http://emotivinsight.com
14. Apple acquires Israeli 3D chip developer prime sense [cited: 2014 Aug 17], http://www.reuters.com/article/2013/11/25/us-primesense-offer-apple-idUSBRE9AO04C20131125

15. Raspberry PI [cited: 2014], http://www.raspberrypi.org
16. Roby Savvy eBox-3350MX [cited: 2014], http://robosavvy.com/store/product_info.php/products_id/1704
17. Panda Board [cited: 2014], http://www.pandaboard.org
18. ZigFu for Unity3D [cited: 2014], http://zigfu.com
19. Unity3D game engine [cited: 2014], http://unity3d.com
20. Kinect marker-less motion capture (Brekel) [cited: 2014], http://www.brekel.com
21. Motion Builder [cited: 2014], http://www.autodesk.com/products/motionbuilder
22. Virtual Room [cited: 2014], https://www.assetstore.unity3d.com/en/#!/content/6468
23. 3D infinite runner toolkit [cited: 2014], http://u3d.as/content/dreamdev-studios/3d-infinite-runner-toolkit/5Nh
24. Lloyd Hooson, Unity 3D stereoscopic development [cited: 2014], http://lloydhooson.co.uk/2010/01/12/unity-3d-stereoscopic-development
25. Human Interface Technology Laboratory Australia (HIT Lab), Stereoscopic 3D, [cited: 2014], http://www.hitlab.utas.edu.au/wiki/Stereoscopic_3D
26. Game WAVE for Leap Motion, https://apps.leapmotion.com/apps/gamewave
27. Myo Gesture Control Armband [cited: 2015], https://www.thalmic.com/en/myo
28. Amazing Skater from Ace Games, http://www.acegames.in
29. Open EMR [cited: 2014], http://www.open-emr.org