A Personalized Limb Rehabilitation Training System for Stroke Patients*

Weibin Wu, Deli Wang, Tianyunyang Wang and Ming Liu

Abstract—**In order to guide patients on their rehabilitation training based on their own conditions and to decrease repetitive work of the rehabilitative physicians, we propose a personalized physical rehabilitation training system using virtual reality technology. On the one hand, with the 9-dof posture sensor and strength sensor, our system realizes collecting patients' data while guiding the patient to complete the rehabilitation task through real-time 3D graphics interface; on the other hand, by gathering historical data of the patient, our system is capable of giving him a personalized rehabilitation plan as reference. Empirical results indicate that our system can offer stroke patients real-time instructions with high accuracy on their rehabilitation training according to the pre-defined training plans. It is able to provide stroke patients with a more convenient, cost-effective and personalized choice for their rehabilitation training.**

I. INTRODUCTION

Stroke has become one of the leading causes of death among people all over the world [1-3]. Stroke patients often suffer from hemiplegia [4], causing a loss of Activities of Daily Living (ADL) at different levels [5]. Related studies show that intensive and repetitive rehabilitation training can facilitate the restoration of patients' body function while improving the patients' ability of life self-care, work and learning [6, 7].

The current clinical method of rehabilitation training is a single-to-single tuition conducted by rehabilitative physicians mainly with the hand operation methods assisted by simple tools. As a result, patients have to receive rehabilitation treatment in hospital or rehabilitation center with the guide of specialized physicians. The disadvantages of the current method are obvious. For example, the period of rehabilitation treatment is usually so long and tedious that it places a great burden on physicians and patients both mentally and financially. At the same time, stroke patients outnumber rehabilitative physicians that the current clinical method depends heavily on. Not only has it increased the physicians' workload and the possibility of making mistakes in their work, but the proper intensity of rehabilitation treatment of single patient can't be guaranteed as well.

People are making great efforts to improve the drawbacks of traditional rehabilitation treatment by the technology of virtual reality. Provided virtual scenes just like in motion sensing games, the rehabilitation training system based on virtual reality technology can guide patients to accomplish the specific training motions repeatedly without rehabilitative physicians. It also helps patients to enjoy the rehabilitation treatment via true-to-life scenarios, which is conducive to their recovery.

Rehabilitation training systems using virtual reality technology are mainly suitable for patients with a certain degree of motor ability. Due to the essential role upper extremities play in our daily life, we focus more attention on developing rehabilitation training systems for upper extremities. To study the influence of this kind of systems on physical exercising and rehabilitation, Taylor et al. did research on existing commercial motion sensing game systems, such as The Nintendo Wii, Sony EyeToy and so on. Empirical results of their work have preliminarily proved the feasibility of rehabilitation training on patients with motor dysfunction using interactive rehabilitation training system based on virtual reality technology [8]. Other researchers have also developed various interactive rehabilitation training systems by the technology of virtual reality. Alana et al. have developed a rehabilitation training system that could guide patients to exercise their shoulders and elbows while the angular ranges of motion had been detected to judge whether patients were doing effective therapeutic movements during their rehabilitation training [9]. Erazo et al. followed Magic Mirror paradigm and developed a rehabilitation training system using Kinect with three kinds of motion sensing games which was designed to enhance the users' ability of arm-stretching and hand-gripping. One motion sensing game of their system, simulating drinking, could train patients by real cups or bottles. And their work has also demonstrated the effectiveness and interestingness of interactive rehabilitation training system using virtual reality technology [10].

But more work should be done to offer better personalized rehabilitation training as to alleviate the burden on physicians while guiding the patients on their rehabilitation training individually. The primary goal of rehabilitation training is to help patients to regain the ability of life self-care, in which case, simple training movements will be more effective if they are designed specifically for the needs of each patient. In a nutshell, our objective is to develop a system that can guide

*This work is supported by the Research Grant Council of Hong Kong SAR Government, China, under project No. 16212815; National Natural Science Foundation of China No. 6140021318; SZSTI Fundamental Research Project 20160197, awarded to Prof. Ming Liu.

Weibin Wu is with the College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China and the Department of Mechanical and Biomedical Engineering, City University of Hong Kong, Hong Kong, China (e-mail: wuweibintj@outlook.com).

Deli Wang and Tianyunyang Wang are with the College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China (e-mail: ephemera_2015@sina.com , wtyy1995@126.com).

Ming Liu is with the Department of Mechanical and Biomedical Engineering, City University of Hong Kong, Hong Kong, China; He is also with the Department of Electronics and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong, China (e-mail: mingliu@cityu.edu.hk).

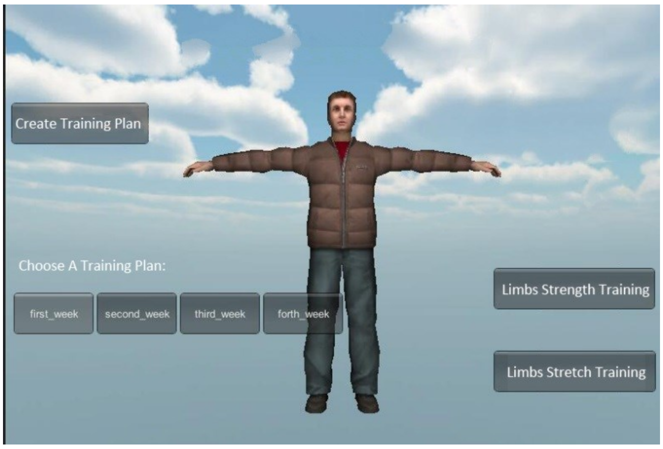

Figure 1. Main Interface of Our System.

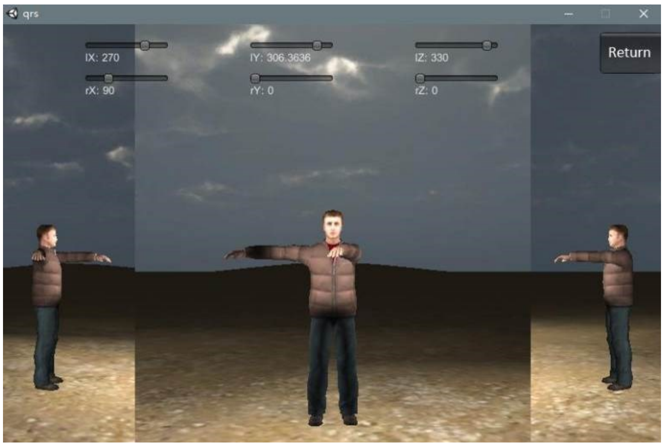

Figure 2. Screenshot of the Stretching Exercise.

patients on their rehabilitation training based on their own conditions. Meanwhile, we reduce the repetitive workload of rehabilitative physicians. The system is to be more convenient, economical and suitable for different patients since they can receive professional rehabilitation training guidance even at home.

The rest of this paper is organized as follows. Section II provides an introduction on the functions and characteristics of our system. Section III describes the system scheme briefly. Then we discuss the hardware implementation scheme and software implementation scheme in detail in Section IV and Section V respectively. Eventually, we will present the testing results of our system in Section VI and conclude the paper in Section VII.

II. FUNCTIONS AND CHARACTERISTICS OF OUR SYSTEM

Three main functions of our system are as follows.

Creating Rehabilitation Training Plan: It allows physicians to create different rehabilitation training plans for different patients according to their inherent conditions. The training plan including the training movements and schedules can totally be designed by physicians. For example, they are able to set the intensity of limbs strength exercise and the standard movements for stretch exercise. After creating some training plans, the main interface of the system is showed in Fig. 1.

Rehabilitation Training Instruction: Using the real-time data collected from different sensors, the system can instruct the patients in how to make limbs stretching exercise and strength exercise. In limbs stretching exercise, the real posture of the user will be shown as a 3D virtual character. Through these characters, users are able to find out whether their movements meet the physicians' requirements and how to do stretching exercise. One of the screenshots of the stretching exercise is shown in Fig. 2. In upper limbs strength exercise, users' strength is reflected by the motion distance of a box, which informs the user whether they reach the training goal or not. In lower limbs strength exercise, the game scene turns to simulate a football-kicking scene.

Rehabilitation Training Evaluation: Using the historical training data of patients, the system not only can show the rehabilitation conditions of each patient, but can give a rehabilitation training plan reference for patients and physicians as well.

In addition to the common merits that other rehabilitation training systems have, such as relatively low cost and convenience , there are other four main advantages of our system compared to them.

Firstly, it is portable enough that our system allows users to move around freely, using wearable device with light sensors and wireless communication equipment. This advantage stands out especially in the posture detection system. Human posture detection system based on optical devices, for example Kinect [11-13], must be used in a relatively fixed area. And when there is a large variation in lighting conditions and background, the recognition accuracy of the system decreases rapidly [14]. However, when it turns to the human posture detection system using wearable sensors, these disadvantages just disappear.

Posture detection and estimation are classical problems in many fields of robotics, such as pose graph establishment [15], robot localization [16], unmanned aerial vehicles attitude estimation [17] and camera pose calculation [18]. And it may be useful to build a multi-sensor system to improve estimation precision, requiring the application of multi-sensor fusion and information retrieval technology [19-22]. What's more, sensor calibration is especially important where wearable sensors are used, as well as some other complex processing work. But taking into account user experience, it's worthwhile to do so.

Secondly, by creating different rehabilitation training plans for different patients, physicians could design specific training movements and workout routine based on the patient's condition during a certain period. It seemed to be more scientific and reasonable than just developing one training plan for all people. What's more, we focused on creating more fundamental training movements, since patients would be capable of practicing more complex movements in daily life if they reach the standard of fundamental training. It will have a good influence on improving patients' confidence in recovery.

0x55	0x53	RollL	RollH	PitchL	PitchH	YawL	YawH

Figure 3. Data Format of Posture Sensor.

What's more, just like other rehabilitation training systems utilizing virtual reality technology, our system guides patients via true-to-life game scenes on their training. What's more, we provide real-time feedback with 3D graphical interface. It adds more funny factors in the training process and motivates individuals to devote themselves in the training. In addition, it reduces the repetitive work of physicians and cost of rehabilitation training.

Finally, collecting user data during rehabilitation training, the system will evaluate the progressing of each patient and give valuable tips when making reasonable and suitable training plans for the patient, which really does a great favor to rehabilitative physicians.

III. SYSTEM SCHEME

A. *System Framework*

The system is developed based on Windows 8 where the Intel Minnow Board Turbot board is the main processor. It contains sensors and wireless communication device, and is considered as a personalized physical rehabilitation training system with the technology of virtual reality, wireless communication, sensor integration and attitude calculation. It's composed by three layers: hardware layer, intermediate layer and application layer.

B. *Introduction to Each Layer*

Hardware layer takes the functions of force detection, gesture recognition, and power supply. It includes the Intel Minnow Board Turbot board, sensors, and some other hardware equipment. Different parts of our system need different power requirements so power adapter or rechargeable batteries are used according to the need of single device. What's more, several sensors are used in the system to detect the posture and strength of users, such as the 9-dof posture sensor and the ultrasonic sensor. The data collected by these sensors will be dealt with and sent to the top layer.

Intermediate layer receives data from the hardware layer and transmits them to the top layer after data fusion. It is mainly made up of the serial communication protocol and the Bluetooth module and realizes the transparent data transmission, which facilitates the development of top layer.

Application layer is the interface between our system and users, mainly developed by Unity 3D and PyCharm. After receiving the data from the intermediate layer, it processes the real-time data and offers services to the users directly.

IV. HARDWARE IMPLEMENTATION SCHEME

Mainly based on the Minnow Board Turbot Embedded Development Platform, the hardware of our system connects sensors and measuring modules by I/O interface and wireless communication devices to realize personalized rehabilitation training functions.

Minnow Board Turbot Embedded Development Platform and wireless master modules exchange data of sensors by serial interfaces. We use the serial ports of posture sensors (JY901 module) to read the posture data while using the ultrasonic sensors connected by I/O interfaces to read the distance data and then transmit these data by Bluetooth module. The transparent transmission can be realized by the Bluetooth master module located on the Minnow Board Turbot Embedded Development Platform and TTL serial port module that can convey the data from the slave module to the embedded development platform. In the application layer, it can distinguish different data sources through the different signal heads which are marked when the data were sent and then does some related data processing to achieve the system functions.

A. *Bluetooth Module*

There are many wireless communication methods, such as visible light communication [23] and Bluetooth communication. In our system, we use Bluetooth Module (HC-05) to achieve the transparent transmission.

The communication parameters of HC05 are as followed: It includes no flow control, no parity bit, 8 data bits and 1 stop bit. Each two of HC05s are considered as a couple of matches to realize the transparent transmission. As for the Bluetooth Module matching with a posture sensor, its data formats are shown in Fig. 3, where 0x55 is regarded as a sign to confirm the start of transmission.

B. *Posture Detection*

Posture Detection Scheme: We choose Chip JY901 as the posture sensor to detect the attitude parameters of four limbs at certain moment. The posture determination accuracy of JY901 is about 0.01degree and the system has a high stability. Users are able to choose two digital interfaces (serial port and IIC) with various rates from 2400bps to 921600bps in serial port and a maximum rate of 400 kHz in IIC.

Altitude Angle Calculation: We use Quaternion Algorithm introduced by Jiang et al. [24] to avoid the problem of singularity from Euler Angle Algorithm. The solution procedure of Quaternion Algorithm is showed as follows.

Step 1: Turn the magnetic field vectors (mxyz) in the body axis system of the electronic compass into magnetic field vectors (hxyz) in the geographical coordinates. Set the value of the vectors in the Y direction equals to 0, and the value in the X direction equals to the projection of the magnetic field vector in plane hxy. Then we get the calibrated vector (bxyz) as showed in (1) where 0.5f means 0.5 float angle.

$$\begin{cases} \mathrm{hx} = 2 \times \mathrm{mx} \times (0.5\mathrm{f} - q_2{}^2 - q_3{}^2) + 2 \times \mathrm{my} \times (q_1 q_2 - q_0 q_3) + \\ \qquad 2 \times \mathrm{mz} \times (q_1 q_3 + q_0 q_2) \\ \mathrm{hy} = 2 \times \mathrm{mx} \times (q_1 q_2 + q_0 q_3) + 2 \times \mathrm{my} \times (0.5\mathrm{f} - q_1{}^2 - q_3{}^2) + \\ \qquad 2 \times \mathrm{mz} \times (q_2 q_3 - q_0 q_1) \\ \mathrm{hz} = 2 \times \mathrm{mx} \times (q_1 q_3 - q_0 q_2) + 2 \times \mathrm{my} \times (q_2 q_3 + q_0 q_1) + \\ \qquad 2 \times \mathrm{mz} \times (0.5\mathrm{f} - q_1{}^2 - q_2{}^2) \\ \mathrm{bx} = \sqrt{hx^2 + hy^2} \\ \mathrm{by} = 0 \\ \mathrm{bz} = \mathrm{hz} \end{cases} \tag{1}$$

Step 2: Premultiply quaternion rotation matrix by the standard gravity vector (0,0,g) in geographical coordinates and calibrates magnetic field vector respectively and turns it into the body coordinates shown in (2) where vxyz refers to gravity reference vector and wxyz refers to magnetic field reference vector.

$$\begin{cases} vx = 2 \times (q_1 q_3 - q_0 q_2) \\ vy = 2 \times (q_0 q_1 + q_2 q_3) \\ vz = q_0 q_0 - q_1 q_1 - q_2 q_2 + q_3 q_3 \\ \mathrm{wx} = 2 \times \mathrm{bx} \times (0.5\mathrm{f} - {q_2}^2 - {q_3}^2) + 2 \times bz \times (q_1 q_3 - q_0 q_2) \\ \mathrm{wy} = 2 \times \mathrm{bx} \times (q_1 q_2 - q_0 q_3) + 2 \times bz \times (q_0 q_1 - q_2 q_3) \\ \mathrm{wz} = 2 \times \mathrm{bx} \times (q_0 q_2 + q_1 q_3) + 2 \times bz \times (0.5f - q_1 q_1 - q_2 q_2) \end{cases} \tag{2}$$

Step 3: Get the cross product of the gravity vector axyz of the acceleration meter and gravity reference vector and do the same for the reference vector and measurement vector of the magnetic field. The result of (3) is used for error correction of gyroscope where exyz is the correction variable.

$$\begin{cases} \mathrm{ex} = (\mathrm{ay} \times \mathrm{vz} - \mathrm{az} \times \mathrm{vy}) + (\mathrm{my} \times \mathrm{wz} - \mathrm{mz} \times \mathrm{wy}) \\ \mathrm{ey} = (\mathrm{az} \times \mathrm{vx} - \mathrm{ax} \times \mathrm{vz}) + (\mathrm{mz} \times \mathrm{wx} - \mathrm{mx} \times \mathrm{wz}) \\ \mathrm{ez} = (\mathrm{ax} \times \mathrm{vy} - \mathrm{ay} \times \mathrm{vx}) + (\mathrm{mx} \times \mathrm{wy} - \mathrm{my} \times \mathrm{wx}) \end{cases} \tag{3}$$

Step 4: Due to the direct proportion between the value of cross product vector and the gyroscope integral error, we need to multiply a factor(Ki and Kp) to revise the data of the gyroscope as in (4) where exInt , eyInt and ezInt stand for integral errors and halfT is the half of calculation period.At last we get the corrected angular velocity vector gxyz.

$$\begin{cases} \mathrm{exInt} = \mathrm{exInt} + \mathrm{ex} \times \mathrm{Ki} \times \mathrm{halfT} \\ \mathrm{eyInt} = \mathrm{eyInt} + \mathrm{ey} \times \mathrm{Ki} \times \mathrm{halfT} \\ \mathrm{ezInt} = \mathrm{ezInt} + \mathrm{ez} \times \mathrm{Ki} \times \mathrm{halfT} \\ \mathrm{gx} = \mathrm{gx} + \mathrm{Kp} \times \mathrm{ex} + \mathrm{exInt} \\ \mathrm{gy} = \mathrm{gy} + \mathrm{Kp} \times \mathrm{ey} + \mathrm{eyInt} \\ \mathrm{gz} = \mathrm{gz} + \mathrm{Kp} \times \mathrm{ez} + \mathrm{ezInt} \end{cases} \tag{4}$$

Step 5: As shown in (5), we calculate the quaternion differential equations and normalize them where norm is the normalization variable. Quaternion differential equations are based on angular velocity, which means we can forecast the altitude in the current cycle according to the angular velocity in the current cycle and the altitude of the last cycle. The error feedback in the angular velocity can adjust the altitude to minimize the error.

$$\begin{cases} \mathrm{q}_0 = \mathrm{q}_0 + (-\mathrm{q}_1 \times \mathrm{gx} - \mathrm{q}_2 \times \mathrm{gy} - \mathrm{q}_3 \times \mathrm{gz}) \times \mathrm{halfT} \\ \mathrm{q}_1 = \mathrm{q}_1 + (\mathrm{q}_0 \times \mathrm{gx} + \mathrm{q}_2 \times \mathrm{gz} - \mathrm{q}_3 \times \mathrm{gy}) \times \mathrm{halfT} \\ \mathrm{q}_2 = \mathrm{q}_2 + (\mathrm{q}_0 \times \mathrm{gy} - \mathrm{q}_1 \times \mathrm{gz} + \mathrm{q}_3 \times \mathrm{gx}) \times \mathrm{halfT} \\ \mathrm{q}_3 = \mathrm{q}_3 + (\mathrm{q}_0 \times \mathrm{gz} + \mathrm{q}_1 \times \mathrm{gy} - \mathrm{q}_2 \times \mathrm{gx}) \times \mathrm{halfT} \\ \mathrm{norm} = \mathrm{invSqrt}(\mathrm{q}_0 \times \mathrm{qo} + \mathrm{q}_1 \times \mathrm{q}_1 + \mathrm{q}_2 \times \mathrm{q}_2 + \mathrm{q}_3 \times \mathrm{q}_3) \\ \mathrm{q}_0 = \mathrm{q}_0 \times \mathrm{norm} \\ \mathrm{q}_1 = \mathrm{q}_1 \times \mathrm{norm} \\ \mathrm{q}_2 = \mathrm{q}_2 \times \mathrm{norm} \\ \mathrm{q}_3 = \mathrm{q}_3 \times \mathrm{norm} \end{cases} \tag{5}$$

Step 6: Convert the quaternions to Euler Angle (i.e. yaw angle,pitch angle and roll angle) in (6), which is more convenient to deal with. And Rad is a constant.

$$\begin{cases} yaw\ angle = \mathrm{atan2}(2 \times \mathrm{q}_1 \times \mathrm{q}_2 + 2 \times \mathrm{q}_0 \times \mathrm{q}_3 - 2 \times \mathrm{q}_2 \times \mathrm{q}_2 - \\ \qquad 2 \times \mathrm{q}_3 \times \mathrm{q}_3 + 1) \times \mathrm{Rad} \\ pitch\ angle = \mathrm{asin}\left(-2 \times \mathrm{q}_1 \times \mathrm{q}_3 + 2 \times \mathrm{q}_0 \times \mathrm{q}_2\right) \times \mathrm{Rad} \\ roll\ angle = \mathrm{atan2}(2 \times \mathrm{q}_2 \times \mathrm{q}_3 + 2 \times \mathrm{q}_0 \times \mathrm{q}_1 - 2 \times \mathrm{q}_1 \times \mathrm{q}_1 - \\ \qquad 2 \times \mathrm{q}_2 \times \mathrm{q}_2 + 1) \times \mathrm{Rad} \end{cases} \tag{6}$$

C. Force Measurement Module

We choose ultrasonic ranging sensor (HC-SR04) to measure the pressure and tensile force. As in the practice, pressure in different areas could lead to varied results for sensors in different positions so we use several ultrasonic ranging sensors to detect the distance information. The average value of these data will be the final result. Matching the distances and values of force, we are able to get the value of force that users are exerting upon the operation panel.

V. SOFTWARE IMPLEMENTATION SCHEME

The developing tools are Unity 3D game engine and Pycharm. Our developing language is JavaScript and Python.

A. Overall Design Process of the Software

The software of our system consists of three main modules. The first module is the data collection module. This module takes advantage of the "pySerial" serial interface communication module of Python to receive the data from sensors. The second module is the scene display module. Developed with Unity 3D game engine, this module is to render 3D models and display the human-computer interface. The third module is the logic processing center. Written by Python, its main job is to exchange data between the data collection module and the scene display module, and to deal with the system logic.

B. System Data Transmission

Data flow in different layers from data collection to human-computer interface. To increase the scalability of the system, we can classify the data transmission into collection layer, processing layer and application layer.

Data collection layer: Collection layer takes charge of the data collection from original hardware sensors, including data sources such as Bluetooth, serial ports and USB. The data collection of host machines in this system uses a third party package of Python (i.e. pySerial). PySerial is one of the tool kits of Python that has a high transmission speed in serial ports.

Data processing layer: After collecting the original data, we need to conduct a unified process in data processing layer, including data calculation, data selection and data format standardization so that it can be used by the upper application layer. Since the data collection layer is written in Python, data processing layer is written in Python as well to avoid the format transformation and the communication problem between different languages.

Application layer: The original data has been transformed into the format that can be used directly in the application layer after processed by the data processing layer. The application layer is mainly made up of Unity 3D game engine. What's

```
Step1: Start the stretching training
Step2: Acquire the attitude data from the 9-dof posture sensor
Step3: Render the attitude of 3D character displayed in screen which is the same as that of users and compare it with the expected motions
Step4:
    If the motion meets the requirement of the stretching training plan, GOTO Step5
    Else, display prompting messages to guide the patients and turn back to Step2
Step5: Display next training motion according to the training plan
```

Figure 4. Program Flow of Stretching Training.

more, two languages, JavaScript and C# language, but not Python can be used to control the logic in Unity 3D. To solve the communication problem among processes and programming languages, our system chooses the UDP protocol to control the Inter-process communication for scalability. On the one hand, the application layer develops a data reception thread to receive the data from data collection layer and motivate application system. On the other hand, application layer can also deliver the human-computer interactive data to data collection layer in order to deal with the logic processing. So the application layer will only be in charge of human-computer interactive process and rendering, excluding the specific processing of system logic. The good point is the core functions will not be affected even when the application layer is replaced.

C. *Limbs Stretching Training*

With the 9-dof posture sensors connected to the limbs of users, the stretching training subsystem enables the 3D character displayed in screen to do the same motions of patients. Compared with the expected motions which are set on the basis of the condition of single patient by rehabilitative physicians in advance, the subsystem will guide users to do the expected motions and judge whether their motions are in accordance with the expected training motions. The program flow is shown in Fig. 4.

D. *Limbs Strength Training*

To help our patients do strength training and acquire a gradual recovery of muscle strength level, we use force transducers to acquire force data from users and set game scenes such as pushing a box and inflating a ball in arms strength training and kicking a ball in legs strength training. In that way, users can get exercised in the game which is more interesting than the traditional methods. The program flow is similar to the stretching training.

TABLE I. ACCURACY OF POSTURE DETECTION

Euler Angles	**Times of Testing**	**Correct Times**	**Accuracy**	**Angular Range**
Roll	100	93	93%	-180° to 180°
Pitch	100	92	92%	-90° to 90°
Yaw	100	90	90%	-180° to 180°

TABLE II. ACCURACY OF STRENGTH TRAINING FUNCTION

Testing item	**Times of testing**	**Correct times**	**Accuracy**
Pressure	100	91	91%
Tension	100	92	92%

VI. SYSTEM TESTING

A. *Functional Testing of Posture Detection*

We chose 10 volunteers to test the accuracy rate of posture detection of our system, where each of them exercised for 10 times. We tied two posture sensors to the upper arms of each volunteer respectively. In each test, the Euler Angles of a single volunteer's upper arms would be presented on the screen. We set one training motion for each test in advance and the testers ought to strike the other nine poses randomly. Then we compared the Euler Angles of volunteers' upper arms with the ones shown on the screen. If the error was within 10%, we defined it as a correct Euler Angle. When the system showed the accurate Euler Angles and made correct judgments about whether testers' movements correspond with the target motions set in advance with an error of 10%, we considered it as a correct test result. The testing results are shown in Table I.

From the testing results we could find out that the accuracy rate of the posture detection of our system is high. The cause of errors could have been the disturbance on the geomagnetism, because the 9-dof posture sensor relied on the geomagnetism to compute the posture data. But we could reduce the error by calibration before using the 9-dof posture sensor.

B. *Functional Testing of Strength Training*

We chose 10 volunteers to test the accuracy of strength training function of our system, where each of them exercised for 10 times. In each test, we have shown force levels with a span of 10N that a single volunteer exerted upon the operation panel on the screen. We set one target force level for each test in advance and the testers could exert a force upon the operation panel randomly for nine times. Then we compared force levels that volunteers exerted upon the operation panel with the ones shown on the screen. If the system has been able to distinguish various pressures and made correct judgments about whether their strength met the requirement of the training plan set in advance, we considered it as a correct result. The testing results are listed in Table II.

From the testing results, we conclude that the accuracy of the strength training function of our system is high enough for the rehabilitation training. The strength training devices consist of spring elements, so it must be used within certain limits and may cause errors after being used for too many times. To solve this problem, we replace the spring elements with new ones whenever the error is intolerable which improves the performance of our system if the system is checked periodically.

VII. CONCLUSIONS

In general, our system offers three main functions which cover the whole process of rehabilitation training for the rehabilitation training of patients. It not only allows the physicians to create various training plans for different patients, but guides the patients how to complete the training plan as well. In addition, it provides the patients and physicians with rehabilitation training assessment reports of users and references to training plans according to patients' own conditions, which makes it more convenient, economical and suitable for different patients.

To enhance our system to a better condition, we believe there's still some future work to be done. Adding more game scenes to our systems will make the rehabilitation training more attractive. We would also like to realize more accurate posture detection of more complex and faster movements so that we are able to improve the user experience of our system and give more reasonable assessment of the patients' progress. We would compare the performance of our system with other rehabilitation training systems if their detailed testing data were available in the future. In addition, we will try to introduce the facial expression recognition [25] and gesture recognition technology to establish a friendlier Natural User Interface and clustering approach [26] to evaluate patient's condition.

REFERENCES

[1] G. B. Prange, M. J. Jannink, C. G. Groothuis-Oudshoorn, H. J. Hermens and M. J. Ijzerman. "Systematic review of the effect of robot-aided therapy on recovery of the hemiparetic arm after stroke", *Journal of Rehabilitation Research & Development*, 2006, 43(2):171-183.

[2] G. Kwakkel, B. J. Kollen and H. I. Krebs. "Effects of robot-assisted therapy on upper limb recovery after stroke: A Systematic Review", *Neurorehabilitation & Neural Repair*, 2008, 22(2):111–121.

[3] X. G. Sun, Y. L.Wang, N. Zhang, T.Wang, Y. H. Liu, X. Jin, L. J. Li and J. Feng. "Incidence and trends of stroke and its subtypes in Changsha, China from 2005 to 2011", *Journal of Clinical Neuroscience*, 2014, 21(3):436-440.

[4] C. Trotti, F. Menegoni, S. Baudo, M. Bigoni, M. Galli and A. Mauro. "Virtual reality for the upper limb rehabilitation in stroke, A case report", *Gait & Posture*, 2009, 30(5):S42.

[5] B. Balaban, F. Tok, F. Yavuz, E. Yaşar and R. Alaca. "Early rehabilitation outcome in patients with middle cerebral artery stroke", *Neuroscience Letters*, 2011, 498(3):204-7.

[6] A. Rizzo and G. Kim. "A SWOT analysis of the field of VR rehabilitation and therapy", *Tele-operators and Virtual Environments*, 2005, 14 (2):119-146.

[7] J. L. Patton, M. E. Stoykov, M. Kovic and F. A. Mussaivaldi. " Evaluation of robotic training forces that either enhance or reduce error in chronic hemiparetic stroke survivors", *Experimental Brain Research*, 2006, 168(3):368-83.

[8] M. J. Taylor, D. Mccormick, T. Shawis, R. Impson and M. Griffin. "Activity-promoting gaming systems in exercise and rehabilitation", *Journal of Rehabilitation Research & Development*, 48.10(2011):1171-86.

[9] A. D. Gama, T. Chaves, L. Figueiredo and V. Teichrieb. "Guidance and movement correction based on therapeutics movements for motor rehabilitation support systems", *Virtual and Augmented Reality, IEEE*, 2012, pp.191-200.

[10] O. Erazo, J. A. Pino , R. Pino, A. Asenjo, C. Fernández and A. Asenjo. "Magic Mirror for neurorehabilitation of people with upper limb dysfunction using Kinect,"in *Proc. 2014 IEEE Int. Conf. System Sciences*, pp. 2607-2615.

[11] I. Pastor, H. A. Hayes and S. J. M. Bamberg. "A feasibility study of an upper limb rehabilitation system using Kinect and computer games," in *Conf. 2012 IEEE Int. Conf. Engineering in Medicine & Biology Society*, pp.1286-9.

[12] C. Y. Chang, B. Lange, M. Zhang and S. Koenig. "Towards pervasive physical rehabilitation using Microsoft Kinect," in *Conf. 2012 Pervasive Health*, pp.159 – 162.

[13] L. Pogrzeba, M. Wacker and B. Jung. "Potentials of a low-cost motion analysis system for exergames in rehabilitation and sports medicine," in *Proc. 2012 Int. Conf. E-Learning and Games for Training, Education, Health and Sports*, pp.125-133.

[14] D. J. Sturman and D. Zeltzer. "A survey of glove-based input", *Computer Graphics & Applications, IEEE*, 1994, 14(1):30-39.

[15] I. Deutsch, M. Liu and R. Siegwart. "A Framework for Multi Robot Pose Graph SLAM", in *Conf. 2016 IEEE Int. Conf. Real-time Computing and Robotics*, Angkor Wat, Cambodia.

[16] S. Wang, F. Colas, M. Liu, F. Mondada and S. Magnenat. "Localization of inexpensive robots with low-bandwidth sensors", *13th International Symposium on Distributed Autonomous Robotic Systems*, 2016, London

[17] W.X. Liu, S. Yang, M. Liu. "A 6D-pose estimation method for UAV using known lines." in *Conf. Rec. 2015 IEEE Int. Conf. Cyber Technology in Automation, Control, and Intelligent Systems*, pp. 953-958.

[18] S.D. Xu, M. Liu. "Feature selection and pose estimation from known planar objects using monocular vision", in *Conf. Rec. 2013 IEEE Int. Conf. Robotics and Biomimetics*, pp. 922-7.

[19] G.Y. Xie, T. Xu, C. Isert, M. Aeberhard, S.H. Li, M. Liu. "Online active calibration for a multi-LRF system", in *Conf. Rec. 2015 IEEE 18th Int. Conf. Intelligent Transportation Systems*, pp.806-11.

[20] M. Liu, L. Wang, and R. Y. Siegwart. "DP-Fusion: A generic framework of online multi sensor recognition." *Autonomous Systems Lab Eth Zurich* (2012):7 - 12.

[21] L. Wang, M. Liu, Q. H. Meng, and R. Siegwart. "Towards real-time multi-sensor information retrieval in Cloud Robotic System", in *Conf. 2012 IEEE Int. Conf. Multi sensor Fusion and Integration for Intelligent Systems,* pp.21-26.

[22] L.J. Wang, M. Liu and Q. H. Meng. "Real-time Multi-sensor Information Retrieval for Cloud Robotic Systems", 2015 IEEE Transactions on Automation Science and Engineering, Vol.12, Number 2.

[23] M. Liu, K.J. Qiu, S.H. Li, F.Y. Che, L. Wu and C. P. Yue. "Towards Indoor Localization using Visible Light Communication for Consumer Electronic Devices", in Conf. 2014 IEEE/RSJ Int. Conf. Intelligent Robots and Systems, pp.143-148.

[24] Y. Jiang, H. Y. Chen and R. P. Cen. "Attitude solution algorithm for four-rotor aircrafts based on four element numbers," *Manufacturing Automation*, 2015,37(23):77-80.

[25] Y. P. Liu, Y. W. Cao, Y. B. Li, M. Liu, R. Song, Y. F. Wang, Z. G. Xu and X. Ma. "Facial expression recognition with PCA and LBP features extracting from active facial patches", in *Conf. Rec. 2016 IEEE Int. Conf. Real-time Computing and Robotics*, Angkor Wat, Cambodia.

[26] M. Liu, F. Colas, F. Pomerleau,and R. Siegwart. "A Markov semi-supervised clustering approach and its application in topological map extraction", in *Proc. 2012 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Vol.7198, pp.4743-4748.