Smartphone's Physiatric Serious Game

Dario Deponti, Dario Maggiorini
Dipartimento di Informatica e Comunicazione
Università degli Studi di Milano
Via Comelico 39, 20135 Milano, Italy
Email: dario@dico.unimi.it

Claudio E. Palazzi
Dipartimento di Matematica Pura e Applicata
Università degli Studi di Padova
Via Trieste 63, 35131 Padova, Italy
Email: cpalazzi@math.unipd.it

Abstract—**Games are not just fun, they can also be used to generate serious benefits for people. The current popularity of smartphones endowed with sensorial, computational and communication capabilities is generating new potential. An interesting case study regards serious games for healthcare purposes and, to this aim, we focus on physiatric rehabilitation of articulations and show how to create a smartphone's serious game for this purpose. In essence, we developed a prototype for a new generation of gametherapies; our application represents a step forward with respect to traditional therapies as it allows for low-cost, ubiquitous exercises with the support of both instantaneous digital feedback from the application and continuous monitoring/interaction with a remote doctor.**

Keywords-**physiatry; smartphone; serious game; ubiquitous healthcare**

I. Introduction

The continuous performance improvement of consumer-level hardware has contributed to the rapid growth in the digital game market. Computer games, once limited to pure entertainment and storytelling, are now trying to propose new experiences [1]–[3].

Among the new generation of games, *serious games* represent class where the user's experience is not just related to entertainment, but may also have a formative training like scientific exploration, healthcare, emergency management, city planning, engineering, religion, and politics [4]–[7]. However, this definition is nowadays outdated as serious games are not anymore solely those involved in training. Instead, they have reached new challenging realms providing the possibility to achieve an important (serious) outcome through the fun and participating paradigm of games [8]. As an example, think to Google Image Labeler, which exploits the effort of a multitude of human players to create an immense database of tagged images [9]; the rationale lies in the fact that appropriately tagging the content of an image can be easily done by humans while it is currently an impossible task for a computer.

Serious games are quickly emerging as a viable solution for many important problems. One of the most promising applicative contexts is certainly represented by healthcare [10]. Indeed, serious games can be successfully exploited in healthcare to foster motivation, enhance prevention, and even offer rehabilitation exercises (*exergames*) such as, for instance, motor rehabilitation of the left arm after a stroke event (*gametherapy*) [11], [12]. Despite the encouraging, even

if few, examples of gametherapies, yet, the following issues about the use of games for healthcare still exist.

- Gametherapy should be tailored for the specific need of each patient; the game should provide the possibility for the doctor to configure the prescribed digital therapy even if both the doctor and the patient are not computer experts.
- During the gametherapy the patient should receive feedback about her/his current performance (e.g., the adherence to the prescription, the improvements of the patient's dexterity) [13].
- The system should be able to memorize information related to the prescription, monitor the patient's behavior, and compare them; the result should be immediately available for the patient and visible, even later, by the therapist.
- Most of the gametherapies developed till now needs some heavy equipment such as a camera, a console, a computer, etc. Instead, as a patient is able to carry her/his pills, she/he should be able also to carry the digital tools for the gametherapy wherever she/he goes.

To this aim, we discuss in this paper an exergame: an application for wrist rehabilitation which runs on a small portable device (i.e., a smartphone) running the Android operating system [14]. Thanks to the hardware capabilities of Android-based smartphones we have implemented our exergame to turn the device into a wrist rehabilitation tool: motion and position sensors will let us drive the player through the right sequence of movements to fully recover hand mobility and articulation [15]. Since the mobile phone is a fully-fledged system a number of complex operations can be performed by the application: i) the therapist can set up exercises uploading them into the smartphone's memory; ii) the system self-adapts to the user as she/he improves during the therapy; iii) movements can be reported back to the doctor to evaluate the effectiveness of the treatment.

Almost everyone has a mobile phone, knows how to use it, and will not get scared to use the device also as a tool for gametherapy. Indeed, differently from other more complex or cumbersome devices, we have a phone in our pocket most of the time and exercises can be performed almost everywhere and at any time.

We named this application *DroidGlove* as an evolution of

the famous Nintendo *PowerGlove* [16]. Whereas the Power-Glove was intended as a purely entertainment control device, the DroidGlove focuses on serious gaming and is a complete standalone system; nevertheless, they share the single-hand intuitive usage and the ability to fit in our everyday life as more technological versions of objects we are already used to carry around.

The rest of this paper is organized as follows. Section II reports on the general gametherapy background. In Section III we present our mobile exergame application, whereas in Section IV we analyze technical issues related to the development of DroidGlove and the adopted solutions. Finally, Section V concludes this paper.

II. RELATED WORK

The interest in developing serious games for general medical applications has grown exponentially in recent years and is expected to keep growing as demonstrated by the increasing number of conferences and initiatives that are eager to discuss the topic. We can distinguish among different specific fields of intervention, each of which has its own dedicated solution.

A first example is the use of serious games by doctors in order to simulate and train. The medical preparation, when supported by appropriate serious games, has been demonstrated to allow precision improvement in surgery [17]. Moreover, serious games can be used also to encourage doctors' prevention and ethics in taking care of hygiene. Hungry Red Planet [18] is a game created by the American National Institutes of Health whose aim is to teach children proper and healthy eating habits. In this field there are also games that teach patients how to deal with issues related to certain diseases, such as Nanoswarm: Invasion from Inner Space and Escape from Diab, by Archimage Inc. [19], which address childhood obesity and diabetes problems.

Another interesting approach is represented by serious games deployed to provide motivational support in case of chronic diseases, such as disabilities or illnesses which need clinical adherence and constant monitoring (e.g., diabetes and post-stroke treatment [11]). Similar, there is the so called *distraction therapy*, aimed to ease the sufferings of the patients forced to difficult or painful treatments. The games, in this case, must provide a distraction therapy and, where possible, play down the suffering [20].

Computer gaming researchers have been offering also a wide panorama of gaming solutions to address problems emerging from the complexity of human mind and personality. Indeed, serious games can be exploited even to address cognitive problems; they can be a useful tool for treatment and prevention, providing immersive and interactive virtual or semi-virtual realities, within which patients can treat their psychological diseases by playing [4], [21]. For instance, mental disease treatment has been studied to treat post traumatic stress lamented by war veterans [17]. In this case, experiments were carried out with the support of commercial off-the-shelf products, such as Full Spectrum Warrior [22]. Similar, virtual reality exposure therapy is a game-derived methodology that

has been employed to address several kinds of pain, anxiety disorder, and phobia [23]–[26]. Games can be very useful for their capability to provide alternative realities where patients could feel safe and step back from the real world. They can be useful even in training cognitive functions such as memory [27], or to ensure a better emotional and mental states control [17].

Pervasive games represent the bridge between mixed realities and computer games, also able to generate augmented reality games, i.e., computer games that mix digital environments with real world elements [28], [29]. These applications can embody portable solutions which are able to foster pervasive healthcare systems. A very interesting field of study in this context is represented by the development of haptic interfaces which have been proposed to reproduce the stimulation a patient could receive from a therapist [30].

Games involving the sensory-motor system, from reaction times to spatial skills, are more than serious candidates for neural and, in general, functional rehabilitation in everyday tasks [31], [32]. Virtual/augmented reality technology has been used in stroke and motor rehabilitation. A study investigated on the influence of a 3D game on motor re-learning on patients suffering from a left arm paresis after stroke and demonstrated the improvement in motor pattern, grip strength, and manual dexterity [12]. These technologies clearly embody promising tools for rehabilitation; yet, their capabilities are far from being fully exploited.

Games have introduced new approaches to rehabilitation and the most interesting part is the factual possibility of using games to improve the life quality of disadvantaged categories. A representative example is represented by AudioDOOM, a game in which players must destroy virtual enemies using clues from a 3D sound environment, with the effect of improving the hearing abilities of those who, temporarily or permanently, lost their sight, in order to allow the construction of spatial and cognitive maps [33].

A. Exergames

The *exergaming* approach is extremely relevant for our purposes. Creating an exergame corresponds to the creation of a context in which the patient can use certain tools to replicate a series of real movements that promote physical rehabilitation. Exergames hence embody an interactive environment where the patient/player is requested to accurately perform a sequence of movements. In this context, elements such as motivation and engagement come from the entertaining nature of games. Exergames are generally designed so as to provide immediate feedback to the patient/player on her/his skills. Moreover, they often provide the doctor with useful information through the possibility of monitoring the patient's behavior and in tracking it over time.

Many exergames have been proposed to promote and support fitness, with positive effects on health. From this point of view, it is worth to mention games such as Dance Dance Revolution, whose usage exploits a dance platform to facilitate the mobility of the lower limbs [34]. Many of

these games propose personalized training schedules that are customizable according to height, weight, hearth rate, training history and goals. To increase the motivation of the player, scoring mechanisms are usually implemented as would be in a classic computer game; however, the score can be based on monitored accuracy or effort, thus reinforcing the correct gesture and perseverance.

This trend has been noticed and adopted even by important game brands and the Nintendo Wii is currently the most popular example of tool enabling physical interactivity between the player and the game console. In fact, the game scenario includes on-screen dynamics reproducing gestures and postures of players according to remote control parameters of position, orientation and acceleration [35], [36]. Recently, even Microsoft has demonstrated to have perceived the potential enclosed in new human-machine interfaces, initially with project Natal, then with the commercial success of Kinect for Xbox [37]. The game market is clearly ready to allow programmers and doctors to develop serious games devoted to the rehabilitation of patients.

III. The DroidGlove Application

The DroidGlove application has been developed using the Android platform for two main reasons: easiness of development and prospective market penetration. Applications running on the Android operating system are developed in Java, which exposes a much quicker learning curve when compared to other platforms using C++ or objective-C. Moreover, market analysts envision a wide spreading for Android-based devices in the next few years; given the low target price, when compared to other brands, this platform is likely to be adopted by a huge number of people.

Despite the low target price mentioned above, Android smartphones (from the HTC G1 model to more recent ones) are equipped with an interesting set of sensors: not just accelerometers, but also a compass and a magnetic field detector, which are not available on other last generation smartphones (like the first generation of Apple iPhone).

A. A DroidGlove Exergaming Session

Usually, a patient in need to rehabilitate her/his wrist goes to the doctor by appointment, gets the wrist checked, and the doctor proposes her/him one or more therapeutic exercises to carry on at home, in order to recover. These exercises have to be easy and quick because the patient will do them unattended, reducing slightly the recovery rate. It is up to the patient to do the exercises and to perform them correctly; the doctor has no way to check how often and how good the patient does the homeworks. On the next visit, the therapist will only rely on the new observed conditions.

With DroidGlove, the patient has to go to the doctor carrying along her/his smartphone. After the visit, the doctor shows to the patient a rehabilitation exercise that she/he will have to repeat in the next days (Picture A in Fig. 1). This exercise involves wrist movements that have to be performed while holding the smartphone. The motion sequence identifying

rehabilitation movements (i.e., the exergame) can be precisely tracked thanks to sensors present in current smartphones and then the exercise is loaded into the phone memory. To save the exergame's track of movements the doctor can use a regular PC or record the right movements on the spot by just moving the device as the patient will be assigned to do. A video can be uploaded on the smartphone for the patient to see again the right sequence of movements embodying her/his personalized gametherapy.

When at home, or wherever she/he likes, the patient can switch on the smartphone, run the DroidGlove application, see the video, and perform the exergame (Picture B in Fig. 1). The application will take care of automatically recording when and how often the exergame has been executed, and how close to the right (saved) sequence of movements the patient performs. A vibration feedback will help the patient in understanding when her/his movements have gone too far from the correct track for that exergame.

On the next appointment, beside the usual physical inspection, the doctor will be also able to check the logs in the device so as to supervise the gametherapy (Picture C in Fig. 1). Information about performed exercises will be available only to the doctor, by authentication, whereas the patient will not be able to see (or tamper with) the data.
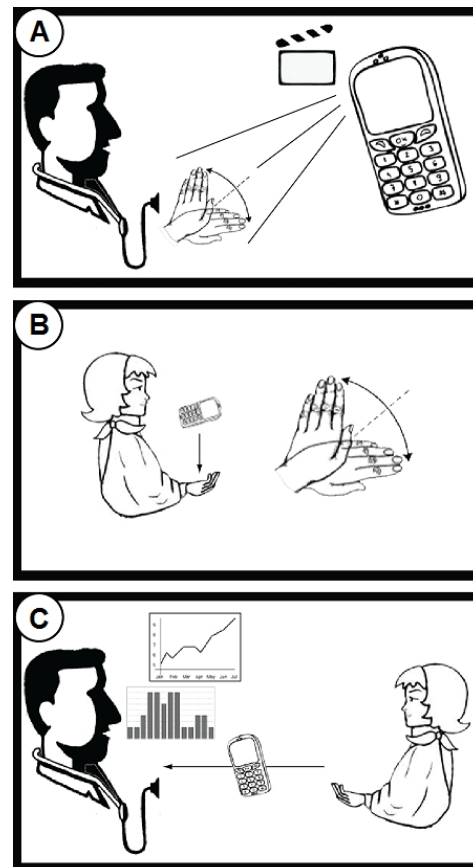

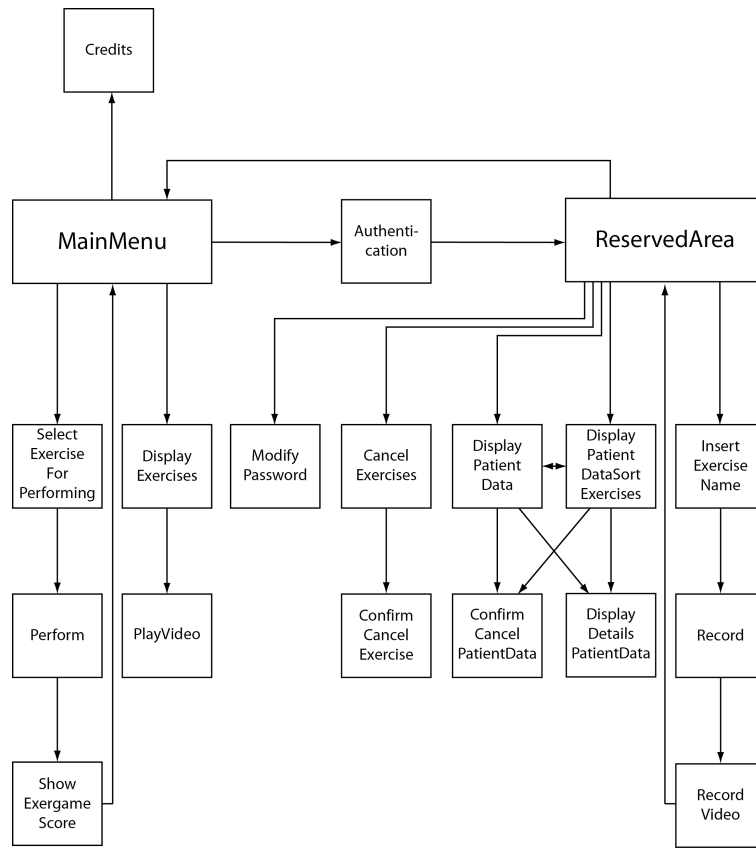Fig. 1. Action sequence in DroidGlove use.

Fig. 2. DroidGlove's activity diagram.

B. Execution Speed Control

DroidGlove verifies the correctness of a performed exercise by verifying its compliance to a pre-stored motion track. More in detail, the motion track is divided into successive steps, clearly delimited by checkpoints. Ensuring that the patient reaches every checkpoint of the motion track is as important as granting execution precision during the exergaming phase. Sudden movements can undermine the correct progress of rehabilitation, so it is important to implement a system able to notice if the user is moving too fast. In that case, the recommended approach from physicians is to stop the exercise and ask the patient to restart it from the beginning. The accelerometer sensors give us an easy way to supervise these behaviors: the application implements a threshold that has not to be crossed. If the acceleration variation is above the threshold for three consecutive samples we assume the patient is actually performing an excessively abrupt gesture. The exergame must be consequently stopped and the user warned.

C. General Architecture

The Android platform defines an application in terms of different activities made available to the user. Each activity defines its own shape for the user interface and reacts to inputs from the user performing specific functions; building an application by means of activities has the advantage to increase modularity and to keep the design simpler.

The application alternates through activities by means of asynchronous messages called intents. Intents may be raised only by the current running activity.

As depicted in Fig. 2, our DroidGlove application provides users (both patients and physiatrists) with many different activities. For the sake of clairty, we describe the main ones in the following list.

- **Main menu**: available at startup to select the next action. A screenshot of this activity is presented in Fig. 3.
- **Motion recording**: turns on, calibrates, and reads data from sensors. This activity is responsible to record an exergame: it retrieves motion information from sensors and stores it on the internal memory.
- **Motion performing**: is the same as the motion recording activity, but data coming from the sensors are validated against stored values to check if the patient is performing the exercise correctly.
- **Video capturing**: uses the camera to record a video; the doctor can record a video to her/himself performing the exercise to better explain the movements to the patient.
- **Video playing**: plays out a video recorded by the video capturing activity.
- **Data managing**: manages stored exercises, retrieves usage statistics, and authenticates the doctor in order to provide access to a reserved area. A screenshot of the user interface in the reserved area can be seen in Fig. 4.

Fig. 3. Screenshot of main menu activity interface.

Fig. 4. Screenshot of data managing activity interface; smartphone in landscape mode.

One of the main problems we faced during the development of our DroidGlove is related to a feature of the Android operating system: if an interactive application becomes irresponsive or overloads the CPU, the system takes over and gives the user the possibility to stop the program. Since data collection and comparison have to be performed in real-time, to avoid interruption of an exercise a multithreading architecture has been adopted. The main thread is connected to the current activity; when sensor calibration, data collection, or storage access is required they are executed as a separate thread; threads not managing user interaction will not be interrupted by the system.

IV. Technical Issues and Adopted Solutions

In this session we outline the major issues we addressed during the development of DroidGlove.

A. Sensors Management

An efficient sensor management is of paramount importance in motion recording and to check the correctness of a performed activity.

Fig. 5. An experimental session with DroidGlove implemented on a HTC G1 smartphone: debug information about acceleration and orientation is reported in real-time on the screen.

During the development of DroidGlove on a real HTC G1 we have realized that acceleration values are not extremely important for this kind of application. Obviously, we are required to prevent the user from performing an exercise too quickly, but physicians we interviewed told us that the extension of each movement is way more important than its speed; the exercise should preferably be performed slowly, but the actual speed is basically up to the patient. Therefore, our application collects data from accelerometers to invalidate the exercise only if the values are beyond a certain threshold. Acceleration information does not, currently, get stored or validated as part of the exercise. Figure 5 shows a test session with a real device: an exercise is recorded and executed in order to tune working parameters and thresholds.

Energy consumption is obviously a crucial issue: when writing software for a mobile device, it is vital to create an energy-savvy application. Using motion sensors amplifies greatly this problem: we had to plan carefully when switching them on or off in order to save battery life.

Another fundamental issue we faced has been related to calibration: data from sensors proved to be sometime unstable, leading to uncertain results. Moreover, the user should always be able to play his exergame, in every moment and situation she/he may be. With these considerations in mind, we have developed a sensor calibration procedure which allows to reduce errors coming from sensors being too sensitive (this sensitivity may also change from device to device) and smooth environmental influences such as, for instance, having the user standing on an uneven ground.

While working with rotation-related data, it is important to consider the cardinal references in the tri-dimensional space relative to the user's starting position. With pitch and roll this problem is significantly less important, as it is dubious that the user will decide, for instance, to perform her/his exergame upside down. The same cannot be said for yaw: most likely, the user will not be facing the same direction toward which the exercise has been recorded. Therefore, yaw

data are calculated starting from a fictitious cardinal point, that will be exactly in front of the doctor or the patient when they are about to start playing. This cardinal point, which we call *fake-north*, will become the relative *0* for yaw-related calculus. However, this computation must handle hardware limitations of the device on which is running, especially those related to the calibration delay of sensors that collect data. When those sensors are switched on, Android activates an internal calibration procedure leading the gyroscope to calculate its own data accordingly to the position of horizon-oriented axes. This means that we have to wait for a variable period of time, while sensors produce slightly-significant data, that firstly drift around the zero (the absolute north in yaw calculus) by infinitesimal values, till achieving values between zero and the actual orientation of the device. Calibration handling becomes extremely significant if connected to the fake-north concept, as a correct rotation measurement passes through having the fake-north as precise as possible.

In order to appropriately compute the fake-north, our application proceeds as follows: the user touches the screen when she/is is ready to start the exergame; this happens in both recording and executing mode. During calibration a thread takes care of the user interface and has the user waiting while another specific thread activates the sensors and starts collecting data used for calibration. Initially, all (infinitesimal) values are discarded; then, measurements are taken in groups of three and the average is calculated. In our empirical model we consider the hardware calibration procedure over when the difference between each of the three samples and their average is below a certain threshold. By practical experimentation we are currently using a threshold of 5 degrees, which seems to be a good compromise between the measurement errors and convergence time.

When the hardware calibration is over, current average becomes the value for the fake-north. All relative yaw values during recording and exercises will be calculated from the absolute values using the following formula:

$$rel\_yaw = (abs\_yaw - fake\_north + 360)\,mod\,360$$

This formula keeps the degrees value of yaw in $[0, 360)$ even when crossing border values.

B. Raw Data Filtering

During experiments we observed that measurement noise produced by sensors was a severe problem for exercise recording: if the reading is too unstable, it will be virtually impossible for the patient to perform the exercise correctly. To solve this problem we had to filter samples coming from the sensors in order to take into account, and record, only the most significant ones.

As anticipated, we used a real HTC G1 as a test device; results showed that even when the device was placed on a hard surface, thus in a situation where motion noise should be irrelevant, sensors were actually reporting drifting phenomena of even 4 or 5 degrees, in particular for yaw values.
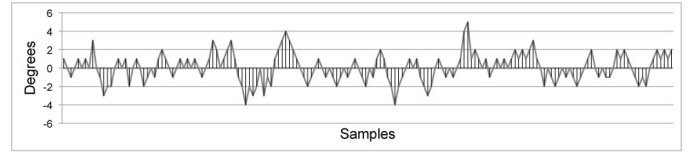

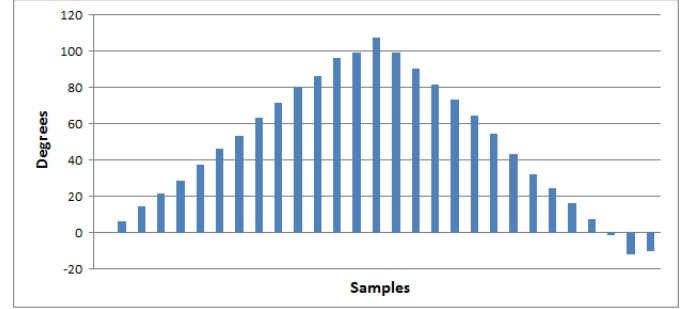Fig. 6. Yaw skew in an HTC G1 smartphone motionless and on a hard surface.

Fig. 7. Filtered yaw variation of real wrist rotation from 0 to 90 degrees and back.

To this aim, Fig. 6 shows yaw noise while keeping the smartphone motionless for about 10 seconds.

This behavior is accentuated by the not complete immobility of the user hand during calibration: in this case the variation can reach drifting values of 6 or 7 degrees. After some tests we converged to define some minimum thresholds for rotation on every axis; below these values the device will consider the information as a noise, and will not keep trace of the perceived rotation. Current thresholds are still in a testing phase; we set them to 8 degrees for yaw, 5 degrees for pitch, and 5 degrees for roll. With these settings, if the device is kept motionless in the hands of an average user, it will not record any movement.

To implement this policy the application compares data coming from the sensor with the current position. If the difference exceeds one or more thresholds, the previous values will be updated with new ones; otherwise, the incoming information is discarded. Checking the threshold for pitch and roll is quite simple, since they assume values in [-180, 180] degrees. On the other hand, yaw may be problematic due to the passage from 360 to 0 degrees. We compute yaw variation using the following formula:

$$gap = |previous\_yaw - new\_yaw|$$

$$\Delta yaw = min\,(gap, 360 - gap)$$

This approach allows us to obtain a consistent result even if the movement takes place very close to the fake-north position. To this aim, Fig. 7 shows filtered yaw values for a rotation from 0 to 90 degrees and back performed by an healthy user.

C. Tracking Exercises and Evaluation of Skill Level

The setting of a sampling threshold for data collection, as described before, allows for a discretization of the exercise motion track. This track can be described, after the sampling,

as a series of checkpoints in a tridimensional vector space. We can use this abstraction together with the geometric notion of distance during the exergaming phase to determine the advancement in the exercise track and the digression from the checkpoints, so as to measure the skill level of a patient.

Indeed, in DroidGlove we have implemented an algorithm that evaluates the patient skill level based on the distance between her/his actual movement track and the correct one. Around each checkpoint we identify a *valid area* based on a tolerance vector defined by three predetermined thresholds. These three thresholds, related respectively to yaw, pitch, and roll, must be chosen wisely: they will represent the maximum distance from the checkpoint where the performed movement will be considered correct.

At this point, we could use the Euclidean formula to compute the distance:

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2}$$

The problem here is that the above calculation has to be performed continuously and in real-time; which is, many times every second. As a consequence, we would have to force the device to onerous computations on a fairly long time-span, with severe impact on CPU usage and energy consumption. For this reason we have preferred to use the Manhattan formula for distance, which is much less CPU demanding.

$$d(x, y) = |x_1 - y_1| + |x_2 - y_2| + |x_3 - y_3|$$

Unluckily, to check if a movement remains inside the valid area we cannot use the Manhattan distance directly: as already discussed, the yaw value has a different tolerance vector than pitch and roll. Instead, a better solution is to compute the distance separately for every vector component and compare it with the related threshold.

$$|x_i - y_i| \leq \sigma_i \qquad \forall i \in \{1, 2, 3\}$$

Where σ_i is the threshold for component i.

When this check fails the smartphone vibrates and the user is aware of her/his poor performance. If the user keeps the movement outside a valid area for 5 seconds in a row, then the movement is assumed to be completely wrong, the exercise is stopped, and the patient has to restart from the beginning.

The next step is to figure out a way to understand at what point of the exercise (at which checkpoint) the patient has arrived. Assuming the patient to be at checkpoint j, we compute the (Manhattan) distance between the current measurement and checkpoints j, $j + 1$, and $j + 2$. If one of the last two checkpoints is closer than j to the current position, it means that the patient is actually proceeding along the exercise track. Consequently, the "current checkpoint" will be associated to the closest checkpoint, whereas the previous one will not be used anymore in the distance calculus.

This checkpoint system resolves, among the others, problems related to different performing speeds: as a matter of fact, we cannot expect that patient and doctor were able to perform the movement track with identical speed. Instead, we can expect that more than one point on the patient's factual movement track correspond to a single doctor checkpoint. In this case, every value must be considered correct if still in the tolerance area. Furthermore, a patient may skip a checkpoint if proceeding too fast; yet, this should not invalidate the whole exercise (granted that the acceleration limit is respected) but just lower the final score.

One of the most important feature of our DroidGlove is the ability to convert a rehabilitation activity into an entertaining challenge; hence, assigning a score as a consequence of the exercise will make the user perceive the therapy as a game and push her/him to constantly improve the performance. The patient could be motivated by the possibility to share with friends her/his progresses demonstrated by increasing scores achieved after performing the exergames. The higher the score becomes, the better her/his physical conditions are likely to be. In point of this, first, we need to define a policy to assign a score at each checkpoint. We decided to associate 33 points to each distance, which means 99 (100) points per checkpoint when the movement is perfect, and to calculate the score at checkpoint j, through the following formula:

$$score_j = 1 + \sum_{i=1}^{3} \left(\frac{\sigma_i - d(x_i, y_i)}{\sigma_i} \times 33 \right)$$

Since it is likely the patient will produce several movements and hence measurements in proximity of every checkpoint, we can expect that, over the time, the patient will increase her/his score by getting closer to the current checkpoint. As a consequence, for each checkpoint we should store only the higher score value. At the end of every exercise the total score will be displayed and the user can get a feeling of her/his improvements.

The last problem to face is the management of the exergame ending. We can expect a variety of situations: different scorings, exercise errors, and unfinished exercises. The program will react to those three different scenarios as follows:

- in case of severe movement errors, the exercise is interrupted and the user invited to retry;
- if the patient has been able to complete the exercise, the exergame is considered valid;
- if the patient has not been able to complete the exercise, the score will be calculated only on the checkpoints she/he was able to reach, but the doctor will be notified of the voluntary interruption.

D. Datastore Management

Program specifications require to store almost all the data collected on the internal non-volatile memory. As anticipated, among these data there are spatial values about recorded exercises and patient's performance statistics. Android provides an SQLite [38] interface to easily store and retrieve structured information. Access to the DroidGlove database is granted only to activities and classes which are owned by the application itself in order to protect data from malicious or unintended modifications.

Since the data size and the query computational burden are not predictable, we have decided to have a separated thread handling the database. Whenever the user wants to perform an exercise, the thread retrieves from the database the track-related information and load it into main memory. In a similar way the recording is performed in memory and, afterward, data are dumped into the database. This kind of approach is memory-consuming and poses a technical limitation to the size of an exercise; despite this, it also solves problems related to real-time access to non-volatile memory, typically much slower than the main memory.

V. Conclusions and Future Work

Games have recently demonstrated how useful they can be in solving serious problems. In this context, an interesting field of research is certainly embodied by healthcare. This paper illustrates an original application we have created that combines serious gaming, healthcare, and smartphones to create a digital tool for wrist rehabilitation, namely DroidGlove.

Based on the Android platform for mobile phones, Droid-Glove has the innovative advantage of allowing ubiquitous gametherapy, with also the possibility for the doctor to verify the exercises performed at home by the patient. Our presentation included the analysis of technical issues, adopted solutions, and the implementation on a real HTC G1 smartphone.

Currently we are working on refining the evaluation algorithms in order to exclude some false positive recognitions. Moreover, network support could be introduced to centralize data collection or to create a public leaderboard to induce patients in improving the quality of their exercise.

References

[1] Roccetti, M., Marfia, G., Zanichelli, M., "The Art and Craft of Making the Tortellino: Playing with a Digital Gesture Recognizer for Preparing Pasta Culinary Recipes," *ACM Computers in Entertainment, ACM*, vol. 8, no. 4, 2010.

[2] Roccetti, M., Marfia, G., "Recognizing Intuitive Pre-defined Gestures for Cultural Specific Interactions: An Image-based Approach," in *3rd IEEE DENVECT'11 - IEEE CCNC 2011*, 2011.

[3] Palazzi, C. E., Maggiorini, D., "From Playgrounds to Smartphones: Mobile Evolution of a Kids Game," in *IEEE CCNC 2011*, 2011.

[4] Myers, D., "Simulation as play: A semiotic analysis," *Simulation and Gaming: An International Journal*, vol. 30, no. 2, pp. 147–162, 1999.

[5] Michael, D., Chen, S., *Serious Games: Games That Educate, Train, and Inform.* Course Technology, 2005.

[6] Gifford, B. R., "The learning society: Serious play," Chronicle of Higher Education, 7.

[7] Susi, T., Johannesson, M., Backlund, P., "Serious games – an overview," Technical Report HS-IKI-TR-07-001, School of Humanities and Informatics, University of Skovde, Sweden, Feb. 2007.

[8] Palazzi, C. E., Roccetti, M., Marfia, G., "Realizing the Unexploited Potential of Games on Serious Challenges," *ACM Computers in Entertainment, ACM*, vol. 8, no. 4, 2010.

[9] Google, "Image Labeler," http://images.google.com/imagelabeler/.

[10] Watters, C., Oore, S., Shepherd, M., Abouzied, A., Cox, A., Kellar, M., Kharrazi, H., Liu., F., Otley, A., "Extending the Use of Games in Health Care," in *HICSS*, 2006.

[11] Jack, D., Boian, R., Merians, A., Tremaine, M., Burdea, G., Adamovich, S., Recce, M., Poizner, H., "Virtual reality-enhanced stroke rehabilitation," *IEEE Transactions on Neurological System and Rehabilitation Engineering*, vol. 9, pp. 308–318, 2001.

[12] Broeren, J., Georgsson, M., Rydmark, M., Stibrant Sunnerhagen, K., "The Rehabilitation Gaming System: A Virtual Reality Based System for the Evaluation and Rehabilitation of Motor Deficits," in 4th *ICDVRAT*, 2002.

[13] Prisco, G. M., Avizzano, C. A., Calcara, M., Ciancio, S., Pinna, S., Bergamasco, M., "Virtual Environment with Haptic Feedback for the Treatment of Motor Dexterity Disabilities," in *IEEE ICRA*, 1998.

[14] Android consortium, "Official website," http://www.android.com/.

[15] Deponti, D., Maggiorini, D., Palazzi, C. E., "DroidGlove: An Android-Based Application for Wrist Rehabilitation," in *IEEE UMSA'09 - IEEE ICUMT 2009*, 2009.

[16] Nintendo, "Power glove controller," http://en.wikipedia.org/wiki/Power_Glove.

[17] Michael, D., Chen, S., *Serious Games: Games that Educate, Train, and Inform.* Boston, MA.: Thomson Course Technology, 2006.

[18] The National Institutes of Health (NIH), "Hungry Red Planet," http://www.healthmedialab.com/html/products.html.

[19] Archimage Inc., "Arch Image Online," http://www.archimageonline.com/.

[20] BreakAway Games, "Healthcare Holutions," http://www.breakawaygames.com/serious-games/solutions/healthcare/.

[21] Gifford, B. R., *The learning society: Serious Play*.

[22] "Full Spectrum Warrior," http://www.fullspectrumwarrior.com.

[23] Emmelkamp, P. M., Bruynzeel, M., Drost, L., Van Der Mast, C. A., "Virtual Reality Treatment in Acrophobia: a Comparison with Exposure in Vivo," *CyberPsychology and Behavior*, vol. 4, no. 3, pp. 335–339, 2001.

[24] Botella, C. M., Juan, M. C., Banos, R. M., Alcaniz, M., Guillen, V., Rey., B, "Mixing Realities? An Application of Augmented Reality for the Treatment of Cockroach Phobia," *CyberPsychology and Behavior*, vol. 8, no. 2, pp. 162–171, 2005.

[25] Wiederhold, B. K., Wiederhold, M. D., "Threeyear Follow-up for Virtual Reality Exposure for Fear of Flying," *CyberPsychology and Behavior*, vol. 6, no. 4, pp. 441–446, 2003.

[26] Riva, G., Bacchetta, M., Cesa, G., Conti, S., Molinari, E., *The use of VR in the treatment of Eating Disorders.* Ios Press, 2004.

[27] Mitchell, A., Savill-Smith, C., *The Use of Computer and Video Games for Learning: A Review of the Literature*.

[28] Lindt, I., Ohlenburg, J., Pankoke-Babatz, U., Ghellal, S., Oppermann, L., Adams, M., "Designing Cross Media Games," in 2nd *International Workshop on Gaming Applications in Pervasive Computing Environments*, 2005.

[29] Benford, S., Magerkurth, C., Ljungstrand, P, "Bridging the Physical and Digital in Pervasive Gaming," *Communications of the ACM*, vol. 48, no. 3, pp. 54–57, 2005.

[30] Bonanni, L., Vaucelle, C., Lieberman, J., Zuckerman, O., "TapTap: A Haptic Wearable for Asynchronous Distributed Touch Therapy," in *ACM CHI '06*, 2006.

[31] Gourlay, D., Lun, K. C., Liya, G., "Virtual Reality and Telemedicine for Home Health Care," *Computers and Graphics*, vol. 24, pp. 695–699, 2000.

[32] Cameirao, M. S., Bermdez i Badia, S., Duarte Oller, E., Zimmerli, L., Verschure, P. F. M. J., "The Rehabilitation Gaming System: A Virtual Reality Based System for the Evaluation and Rehabilitation of Motor Deficits," in *Virtual Rehabilitation*, 2007.

[33] Lumbreras, M., Sanchez, J., "Usability and Cognitive Impact of the Interaction with 3D Virtual Interactive Acoustic Environments by Blind Children," in 3rd *ICDVRAT*, 2000.

[34] De Maria, R., " Games for Health 2006: Dance dance...Revolution in Fitness!" http://www.seriousgamessource.com/features/feature_051906.php.

[35] Mäyrä, F., "The Contextual Game Experience: On the Socio-Cultural Contexts for Meaning in Digital Play," in *DiGRA 2007 Situated Play Conference*, 2007.

[36] Pearson, P., Bailey, C., "Evaluating the Potential of the Nintendo Wii to Support Disabled Students in Education," in *Ascilite*, 2007.

[37] Microsoft, "Xbox – Kinect," http://www.xbox.com/kinect/.

[38] SQLite, "Official website," http://www.sqlite.org/.