Enter the ROBiGAME: Serious game for stroke patients with upper limbs rehabilitation

François Rocca[1], Pierre-Henri De Deken[1], Alessandra Bandrabur[2], and Matei Mancas[1]

[1] Numediart Institute, University of Mons, Belgium
Place du Parc 20, Mons, Belgium
surname.name@umons.ac.be
[2] IPAL, University Politehnica of Bucharest, Romania

Abstract. The objective of the project is to develop an intelligent serious game for rehabilitation of the upper limbs for stroke patients using an interactive rehabilitation robot. The robot screenplay adapts to the patient's functional abilities and the robot mechanical assistance evolves according to patients motivational, motor and cognitive performances when playing a serious video game. This work will be developed on the REAPlan robot providing a distal effector which can mobilize the patient upper limb(s) in a horizontal plane [1] but could also be transferred to other robots or even simpler rehabilitation setups.

Keywords: serious game, strokes, rehabilitation, user tracking, head position, facial expression.

1 Introduction

Thanks to advances in basic and clinical research, neurological rehabilitation knowledge has greatly developed in recent years [2] and [3]. During the rehabilitation, to learn a task, the exercises in brain-injured patients are improved by the use of rehabilitation robotics that achieves intensity (number of movements by time) higher than the conventional rehabilitation([4] and [5]). One of the possible optimization of this technique is the implementation of serious games with robots that would combine the benefits of rehabilitation robotics and those from the serious games mainly in terms of patient motivation. Indeed, they have already proved their specific interest in the adult brain-injured patient [6]. The "Enter the ROBIGAME" project can be broken down into four components: medical, robot, video games and analysis of motivation.

This paper is structured as follows. Section 2 provides information about related systems and work, section 3 gives details about the rehabilitation system used in this project. Section 4 gives information about Serious gaming using the game engine Unity. The next section 5 shows how we do user tracking and analysis in three parts: the first one explains the body face tracking, the second one facial action coding system theory and the third explain how we use a machine learning toolkit for expression recognition. Section 6 describes the whole setup in this project and relates the results of the experiment. Finally we conclude in section 7.

2 Related works

For the hemiplegia, the most popular and most effective method of rehabilitation is to restore function rather than offsetting the deficit [7]. Different restoration protocols have been developed to relearn both a specific skill than the general activities of daily living [8]. Research in neuroscience shows that relearning protocols result in positive changes in the structure and activity of the brain ([9] and [10]). Different technologies have emerged in recent years to assist doctors and patients during therapy. Some of them are linked to serious games in order to perform not abstract tasks.

2.1 Technologies for rehabilitation

While the video game is sometimes seen by the wider public as an entirely fun activity, it has rather great potential for serious activities. Indeed, it can be used as a psychological, cognitive or motor help allowing the player to carry out meaningful tasks. In the medical world, particularly rehabilitation is an area where serious games are becoming more developed ([11] and [12]). For the rehabilitation, different systems use sensors with effectors or not, these systems are generally used with games. Some of them use optical motion system to track the user body (webcam, kinect, etc.), electronic sensors (gyroscope, accelerometer, etc.) or robotic arms. The category of robotic arms for rehabilitation is divided into two families depending on they support or not the entire ”joint chain” located between the patient's trunk and hand. Table 1 gives a summary of some of these systems. In the first case we speak of exoskeletons, while in the second we speak of manipulators distal effector. The most famous robot belonging to this second family is the Manus robot developed by Krebs [13] at the Massachusetts Institute of Technology (MIT) and marketed by Interactive Motion Technologies. Robots REAplan and REA2plan developed at Mechatronics Research Centre (CEREM UCL, Belgium) also belong to this category. The latter two robots are those used in this project.

2.2 Motivation extraction

When performing an exercise, the ROBiGAME tracker process must be able to capture information related to motivation performance of the patient. The patient's motivation will be assessed from two features extracted from patient's behavior. The first one concerns the orientation of the head [14] and this will be extracted using a 3D camera (Microsoft Kinect sensor). The second one concerns the analysis of the patient's emotional state. From the patient's face that can be automatically detected with the Kinect SDK, it is possible to extract emotional parameters such as ”FACS” [15] which are basic units related to the emotions . This would make it possible to know the valence of the face expressions of the patient to determine positive or negative emotions based on expressions and also the degree of energy spent by the patient during the task (neutral, very involved, etc.).

Table 1. Summary of some rehabilitation system with serious game for stroke pathologies

Reference	Function spotted	Technolgies used	Assistance	Type of Serious Game
Burke 2009 [16]	Upper limbs motricity	Webcam + Screen	No	Basic Arcade
Cameiro 2011 [17]	Upper limbs motricity	Webcam + Data gloves + Screen	No	Basic Arcade
Buxbaum 2012 [18]	Hemineglect rehabilitation	Screen	No	Simulation of real task
Kim 2011 [19]	Hemineglect rehabilitation	Webcam + Data gloves + Screen	No	Simulation + Arcade
Katz 2005 [20]	Hemineglect rehabilitation	Screen	No	Simulation of real task/life
Myers 2000 [21]	Hemineglect rehabilitation	VR headset	Yes : Visual +audio	Simulation of real task/life
Klinger 2013 [22]	Hemineglect rehabilitation	Webcam +Screen +Keyboard +Gamepad +Mouse +Micro	No	Simulation of real task/life
Hasomed [23]	Cognitive Function	Screen +Special Keyboard	Yes: Visual	Object spotting
R.O.G.E.R.S. [24]	Cognitive Function	Kinect +Screen	No	Simulation of real task/life
Voracyfish [12]	Upper limbs motricity	Kinect +Screen	No	Arcade

3 Reaplan system

The reaplan system is a robotic device for medical assistance which helps in rehabilitation of adults and children affected by stroke. It was developed by Belgian start-up Axinesis (CEREM UCL) to facilitate the recovery of motor function of the upper limbs.

3.1 Description and usability

Motor deficiencies (strength, joint mobility, dexterity) will be evaluated clinically according to the validated protocols. An evaluation protocol of these deficiencies by the robot will be developed to provide ROBiGAME quantitative and objective data. Force and position sensors will be used to assess:

- Patient force;
- Comprehensive passive and active range of motion by measuring the largest perimeter the subject can perform with or without assistance;
- Kinematics and kinetics of the arm in standardized movements.

The patient's motor performance will be quantified during rehabilitation exercises using the robot's sensors. The reaplan robot is shown in figure 1, on the right part of this figure you can see the distal effector (in blue), the working plane (in green), the screen (in yellow) and the stop button (in red).

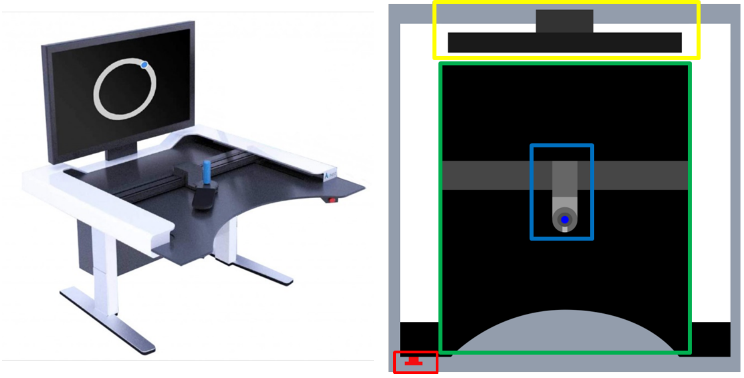


Fig. 1. Reaplan systeme with the distal effector in blue, the working surface in green, the screen in yellow and the emmergency button in red.

3.2 Communication

From the point of view of the connections, the communication between the REAPlan and the computer is done by USB 2.0 which allows to simulate a port COM. For the software part, an interface created by Axinesis allows to manage the exchange of information and commands between games and robot.

Existing interface of the REAPlan. This interface allows the therapists to manage the profiles of the patients. They can also by this way configure the games which will serve during a session of reeducation. During this session, it is easy to verify in live the evolution of the parameters of the patient. At the end of the session, the data and the results are saved for a further analyse and re-used for a later session.

If a new game must be added to the interface, it is necessary to add compiled files in the code of the interface and then to recompile this one in order to test the new game.

Design without this interface. To avoid all this procedure of insertion and by the way to improve the development speed, we decided to work without this interface to set up the first games of tests. The Dynamic Link Library file (DLL) which contains all the functions allowing the communication between the interface and the REAPlan was extracted. This file was directly used by the new games.

However the program used for the creation of these games accepts only DLL files with a version of the framework lower or equal to the .net 3.5. This DLL having been realized with the framework .net 4.0, we had to recompile it. This stage also taking some time, we limited the recompilation to the part of code which allows the control of the REAPlan in free mode, that is the mode where the distal effector brings into conflict no forces to the patient. Therefore, the REAPlan will not assist the players in our first game of tests.

4 Unity

To create our first games, we used to the Unity framework. This software presents a lot of qualities. It is free in its basic version, numerous tutorials can be found and it has an excellent documentation. Furthermore, an online store allows us to obtain and reuse numerous assets. In the 3.2, we mentioned the use of a DLL, Unity allows the integration of new modules and DLLs. All the programming is realized in c# with the environment of development Monodevelop[25].

4.1 Unity description

Unity is a multiplatform game engine and a development platform created by Unity Technologies for designing multiplatform 3D and 2D games and interactive experiences. It is one of the most popular in the gaming industry.

The philosophy in unity is Object-Oriented programming. Any element placed in Unity is a GameObject. A cube with a 3D picture is a GameObject, a camera which gives the point of view of the player is also a GameObject, as well as light sources or sound sources. All the GameObjects have basic properties such as their position, angles of rotation and a size according to axes x, y and z. What differentiates them are the modules which we can bind to these GameObject. For example, the cube will have a module which will allow its visual display. In a game, the various elements can have particular behavior, react to events or simply move. All this is defined in what is called a script. For example, a button will have a module which will bind it to a script which will define what has to occur when the player presses on the button. Finally, the whole level is saved in a file which we call scene. It is therefore possible to have several scenes in a game.

4.2 Design of the games

During our first developments, we succeeded in making communicate a Unity-based game with the REAPlan and to build two versions of this game. The first one is a shooting game in 2D with a top view. Not to waste time by recreating from scratch a quite new game and in a purpose of Unity's handling, we decided to rewrite the first game by changing its assets, a part of its code and its parameters. This second game is called ”A l'abordage”. While the first one was based on spaceships that must destroy asteroids and other enemies, this second one takes place in a maritime environment and implements ships.

Menu and communications We gathered both games in a single program with a selection menu which we can see in figure 2. In the starting up of the main program, two scripts are loaded. The first one initializes the communication towards the REAPlan thanks to the dll. The games of the menu are accessible only if the initialization is finished. In case the initialization failed, the communication towards the REAPlan is cancelled and the games become accessible and playable with the control of the mouse. The second script initializes a OSC communication [26] with another computer which takes care to analyze the gaze and motivation of the patient (see section 5.1). These information allow us to edit in real time the level of the difficulty of our game in order to adapt it to the patient's behavior.

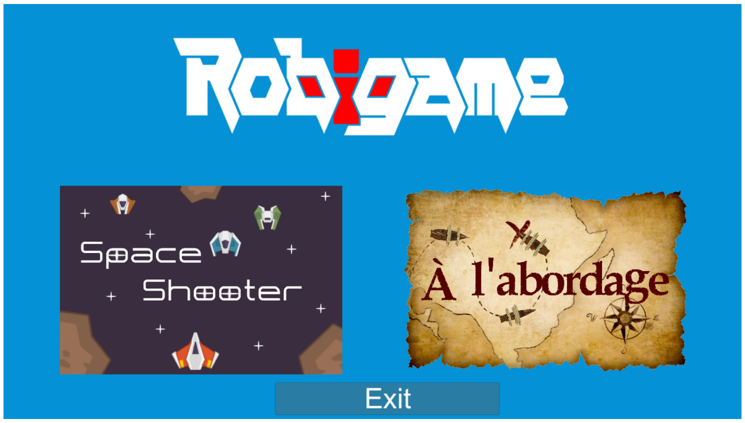


Fig. 2. The menu to select the game.

Games and interface with REAPlan. Our choice went towards the shooting game in the space for two main reasons. The first one is that the free movement is the only accessible mode for the REAPlan at the moment (see section 3.2), this mode represents the physics in a spatial environment. The second is that the game takes place in a 2D world what corresponds to the movements of control which are allowed by the REAPlan.

To control correctly the game with the REAPlan, a mapping must be realized between the zone of movement of the effector distal (left image of the figure 3) and the play area (right image of the figure 3). We use formule (1) and (2) to reach our purpose.

As it was previously mentioned at the beginning of this section, the second game is a rewriting of the first one with changes of the assets and the parameters. The figure 4 lists the correspondences between both games and specify the parameters used according to the difficulty of the game.

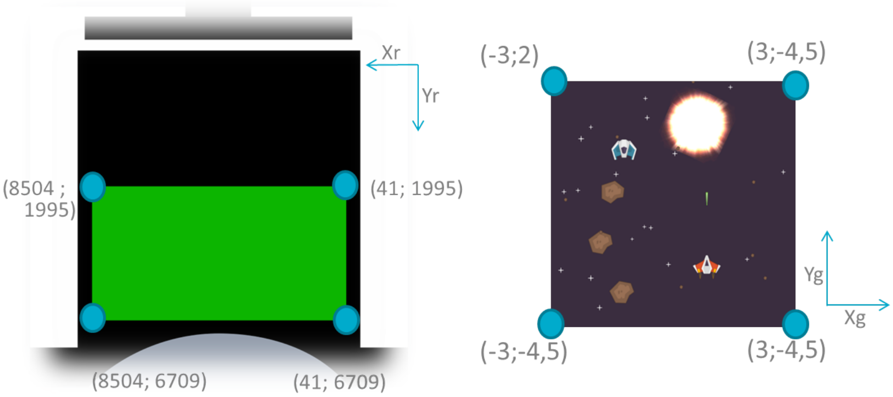


Fig. 3. Working plan for rehabilitation to gaming coordinate.

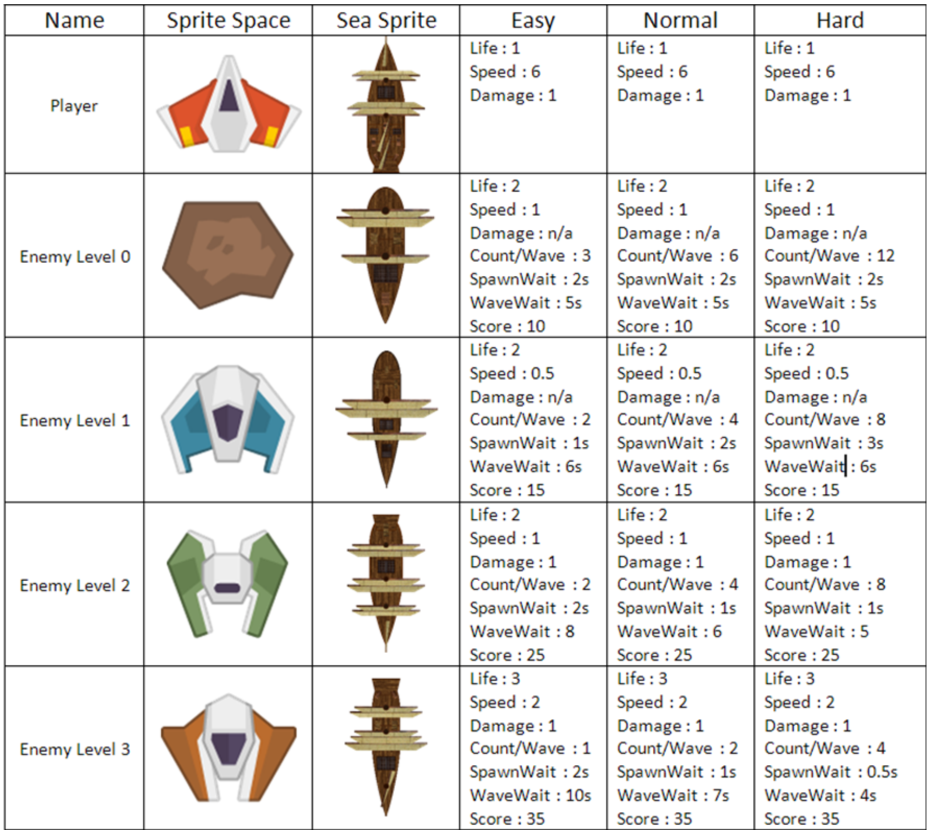
$$Xg = \left[- \left(\frac{Xr - 41}{8504 - 41} \right) * 6 \right] + 3 \tag{1}$$

$$Yg = \left[- \left(\frac{Yr - 1995}{6709 - 1995} \right) * 6.5 \right] + 2 \tag{2}$$

5 Body and face Analysis

5.1 User body and smile detection

For the user body and face tracking, we use the Kinect V2 sensor with Microsoft Kinect SDK. The main use of the Kinect is the user skeleton tracking. Skeletal



Name	Sprite Space	Sea Sprite	Easy	Normal	Hard
Player			Life : 1 Speed : 6 Damage : 1	Life : 1 Speed : 6 Damage : 1	Life : 1 Speed : 6 Damage : 1
Enemy Level 0			Life : 2 Speed : 1 Damage : n/a Count/Wave : 3 SpawnWait : 2s WaveWait : 5s Score : 10	Life : 2 Speed : 1 Damage : n/a Count/Wave : 6 SpawnWait : 2s WaveWait : 5s Score : 10	Life : 2 Speed : 1 Damage : n/a Count/Wave : 12 SpawnWait : 2s WaveWait : 5s Score : 10
Enemy Level 1			Life : 2 Speed : 0.5 Damage : n/a Count/Wave : 2 SpawnWait : 1s WaveWait : 6s Score : 15	Life : 2 Speed : 0.5 Damage : n/a Count/Wave : 4 SpawnWait : 2s WaveWait : 6s Score : 15	Life : 2 Speed : 0.5 Damage : n/a Count/Wave : 8 SpawnWait : 3s WaveWait : 6s Score : 15
Enemy Level 2			Life : 2 Speed : 1 Damage : 1 Count/Wave : 2 SpawnWait : 2s WaveWait : 8 Score : 25	Life : 2 Speed : 1 Damage : 1 Count/Wave : 4 SpawnWait : 1s WaveWait : 6 Score : 25	Life : 2 Speed : 1 Damage : 1 Count/Wave : 8 SpawnWait : 1s WaveWait : 5 Score : 25
Enemy Level 3			Life : 3 Speed : 2 Damage : 1 Count/Wave : 1 SpawnWait : 2s WaveWait : 10s Score : 35	Life : 3 Speed : 2 Damage : 1 Count/Wave : 2 SpawnWait : 1s WaveWait : 7s Score : 35	Life : 3 Speed : 2 Damage : 1 Count/Wave : 4 SpawnWait : 0.5s WaveWait : 4s Score : 35

Fig. 4. Asset comparisons with games values.

tracking is able to recognize users sitting on the REAPlan. To be correctly tracked, users need to be in front of the sensor, making sure their head and upper body are visible (see Figure 5). The tracking quality may be affected by the image quality of these input frames (that is, darker or fuzzier frames track worse than brighter or sharp frames). The Kinect sensor contains two CMOS sensors, one for the RGB image (1920 x 1080 pixels) and another for the infrared image (512 x 424 pixels) from which the depth map is calculated. The technology behind the sensor is infrared TOF (Time Of Flight). This sensor measures the time it takes for pulses of laser light to travel from the laser projector to a target surface, and then to come back to an image sensor. Based on this measure, the sensor gives a depth map.

To achieve head pose, at least the upper part of the user's KinectV2 skeleton has to be tracked in order to identify the position of the head. The position of the head is located using the head pivot from the 3D skeleton only on the depth map. The head pose estimation is based on the face tracking and it is achieved on the color images. Consequently, the face tracking is dependent on the

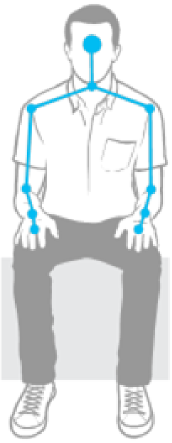


Fig. 5. User tracking in seated mode

light conditions, even if KinectV2 is more stable into dark light conditions than KinectV1. The head pose estimation method returns the Euler rotation angles in degrees for the pitch, roll and yaw as described in Figure 6, and the head position in meters relatively to the sensor being the origin for the coordinates. Based on the head pose estimation, it is possible to know where the user is looking on the screens. It's also possible to know how long they spend watching a part of the screen. Different duration are used to describe the level of attention.

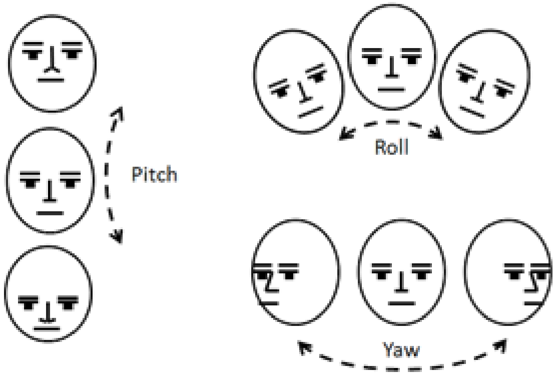


Fig. 6. Three different degrees of freedom of the head [27]

Based on the face analyze from the Kinect SDK, we extract: the neutral position of the mouth, brows, eyes, and so on. The Action Units (AU) represent the difference between the actual user face and the neutral face. Each AU is expressed as a weight between -1 and +1. Some basic functions are used to measure the smile (Figure 7) and its possible to determine if user is smiling, is not smiling or is maybe smiling.

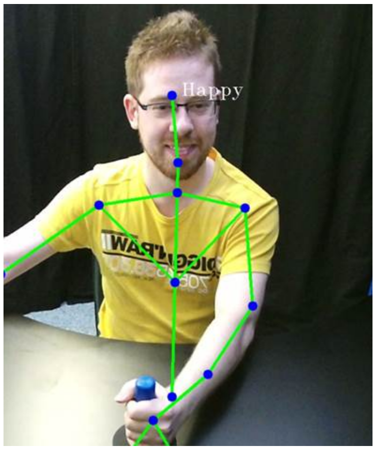


Fig. 7. User tracking with an happy face during the therapy.

5.2 Facial Action Coding System

The most used system for description of facial expressions was proposed by Ekman and Friesen in [15] and reviewed in [28] and its name is FACS, Facial Action Coding System. The FACS is based on the universality of emotions, namely there are a set of basic emotions which are recognized by people from all cultures. In his first work, Ekman et al [15] proved that there are six basic emotions: happiness, sadness, fear, anger, disgust and surprise. According to FACS, the muscles of a face are able to produce 46 Action Units (AU), where different combinations of AUs form the basic emotions.

An AU is a basic facial action and has three phases onset, apex and offset. This temporal aspect of the AUs can be seen in Figure 8. Facial temporal dynamics are used for recognition of different psychological states as pain [29] or continous affective states [30]. A set of rules that defines AUs and corresponding emotions is shown in Table Table 2 and 3 .

5.3 The Wekinator

Definition. The Wekinator is a framework for real-time machine-learning, created by Rebecca Fiebrink et al[32]. It is an open-source software application based on Weka framework[33] and it allows one to train and modify machine-learning algorithms in real-time. The application maps the user inputs to specific parameters of synthesis during the training stages.

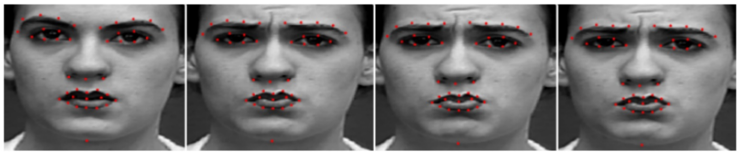


Fig. 8. Temporal modification of facial expression from neutral to apex. Image sequence taken from Cohn-Kanade+ database [31]

AU Code	Description
AU1	Inner Brow Raiser
AU2	Outer Brow Raiser
AU4	Brow Lowerer
AU5	Upper Lid Raiser
AU6	Cheek Raiser
AU7	Lid Tightener
AU12	Lip Corner Puller
AU15	Lip Corner Depressor
AU16	Lower Lip Depressor
AU20	Lip stretcher
AU23	Lip Tightener
AU25	Lips part
AU26	Jaw Drop

Table 2. AUs Description

The datasets consist in features from video, audio or 3D data and they are used to train the learning algorithm responding to this specific inputs. This framework is compatible with Weka, therefore any dataset and classifier can be exported to Weka, and any classifier trained in Weka can be run within Wekinator.

The GUI provides several options for selecting the appropriate learning algorithm that fits a specific problem. Some of these algorithms are SVMs, AdaBoost, decision trees and k-nearest neighbor. The Wekinator provides also an option to modify the training data during the process execution, therefore one can re-train data in real time.

Emotion	AUs Used
Sadness	AU1, AU4, AU15
Happiness	AU12, AU6
Fear	AU1, AU2, AU4, AU5, AU20, AU26
Surprise	AU1, AU2, AU5, AU26
Anger	AU4, AU5, AU7, AU23

Table 3. Recognizing facial expressions through AUs

The real-time user interaction with a complete learning process, allows a play-along mapping for generating the training dataset while the computer performs the actual task. The Wekinator was originally conceived to create controller mappings for sound synthesis, which provide to the subjective and unique user musical expression. Its purpose was to create an user-centered application, in order to promote real-time exploration of synthesis algorithms into compositional performances.

Rebecca Fiebrink has proven that using a small training dataset, created in a short period of time, could be enough for a personal performance. The training task is a data entry task, but in the same time a musical one as well, because the performer supervises the sounds that he hears during the learning through specific gestures. Thence the user set a musical score, which can be as well a random number, and choose a gesture family to interact with the algorithm.

Thanks to its properties the Wekinator is also suitable for human computer interactions in real-time video applications, interactive games or any other play-along system.

Algorithm. The user controller dataset may consist in FFT bin magnitudes or 2D/3D axis positions, that are provided by a joystick, a body tracked with Kinect, a hand tracked by a Leap Motion Controller and so on. The algorithm consists in several real time stages as specifing the input controllers, setting a specific learning method, creating or using an existent training dataset and running the trained to model to perform.

The input dataset and the output response are performed within OSC [34]. The input data is chosen among the built-in feature extractors or one can supply his own specific feature extractors. The former consists in time, spectral, edge detection or color-tracking features, implemented in Processing [35] or ChucK [36]. For training the model, the system has implemented several algorithms with Weka, such as neural networks, decision trees, nearest neighbor algorithm, AdaBoost and support vector machines. Weka is a very well known open-source library written in Java, which provides different classifiers and regressors. Due to this library, the Wekinator is able to furnish as output either a probability distribution, as a likelihood, either a specific class label, as a maximum likelihood. All the communication is achieved through OSC, therefore the output response of the trained model is used in any application that speaks through OSC.

The real time interactive algorithm of this system can be seen in Figure 9. One can set on-the-fly the machine learning methods with inputs, the specific features, choosing the type and the right parameters. In the same process the user builds the training dataset, the system trains the learner and finally the user can evaluate the results.

In the first stage, the user can setup a machine learning method and can also specify the features that will be used for learning. The type of the selected features influences the way of the learned mapper and the discretization over the system.

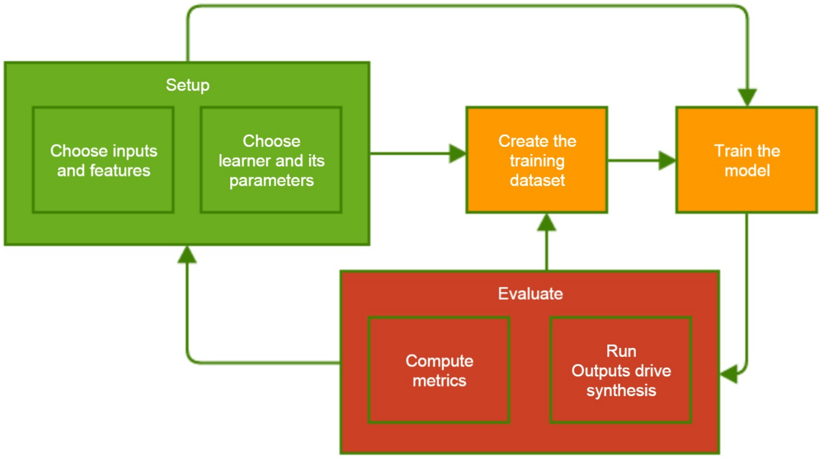

Setup
Choose inputs and features
Choose learner and its parameters
Create the training dataset
Train the model
Evaluate
Compute metrics
Run Outputs drive synthesis

Fig. 9. The interactive algorithm of the Wekinator

The training step provides the possibility of choosing the class labels or the regression function. In the learning stage, the Wekinator trains the model with the specific features, extracted in real-time. For example, one can set "1" as the desired output value, while someone raise up several times the left hand, tracked by Kinect. The training process is taking several seconds, and if it is not happening this way, the user can halt the process and tune the model to be faster, by readapting the parameters.

The testing phase consists in a real-time feature extraction, to be fed to the trained model in order to return the specific output values. For example, if one raises up the left hand, the output of the system will be "1" and this value may be used as a shutter for a thirdy part system.

In the evaluation step, if the results are not satisfying, the user is able to increase the size of the training dataset in order to reinforce positive behaviors. Then a new retraining process is needed, to remodel the system.

5.4 Application with Wekinator

This system is suitable to our case because it is user orientated, therefore the user can train the model in real time with the specific features, and also because it has a very fast pace of learning and retuning on the order of seconds. Since Robigame is a video game played by patients in real-time, one can very fastly tune the algorithm and start the training for the patient, setting two types of short games, one very easy, which should lead to the lacking of the patient interest and one very hard, which should lead to the frustration of the patient.

The learning is done in a subjective context, since each patient has different ways and different levels of intensity to express their frustration or their motivation. The explicit goal is to create a controllable and suitable map for each user.

The best way to describe the task is by using a regression model, which maps values between 0 and 1. In the training phase, one should start with a value of 0, corresponding to the beginning of the training game and ending with a value of 1, corresponding to the last part of the training game and fitting the apex of the expression.

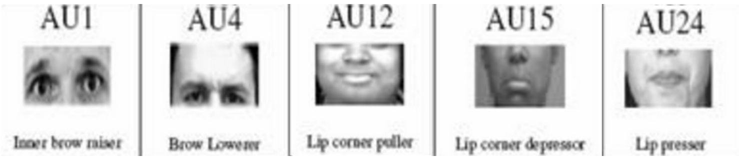


Fig. 10. Examples of the required action units extracted from Cohn-Kanade+ database [31]

In order to train the model, the following feature descriptors are used: AU1, AU4, AU20, AU15, AU12. Each of them are described below and can be seen in Figure 10. These algorithm try to track a combination of different emotions as confusion, frustration, anger, agitation, insecurity and sadness, which sum up a negative level in the patient and trigger the decreasing of the game level ([37] and [38]):

- confusion and frustration are described by a high value of AU1 (inner brow raiser)
- anger, agitation and insecurity are described by a high value of AU4 (brow lowerer)
- sadness is described by the following combination
  - low value of AU20 (lip stretcher), high value of AU15 (lip corner depressor) and low value of AU12 (lip corner puller)
  - low value of AU20 (lip stretcher), high value of AU15 (lip corner depressor) and normal value of AU12 (lip corner puller)

6 Setup ans Results

In this section we will describe the final setups used in this project and we will give some qualitative results. For lack of time during the project, the previous part about wekinator has not been integrated in the final process but it is interesting to analyze the results without this deeper face analyze. When the patient comes into the field of view of the KinectV2, seated in front of the reaplan, his skeleton is tracked, the head orientation is estimated and the face is analyzed for smiling detection. The Kinect is placed above the screen according result obtained after comparison of the two kinect position for the kinect (figure 11).

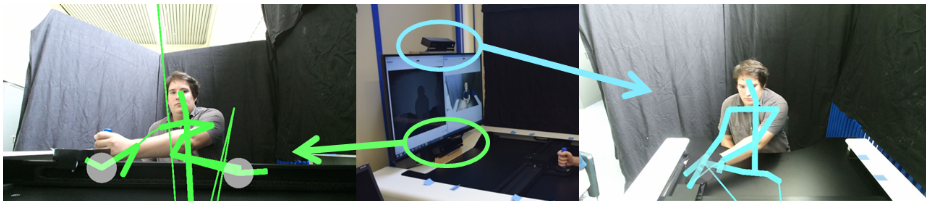


Fig. 11. Comparison of the view from the kinect below (green) and above the screen (light blue).

above the screen is the only position to track face correctly and also to track upper limbs. This position is also the one who gives quicker body detection.

When the user body is tracked, a computer is used to perform the kinect process and determines what the user is watching with an accuracy of a few centimeters: Main screen, effector distal plane or elsewhere. These information are completed by the smiling detection and are sent to the game (Figures 7). The screen displays the game and the user moves the distal effector to control the game. The difficulty of the game is also displayed on the screen and it changes with the user happiness. The final setup is given on Figure 12.

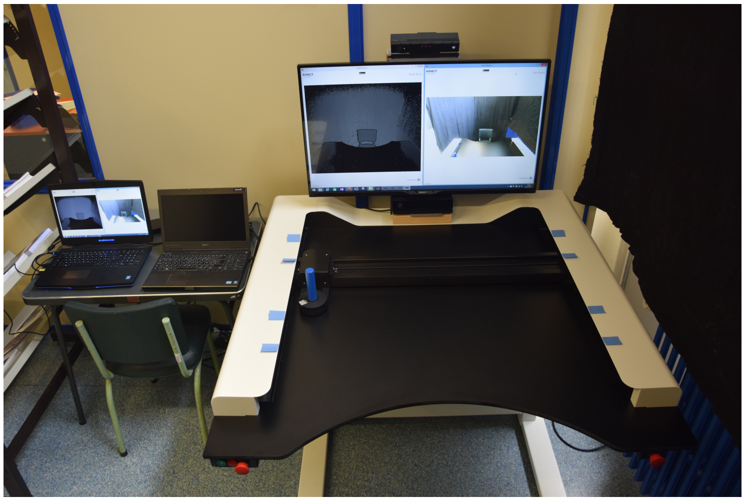


Fig. 12. Final setup with computers and REAPlan system .

The results obtained with healthy users are fully satisfactory. Users enjoyed the different games and have not got the feeling the use medical equipment. In addition, users say that the expressive facial analysis creates a new dimension in video games by changing the difficulty of the game. This first version of the robigame setup is functional in real time and is ready to be tested with patients in a therapeutic setting.

7 Conclusion and future work

The objective of the main project is to develop an intelligent serious game for rehabilitation of the upper limbs for stroke patients using an interactive rehabilitation robot which screenplay adapts to the patient's functional abilities, and which assistance evolves according to patients motivational, motor and cognitive performances. ROBiGAME will be developed on the REAPlan robot providing a distal effector which can mobilize the patient upper limbs in a horizontal plane but could also be transferred to other robots or even simpler rehabilitation setups. The most important benefit of this work is the integration between Kinect for Windows v2 and REAPlan robot. The application extracts facial features of the user from the data provided by the Kinect. Two games are developed for this purpose and motivation data, computed using the kinect, are used to increase or decrease game difficulty in real-time. In perspective of this work, we need to link face analysis with wekinator to main system. Then we must do quantitative analyses of arm movements. In addition, the analysis of movements must be validated with patients in a therapeutic setting for monitoring patients with real motor difficulties. Finally, a third game is in development and it would be interesting to finish it for testing.

Acknowledgment

This work is supported by the ROBiGAME Project funded by the Walloon Region from Belgium through the WBHealth Program. This work is also partially funded by the Sectoral Operational Programme Human Resources Development 2007-2013 of the Ministry of European Funds through the Financial Agreement POSDRU/159/1.5/S/132395.

The authors would like to thank our partners of the Robigame project, especially Martin Vanderwegen and Adrien Denis for the handling of the Reaplan.

The authors would like to thank the local organizers of eNTEFACE'15 for their kindness, their local assistance regarding logistics, their nice schedule of social events.The authors would also thanks to Radhwan and Ambroise for their sympathy and for sharing good and bad moments with us during the workshop.

References

1. M. Gilliaux, T. Lejeune, C. Detrembleur, J. Sapin, B. Dehez, and G. Stoquart, "A robotic device as a sensitive quantitative tool to assess upper limb impairments in stroke patients: a preliminary prospective cohort study," *Journal of rehabilitation medicine*, vol. 44, no. 3, pp. 210–217, 2012.
2. S. C. Cramer and R. J. Nudo, *Brain repair after stroke.* Cambridge University Press, 2010.

3. P. Langhorne, F. Coupar, and A. Pollock, "Motor recovery after stroke: a systematic review," *The Lancet Neurology*, vol. 8, no. 8, pp. 741–754, 2009.
4. A. C. Lo, P. D. Guarino, L. G. Richards, J. K. Haselkorn, G. F. Wittenberg, D. G. Federman, R. J. Ringer, T. H. Wagner, H. I. Krebs, B. T. Volpe *et al.*, "Robot-assisted therapy for long-term upper-limb impairment after stroke," *New England Journal of Medicine*, vol. 362, no. 19, pp. 1772–1783, 2010.
5. N. Norouzi-Gheidari, P. S. Archambault, and J. Fung, "Effects of robot-assisted therapy on stroke rehabilitation in upper limbs: systematic review and meta-analysis of the literature," *J Rehabil Res Dev*, vol. 49, no. 4, pp. 479–496, 2012.
6. G. Saposnik, R. Teasell, M. Mamdani, J. Hall, W. McIlroy, D. Cheung, K. E. Thorpe, L. G. Cohen, M. Bayley *et al.*, "Effectiveness of virtual reality using wii gaming technology in stroke rehabilitation a pilot randomized clinical trial and proof of principle," *Stroke*, vol. 41, no. 7, pp. 1477–1484, 2010.
7. P. Langhorne, R. Wagenaar, and C. Partridge, "Physiotherapy after stroke: more is better?" *Physiotherapy Research International*, vol. 1, no. 2, pp. 75–88, 1996.
8. E. Taub, N. Miller, T. Novack, E. Cook 3rd, W. Fleming, C. Nepomuceno, J. Connell, and J. Crago, "Technique to improve chronic motor deficit after stroke." *Archives of physical medicine and rehabilitation*, vol. 74, no. 4, pp. 347–354, 1993.
9. J. D. Schaechter, "Motor rehabilitation and brain plasticity after hemiparetic stroke," *Progress in neurobiology*, vol. 73, no. 1, pp. 61–72, 2004.
10. A. Pascual-Leone, A. Amedi, F. Fregni, and L. B. Merabet, "The plastic human brain cortex," *Annu. Rev. Neurosci.*, vol. 28, pp. 377–401, 2005.
11. P. Rego, P. M. Moreira, and L. P. Reis, "Serious games for rehabilitation: A survey and a classification towards a taxonomy," in *Information Systems and Technologies (CISTI), 2010 5th Iberian Conference on.* IEEE, 2010, pp. 1–6.
12. GENIOUS Interactive. (2012) Voracy fish: New multiplayer serious game for physical rehabilitation. [Online]. Available: http://www.voracy.com/
13. H. Krebs, N. Hogan, B. Volpe, M. Aisen, L. Edelstein, and C. Diels, "Overview of clinical trials with mit-manus: a robot-aided neuro-rehabilitation facility." *Technology and health care: official journal of the European Society for Engineering and Medicine*, vol. 7, no. 6, pp. 419–423, 1998.
14. F. Rocca, M. Mancas, and B. Gosselin, "Head pose estimation by perspective-n-point solution based on 2d markerless face tracking," in *Intelligent Technologies for Interactive Entertainment.* Springer, 2014, pp. 67–76.
15. P. Ekman and W. Friesen, *Facial Action Coding System: A technique for the measurement of facial movement.* Palo Alto, U.S.: CA: Consulting Psychologists Press, 1978.
16. J. W. Burke, M. McNeill, D. K. Charles, P. J. Morrow, J. H. Crosbie, and S. M. McDonough, "Optimising engagement for stroke rehabilitation using serious games," *The Visual Computer*, vol. 25, no. 12, pp. 1085–1099, 2009.
17. M. da Silva Cameirão, S. Bermúdez i Badia, E. Duarte, and P. F. Verschure, "Virtual reality based rehabilitation speeds up functional recovery of the upper extremities after stroke: a randomized controlled pilot study in the acute phase of stroke using the rehabilitation gaming system," *Restorative neurology and neuroscience*, vol. 29, no. 5, p. 287, 2011.
18. L. J. Buxbaum, A. M. Dawson, and D. Linsley, "Reliability and validity of the virtual reality lateralized attention test in assessing hemispatial neglect in right-hemisphere stroke." *Neuropsychology*, vol. 26, no. 4, p. 430, 2012.
19. Y. M. Kim, M. H. Chun, G. J. Yun, Y. J. Song, and H. E. Young, "The effect of virtual reality training on unilateral spatial neglect in stroke patients," *Annals of rehabilitation medicine*, vol. 35, no. 3, pp. 309–315, 2011.

20. N. Katz, H. Ring, Y. Naveh, R. Kizony, U. Feintuch, and P. Weiss, “Interactive virtual environment training for safe street crossing of right hemisphere stroke patients with unilateral spatial neglect,” *Disability and rehabilitation*, vol. 27, no. 20, pp. 1235–1244, 2005.
21. R. L. Myers and T. A. Bierig, “Virtual reality and left hemineglect: A technology for assessment and therapy,” *CyberPsychology & Behavior*, vol. 3, no. 3, pp. 465–468, 2000.
22. E. Klinger, A. Kadri, E. Sorita, J.-L. Le Guiet, P. Coignard, P. Fuchs, L. Leroy, N. Du Lac, F. Servant, and P.-A. Joseph, “Agathe: A tool for personalized rehabilitation of cognitive functions based on simulated activities of daily living,” *IRBM*, vol. 34, no. 2, pp. 113–118, 2013.
23. Hasomed. Rehacom : Computerassisted cognitive rehabilitation. [Online]. Available: https://www.hasomed.de/
24. Fishing Cactus. (2010) R.o.g.e.r a kinect prototype for cognitive observation. [Online]. Available: http://seriousgaming.fishingcactus.com/en/Projets/Roger/roger_ulb_hemi_negligence_kinect_medical.html
25. “Monodevelop,” 2015. [Online]. Available: http://MonoDevelop.com/
26. “Osc opensoundcontrol,” 2015. [Online]. Available: http://opensoundcontrol.org
27. “Face tracking. microsoft developper network.” [Online]. Available: http://msdn.microsoft.com/en-us/library/jj130970.aspx
28. P. Ekman, W. Friesen, and J. Hager, *Facial Action Coding System (FACS) Manual*. U.S.: Research Nexus division of Network Information Research Corporation, 2002.
29. L. Lo Presti and M. La Cascia, “Using hankel matrices for dynamics-based facial emotion recognition and pain detection,” June 2015.
30. J. Cohn and F. D. la Torre, *Automated Face Analysis for Affective Computing*. Oxford University Press, 2002.
31. P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, “The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression,” in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*. IEEE, 2010, pp. 94–101.
32. R. Fiebrink and P. R. Cook, “The wekinator: a system for real-time, interactive machine learning in music,” in *Proceedings of The Eleventh International Society for Music Information Retrieval Conference (ISMIR 2010). Utrecht*, 2010.
33. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The weka data mining software: an update,” *ACM SIGKDD explorations newsletter*, vol. 11, no. 1, pp. 10–18, 2009.
34. M. Wright and A. Freed, “Open sound control: A new protocol for communicating with sound synthesizers,” in *ICMC 97*, 1997.
35. C. Reas and B. Fry, “Processing: a learning environment for creating interactive web graphics,” in *ACM SIGGRAPH 2003 Web Graphics*. ACM, 2003, pp. 1–1.
36. G. Wang, P. R. Cook *et al.*, “Chuck: A concurrent, on-the-fly audio programming language,” in *Proceedings of International Computer Music Conference*, 2003, pp. 219–226.
37. J. F. Grafsgaard, J. B. Wiggins, A. K. Vail, K. E. Boyer, E. N. Wiebe, and J. C. Lester, “The additive value of multimodal features for predicting engagement, frustration, and learning during tutoring,” in *Proceedings of the 16th International Conference on Multimodal Interaction*. ACM, 2014, pp. 42–49.
38. J. Grafsgaard, J. B. Wiggins, K. E. Boyer, E. N. Wiebe, and J. Lester, “Automatically recognizing facial expression: Predicting engagement and frustration,” in *Educational Data Mining 2013*, 2013.