Human Interaction and Teleoperation

PRACTICUM:
Kinect

Ferran Larriba Gil
Felip Martí Carrillo
Edgar Riba Pi

January 2013

Table of Contents

1 Abstract....1

2 Introduction....2

3 Design....3

- 3.1 Hardware....3
- 3.2 Software....3
  - 3.2.1 Moving the slider....4
  - 3.2.2 Click....5
- 3.3 Driver requirements....5
  - 3.3.1 Requirements....5
  - 3.3.2 How to install....6
  - 3.3.3 How to execute....6

4 Tests....6

- 4.1 Test 1....6
- 4.2 Test 2....7
  - 4.2.1 Game 1: Click and wipe out the ice....8
  - 4.2.2 Game 2: Push the water....8
  - 4.2.3 Game 3: Wipe out the ice....9
  - 4.2.4 Game 4: Click the fish....9
  - 4.2.5 Game 5: Click the blue circle....10

5 Problems....10

6 Conclusions....11

7 References....12

8 Annex....13

Table of Figures

Fig 1: Kinect sensor....3
Fig 2: Proposed program....3
Fig 3: No hand detected....4
Fig 4: Hand detected....4
Fig 5: Closed hand - Click....5
Fig 6: Test 1 screenshot-1....7
Fig 7: Test 1 screenshot-2....7
Fig 8: Test 1 screenshot-3....7
Fig 9: Test 1 screenshot-4....7
Fig 10: Game 1: Wipe out the ice....8
Fig 11: Game 2: Push the water....8
Fig 12: Game 2: Push the water....8
Fig 13: Game 3: Wipe out the ice....9
Fig 14: Game 4: Click the fish....9
Fig 15: Game 5: Click the blue circle....10
Fig 16: Detection of forearm....10
Fig 17: Centroid displaced....10

1 Abstract

In this project, we propose a complementary system for some rehabilitation tasks that permits a human teleoperate a computer with hand movements using a Kinect sensor. The system consists in a Kinect (sensor to capture image and depth measurements), a laptop with GNU/Linux operating system, OpenCV libraries for the image processing and hand detection. And finally, a set of tests to evaluate the utility of the system.

2 Introduction

In the last years the physiotherapy community are focused in study how to combine all their knowledge with the new technologies to perform some rehabilitation tasks. In the rehabilitation context the virtual games can offer to the therapists and the patients a whole of possibilities and benefits complemented with the traditional therapies. In order to design the system it is very important to know in which kind of disease we are focused.

The system that we propose moving the mouse pointer of a computer with some virtual games could be useful to facilitate the exercise and rehabilitation of patients with ***Parkinson's disease*** **(PD)**. ***PD*** also known as ***idiopathic or primary parkinsonism*** is a degenerative disorder of the central nervous system. The motor symptoms of Parkinson's disease result from the death of dopamine-generating cells in the substantia nigra, a region of the midbrain; the cause of this cell death is unknown. Early in the course of the disease, the most obvious symptoms are movement-related; these include shaking, rigidity, slowness of movement and difficulty with walking and gait.

With the proposed system are required the patient/player movements and in some cases we could also require this movements with a payload. As is said, this kind of games in patients with PD could offer to the therapists an additional information to evaluate the patients state. This kind of games are very entertained and force the patient to focus all his efforts in the game instead of his motor deficits making the rehabilitation exercise more pleasant and giving him more adherence in the rehabilitation.

3 Design

3.1 Hardware

Inside the sensor case, a Kinect for sensor contains:

- An RGB camera that stores three channel data in a 1280x960 resolution. This makes capturing a color image possible.

- An infrared (IR) emitter and an IR depth sensor. The emitter emits infrared light beams and the depth sensor reads the IR beams reflected back to the sensor. The reflected beams are converted into depth information measuring the distance between an object and the sensor. This makes capturing a depth image possible.

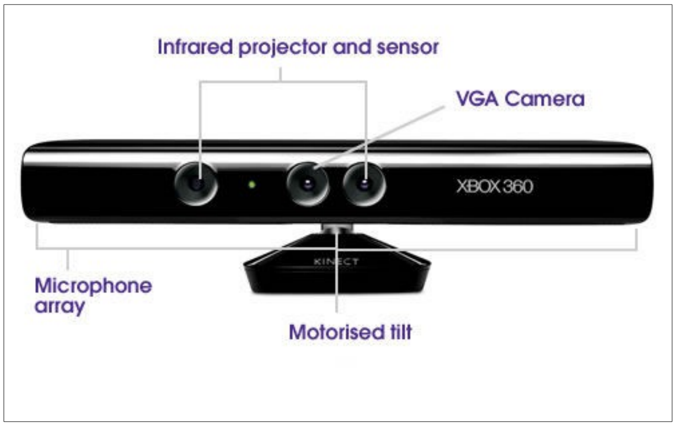

Fig 1: Kinect sensor

- A multi-array microphone, which contains four microphones for capturing sound. Because there are four microphones, it is possible to record audio as well as find the location of the sound source and the direction of the audio wave.

- A 3-axis accelerometer configured for a 2G range, where G is the acceleration due to gravity. It is possible to use the accelerometer to determine the current orientation of the Kinect.

3.2 Software

Figure 2 shows a user working with the proposed program. The yellow figure is the body of the user. When the hand of the user comes closer to the Kinect beyond some threshold, the program automatically detects it (green figure).

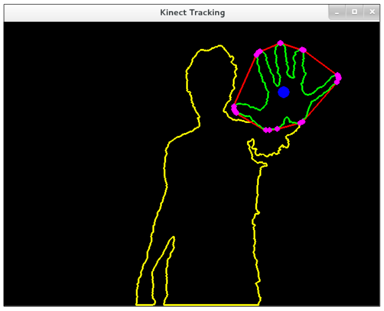

Fig 2: Proposed program

The object *'Hand'* is composed for the folowing elements:

1. **Hand perimeter:** The hand profile, corresponds to the green figure of the last picture.
2. **Convex Hull:** Convex hull of the hand, represented with red lines. It controls the clicking motion.
3. **Vertices:** Vertices of the convex hull, as pink dots in the picture.
4. **Centroide:** Centroide of the convex hull, represented as a blue dot. This is what really controls the mouse.

3.2.1 Moving the slider

Once the program detects that the hand of the user, the slider moves proportionally to the Centroide movements.

Sometimes, the user can not achieve the goal position of the slider and a “clutch function” is needed, resembling to lifting the mouse of your computer. This “Clutch function” is performed moving the hand away of the Kinect, until it pass the threshold. The link between your hand and the slider stops. Then the user can move its hand to a more comfortable position and move your hand forwards closer to the Kinect again, to continue the trajectory.

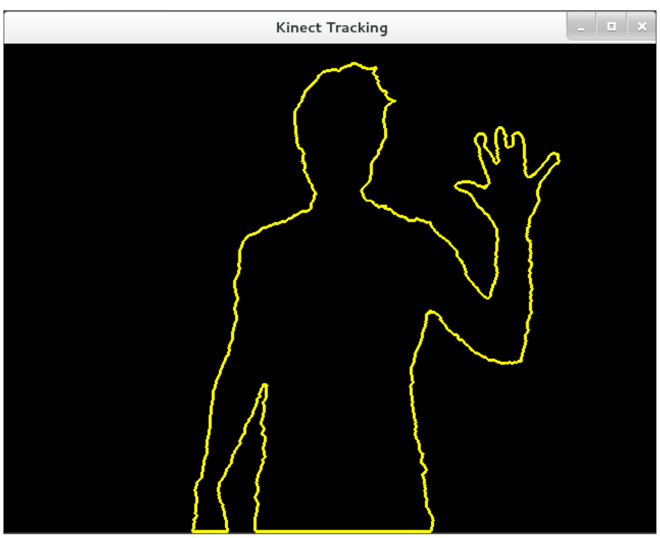

Fig 3: No hand detected

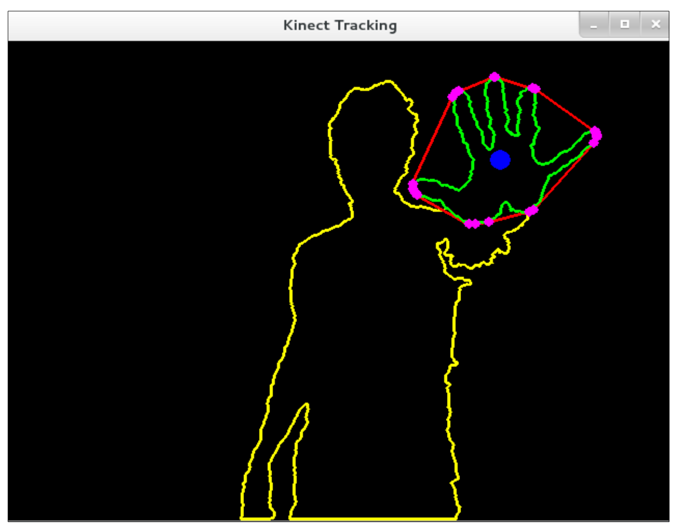

Fig 4: Hand detected

3.2.2 Click

To do a click the user must close his hand. The program detects that the hand is closed for two reasons:

1. **The Convex Hull area becomes smaller.**
2. **The ratio of wasted space increases until almost 100%.**

It is called "Wasted space" to the difference of the Convex Hull area and the Hand area.

$$Ratio\,of\;\;wasted\,space=\frac{Hand\;Area}{Convex\;Hull\;Area}$$

```
if Ratio of wasted space ~ 60%
  OpenHand = True
else if Ratio of wasted space ~ 90%
  CloseHand = True
```

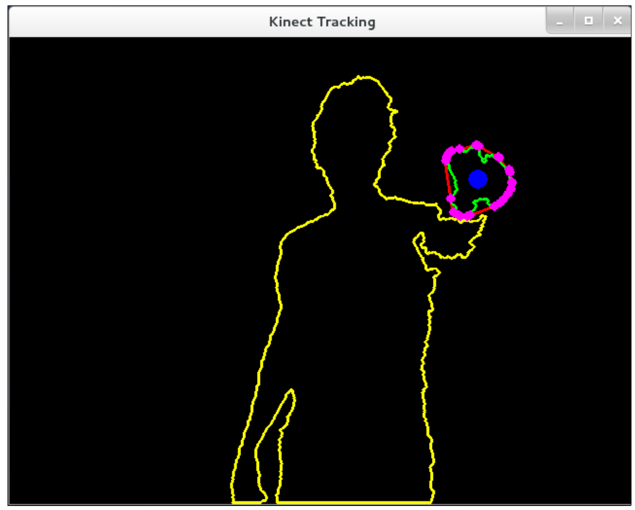

Fig 5: Closed hand - Click

3.3 Driver requirements

Once we have the proposal, we have found a code in python made by Alexander James Wallar that fulfills our requirements. This program runs on GNU/Linux, concertedly in Debian and Debian-like distributions (Ubuntu, Linux Mint, etc).

3.3.1 Requirements

In the list above are all the software and drivers requirements to use the program proposed:

- OpenKinect
- Python Wrapper for OpenKinect
- A Linux machine using Ubuntu
- OpenCV 2.1
- OpenCV 2.3
- Python 2.7.2
- A Microsoft Kinect
- A Microsoft Kinect USB Adapter
- PyGame
- Xlib for Python
- USB 2.0

3.3.2 *How to install*

Executing the following command, all the libraries and requirements will be installed in a GNU/Linux operating system:

```
$ sudo apt-get install freenect python-freenect python-opencv opencv-doc python-pygame python-xlib python-gpgme python-numpy -y
```

3.3.3 *How to execute*

1. Connect the kinect to a USB 2.0 port (Important, it doesn't work in a USB 3.0 port)
2. Execute the code with the following command:

   ```
   $ sudo python Hand_tracking.py
   ```

4 Tests

In order to test the effectiveness of the system, first the system as itself should be tested with some basic exercises to determine if it works or not; second the system should be tested with a little more complex exercises to determine if patients with PD could handle it.

4.1 *Test 1*

The first test is to determine if the system works properly to later achieve more complex exercises. It should be a very simple game where colored circles of different size will be appearing in the screen as the player are clicking it. When you click one circle it disappears and then another one appears in a different place and size on the screen. A clock and a counter of the clicked circles should be in the screen in order to extract some statistics.

In the figures above we can see screenshots of the appearance of the game:

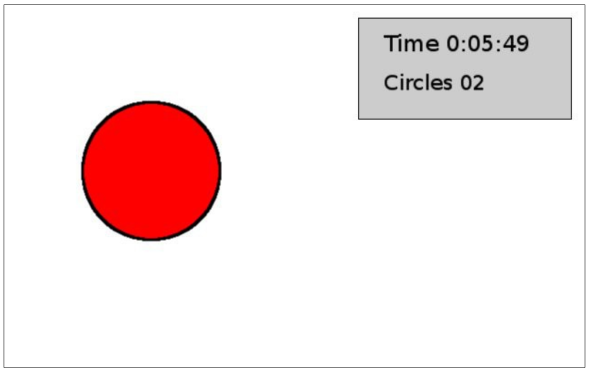

Fig 6: Test 1 screenshot-1

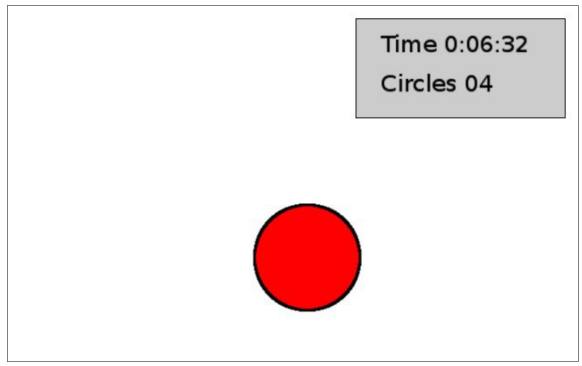

Fig 7: Test 1 screenshot-2

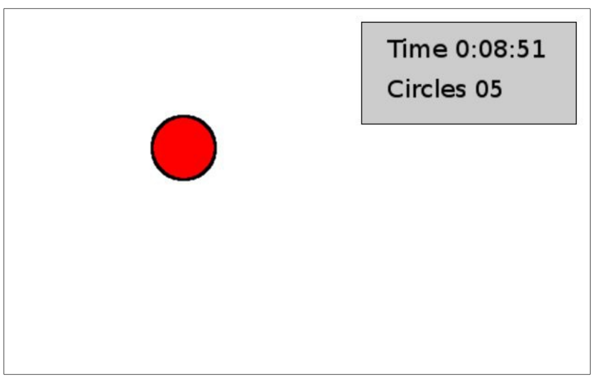

Fig 8: Test 1 screenshot-3

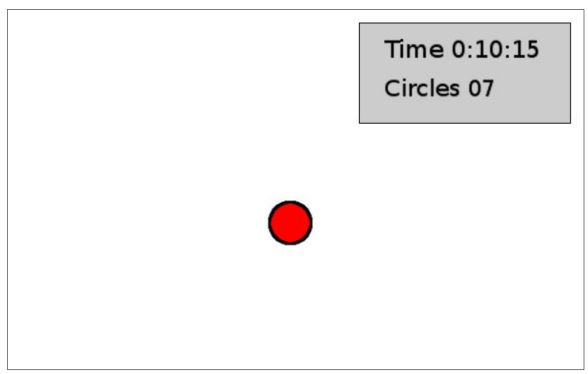

Fig 9: Test 1 screenshot-4

4.2 Test 2

The second test will be done with a more complex interactive games. In this case we propose to use GCompris which is a high quality educational open-source software suite comprised with numerous activities for children.

The reason why we choose this software is because it have many games oriented for children and those games could be easy to play for people without PD with a mouse or with the system proposed, but as we will see in the results obtained after some testing, it's not as easy as it seems because you have to put all of your attention to perform the exercise, and this is one of the main objectives in the rehabilitation for patients with PD.

One important point to remark is that as all in the medical fields, for each patient will be his therapist who will design the exercises routine. In this document we only propose some exercises to the therapists in order that they choose which ones will be suitable for the patient depending on the grade of his disease.

Some of the exercises comprised in Gcompris are the following:

4.2.1 Game 1: Click and wipe out the ice

The aim of this game is that the patient have move the cursor with the hand and click all the statics ice blocks to eliminate it. When you wipe out all the blocks you pass to the next level.

Fig 10: Game 1: Wipe out the ice

4.2.2 Game 2: Push the water

The aim of this game is that the patient have move the cursor left to right to push the water out of the pipe. If you are slow, the water comes back. When you push all the water out, you pass to the next level.

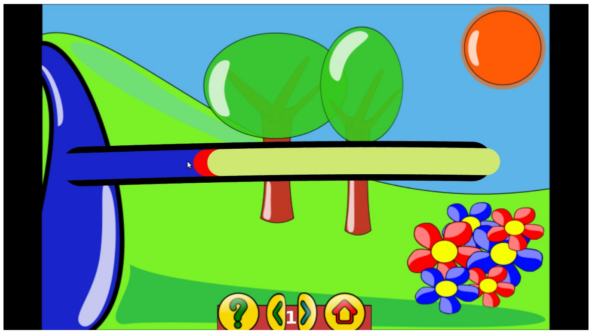

Fig 12: Game 2: Push the water

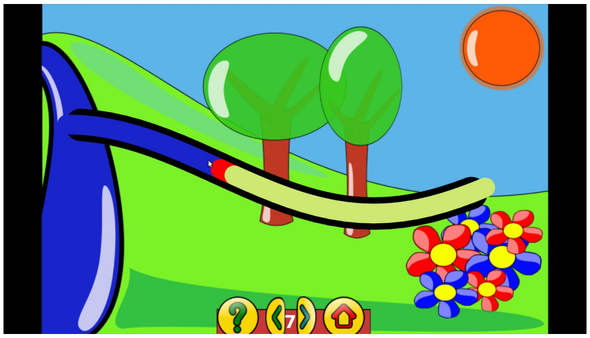

Fig 11: Game 2: Push the water

4.2.3 Game 3: Wipe out the ice

The aim of this game is that the patient have move the cursor with the hand over all the statics ice blocks to eliminate it. When you wipe out all the blocks you pass to the next level.

Fig 13: Game 3: Wipe out the ice

4.2.4 Game 4: Click the fish

The aim of this game is that the patient have to click on the fishes that are swimming in the sea. Now the target is not static, is moving. When you click all the fishes in the sea you pass to the next level.

Fig 14: Game 4: Click the fish

4.2.5 Game 5: Click the blue circle

The aim of this game is that the patient have to click on the blue circles. When a blue circle is clicked it changes to green and a shape is drawn. In this case the target is static. When you click all blue circles the you pass to the next level.

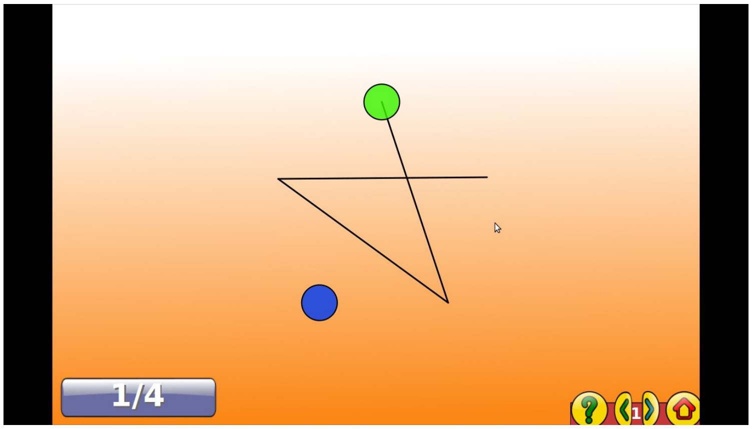

Fig 15: Game 5: Click the blue circle

5 Problems

5.1 Ergonomics

Using the kinect to control the mouse of the computer is not ergonomic at all. One of the problems that we have found is when we use this program in a more relaxed and ergonomic way, the forearm is also detected.

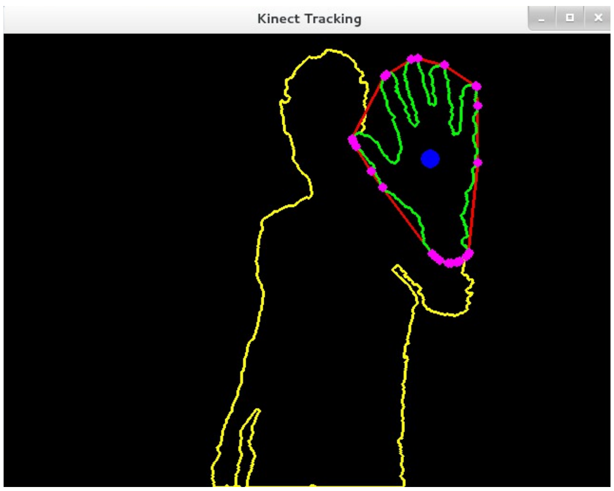

Fig 16: Detection of forearm

Fig 17: Centroid displaced

This is a problem because (as is shown in the above figures) when we want to close the hand to make click the centroid is displaced. So, you will never click in the desired position if you use this system in a relaxed or more ergonomic way.

5.2 Double click

Other of the problems that we have found is that to make the double click action is very difficult. We are not used to do this kind of movement with our hand. Therefore, it is important to find another kind of movement to make this action, or to increase the time delay between clicks. There are some videos in Annex that shown this problem.

5.3 Precision

For sure this system is not suited for precision tasks. It would be an improvement of the code to implement an option for precision, for example, when the other hand also appears in the screen; or using the profundity, more closer to the kinect more precision. This precision option should include a filter and a referencing system for the position control.

6 Conclusions

The objective of controlling the slider of the computer using a Kinect sensor has been accomplished, but this method has a couple of troubles to take in account:

- It is not as ergonomic as the use of a mouse: it needs more effort of the user and also a time of training for the user is necessary to control the slider.
- Hardness of clicking: when the user closes the hand for do the click, the centroid position usually does not remain still, but moves a little bit. Then the click is done outside where the user goal.

For that reasons, is unthinkable to work with this system for a regular user to replace the use of the mouse. Anyway, this system can be used for the purposed context of the rehabilitation. Other uses of this system would be for example:

- Virtual clothing testers.

- Information points in public places, i.e. in airports, universities…
- TV teleoperation. - etc.

Future studies could make a clustering of different patterns to control different actions beyond of only “click” control, like zoom of the image or close the current window, in an easy and intuitively way for the user.

From the point of view of the rehabilitation this kind of systems are not directly a solution, but as complement for the therapy may help the patient and the therapist. Other future study could be to implement a module that send some statistics to the therapist in order to follow the evolution of the patient without having to go to therapist office.

7 References

[1] Michelle J Johnson. *Recent trends in robot-assisted therapy environments to improve real-life functional performance after stroke*. Journal of neuroengineering and rehabilitation. USA, 2006 3:29.

[2] Judith e. Deutsch. *Virtual Reality and Gaming Systems to Improve Walking and Mobility for People with Musculoskeletal and Neuromuscular Conditions*. Doctoral programs in Physical Therapy. USA, 2011.

[3] Alexander James Wallar, *OpenKinect Mouse Control Using Python* (Python recipe) [On line] (12/01/2014) http://code.activestate.com/recipes/578104-openkinect-mouse-control-using-python/?in=user-4179768

[4] Bruno Coudoin, *Gcompris* [On line] (12/01/2014) http://gcompris.net/index-en.html

8 Annex

The code of the kinect program can be download in the following link:

https://www.dropbox.com/s/smd8qj65tu37v4d/Hand_tracking.py

Some videos of the results can be found in the following link:
https://www.dropbox.com/sh/xgq6lsm1iq2bai2/oI8N2BvPHt

- Using the kinect in the operating system (Debian)
- Using the kinect to play the click the fish game
- Using the kinect to play more games
- Using the kinect without ability in the operating system