Cervical amplitude recording with virtual reality device

Florian Garibal

Computer sciences and Applied Mathematics, E.N.S.E.E.I.H.T, Toulouse, France

florian.garibal@etu.enseeiht.fr

Abstract

This two months project was requested by the Toulouse Institute of Osteopathy, represented by Denis Ducommun, in collaboration with two ENSEEIHT teachers, Sandrine Mouysset and Jerome Ermont. The main goal was to create a software that allows osteopath to record cervical amplitude movements thanks to a virtual reality device put on patient head and then make them follow a target visually and thanks to curves obtained distinguish healthy and unhealthy patient by comparing them with a mathematical model. We worked separately on the mathematical and target/GUI part to improve our efficiency. On the GUI part we faced some troubles about improving last year project and thereby we decided to restart everything from the beginning because it should be quicker. The second part was to integrate both headset application and mathematic modeling into the GUI code. This integration was a bit difficult especially the socket communication with the Oculus headset. But we got it working and we achieve almost every objective of the project. Indeed, we faced some troubles about mathematic modeling because of the lack of unhealthy patient data so we created and displayed three models in order to help the osteopath to find the best one in the future.

Categories and subject descriptors

[SOFTWARE ENGINEERING]: GUI design, Oculus Rift, PyQt, Python

Keywords

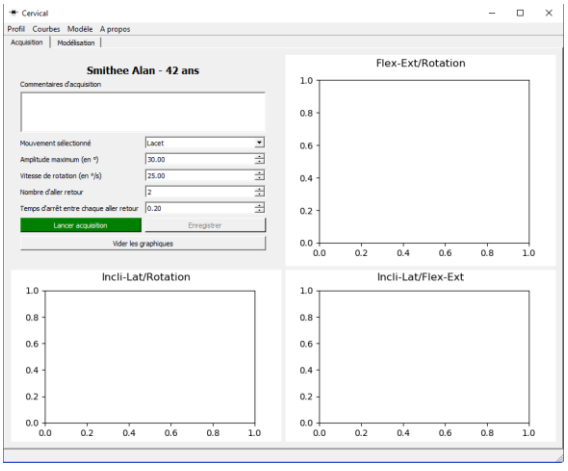
Virtual reality, Cervical movements recording, GUI design

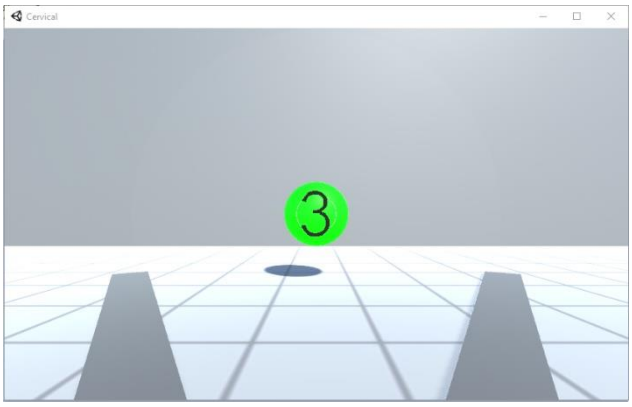
1. Introduction

This two months project was requested by the Toulouse Institute of Osteopathy, represented by Denis Ducommun, in collaboration with two ENSEEIHT teachers, Sandrine Mouysset and Jerome Ermont. The main goal was to create a software that allows osteopath to record cervical amplitude movements thanks to a virtual reality device and then distinguish healthy and unhealthy patient by comparing results with a mathematical model. Our part of the project was to improve the application that was developed last year, in the same context, by normalizing head movements with a target displayed inside the headset and also create the mathematical model that allow to distinguish patient. This target should take parameters from a GUI (Graphical User Interface) to change its speed, its maximum angle, etc.

2. Application overview

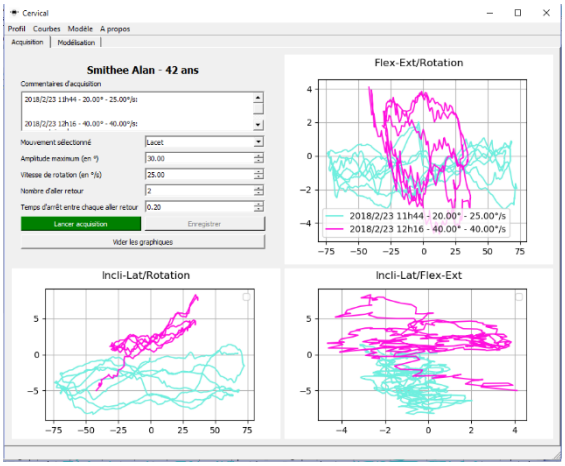
At first, we decided to design the overall architecture to avoid confusions in the future. To record cervical movements the workflow will be the following one: the patient put the headset on his head, the osteopath launches our application. Then the patient sees a target inside the headset, meanwhile the osteopath creates (or load) patient profiles and set up the acquisition parameters and launch it. Then the GUI send parameters to the headset and the target movements starts. At the end of the acquisition, the osteopath sees the results on the GUI and then compares it with mathematical models to know whether the patient is healthy or not.





3. Interface design

The main part we had to do during this project was to adapt the graphical user interface in order to allow the osteopath to specify the parameters he wanted for the movements acquisition. Thereby we had to read and understand last year project to know where we should plug in the code we needed to add some inputs. Unfortunately, after a week of trying, we could not achieve this objective because of a lack of documentation. Moreover, the application architecture was not designed to receive new code easily. Thereby, we had to make a choice if whether we tried one more week to understand this project and add the code we needed or if we restarted everything from the very beginning. After reading some information on library we should use and assess the time it takes me to restart everything, we arranged a meeting with the team in order to present them my point. Everybody agreed on the fact that we should restart everything and we should use Python as main language for three reasons: it is a very good and easy to learn language, it should be easy to use socket with it and mathematical modeling already use this language so it will be easy to merge our code in one application.



As expected, restarted everything was way quicker than adding our code and we achieved our main goal easily. The next part was to integrate both the connection with the Oculus and the mathematic modeling into the graphical user interface.

4. Integration

After developing each application and functionality separately we needed to connect them. We decided one the one hand to connect the Oculus headset and the GUI through sockets and on the other hand to merge the code of mathematic data analysis into the GUI code. To complete the first connection, we needed to design a sequence diagram to know which message will be sent, in which order and with which parameters. Indeed, on the one hand the GUI had to send the acquisition parameters through this socket to match the osteopath acquisition requirements and on the other hand the headset had to send back the result it got. Concerning the second part, we just moved the files into the main GUI project and we called data analysis function to create, load and display model and also to compare acquired data with loaded model.

5. Conclusion

To conclude, our project was almost 100% successful. Indeed, we had some troubles about creating efficient mathematical models because of a lack of unhealthy patient data which brings us to ask ourselves “What is a pathology in term of data?”. Thereby we created three models and we display them to the osteopath through the GUI to help him, in the future, in deciding which one is the best one. That aside, we successfully managed to get everything else working, from the moving target in the headset to the GUI that allows to specify parameters.

6. References

[1] https://github.com/guillaumehottin/CervicalKineRecorder

[2] https://github.com/guillaumehottin/CervicalKineRecorder/tree/master/Report/user_guide

[3] https://github.com/guillaumehottin/CervicalKineRecorder/tree/master/Report/technical_guide