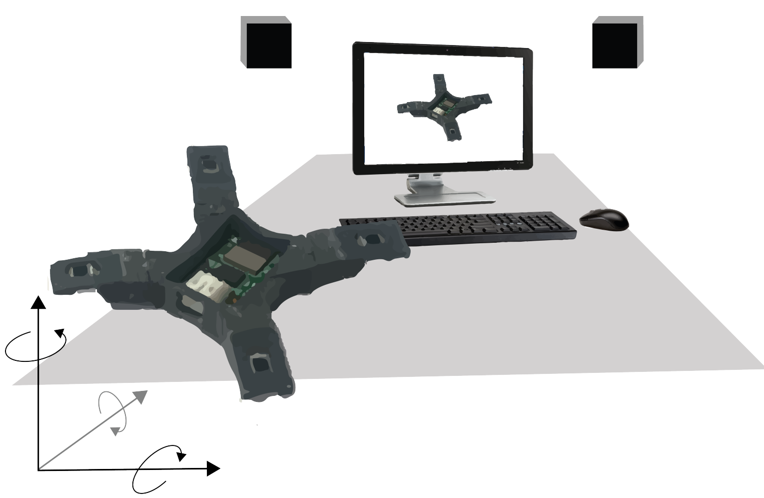
4TH YEAR INTERNSHIP

Sensor Fusion of IMU and 3d Positioning data for Virtual/Augmented/Mixed Reality



By **Julien Mellet**

Referring teacher:
Serge Dos Santos

Company tutor:
Sinan Haliyo

Academic year 2017/2018

Abstract

In an international work team, a sub-millimeter and cheap position tracker is in development. Electronics is functional since a proof of concept as been developed. The tracker embeds two types of sensor (optical measurement and IMU), the purpose of this stage is to make fusion of the two types of data to improve accuracy and stability of measurements.

The internship took place in the following way. A first study allowed to understand the project tools by making a simulation of the system. It follows a phase of implementation of these tools, then a phase of designing a rigid enclosure. Finally, a phase of characterization of various tools implemented was carried out.

After the internship, the tracker has theoretical tools set up and functional. However, the expected accuracy is not here. The noise of the measurements is greater than expected. The implementation of other tools will then be necessary.

Key words : accelerometer, augmented reality, Blender, HIVE Tracker, HTC Vive, Kalman, IMU, Lighthouse, Madgwick, mixed reality, sensor fusion, tracker 3D, Unity, virtual reality

Summary

Introduction

The first animated films have largely helped the implementation of position trackers. Cleverly placed on the body, these markers can virtually reconstruct the movements of actors. The entry level of these devices cost tens of thousands of euros, so only the big productions can afford to invest in this equipment. But the arrival of virtual reality (VR) systems have drastically reduced costs. This was followed by an expansion of the range of localization technologies. The one we will study here comes from the HTC Vive. We will see later how it works and its advantages.

We will first present in this report the laboratory where the training course took place, then we will explain the principle of operation of the tracker. We will continue on the realization of a simulation which resumes the principle of functioning of the tracker. We will then see how to carry out a fusion with the various sensors that the system embarks. Finally we will study the HIVE Tracker which will allow us to discuss its performance and the limits of its use.

Nota: Only the general concepts and their explanations are presented here. The complete code and files are available on my GitHub repository at the following address https://github.com/JulienMellet/Kalman-Filter

Chapter 1

Presentation of the laboratory and the tracker

1.1 The ISIR laboratory

Founded in 2007, the Institute of Intelligent Systems and Robotics (ISIR) is a Joint Research Unit (UMR). It is common to Sorbonne University (formerly UPMC) and the National Center for Scientific Research (CNRS). Located in the heart of Paris on the historical campus of Jussieu, this university was the place of Pierre and Marie Curie's work on radioactivity. The main building of the robotics works, the pyramid, gathers researchers, teacher-researchers and engineers for a staff of 145 people. The laboratory is multidisciplinary and includes mechanics, automation, signal processing and computer science. The fields of application are extremely varied since they include surgical assistance, functional rehabilitation, micro/nano-manipulation, haptics, artificial intelligence...

Four teams compose the laboratory and have different fields of action: AGATHE, AMAC, INTERCATION and SYROCO. The INTERACTION team, which develops techniques for the interaction of physical and virtual worlds, is composed of 5 working groups. One of them, Human-Computer Interaction (HCI), seeks to improve human-machine interactions known with new technologies. This group welcomes Cédric Honnet, the engineer I worked with. He is one of the initiators of the HIVE Tracker.

1.2 The HIVE Tracker

The HIVE Tracker is a miniaturization of the HTC Vive Tracker.

Figure 1.1: Vive Tracker and HIVE Tracker

This tracker is developed in collaboration with 3 laboratories: UPV (Valencia), the Kampff Lab at UCL (London) and ISIR at the Sorbonne (Paris). It aims to offer a sub-millimeter 3D localization system at an affordable price.

Chapter 2

System simulation

2.1 The Lighthouses

Lighthouses are sold with the HTC Vive virtual reality (VR) hardware. They are small, fixed, stand-alone systems that allow you to locate VR accessories in space. They are positioned high in a room and must be visible to the controllers and the HTC Vive headset.

The principle of operation of the bases is as follows: a global emission (or flash) then a precise scan of the space.

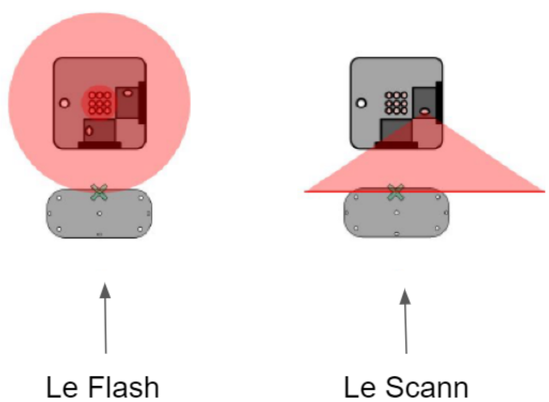


Figure 2.1: The flash and scan of a Lighthouse

Another Flash follows, then a scan perpendicular to the previous one. These steps are repeated in a very stable manner over time.

2.2 The principle of positioning

2.2.1 The theoretical principle

A photodiode, sensitive to the light of a Lighthouse can then measure the time between a flash and a scan. Two measurements, propositional to the angles of two scans, allow to obtain a vector pointing to the photodiode. With two lighthouses, two vectors can be obtained. Knowing the position and orientation of the lighthouses, the intersection of these two vectors gives an estimate of the position of our photodiode.

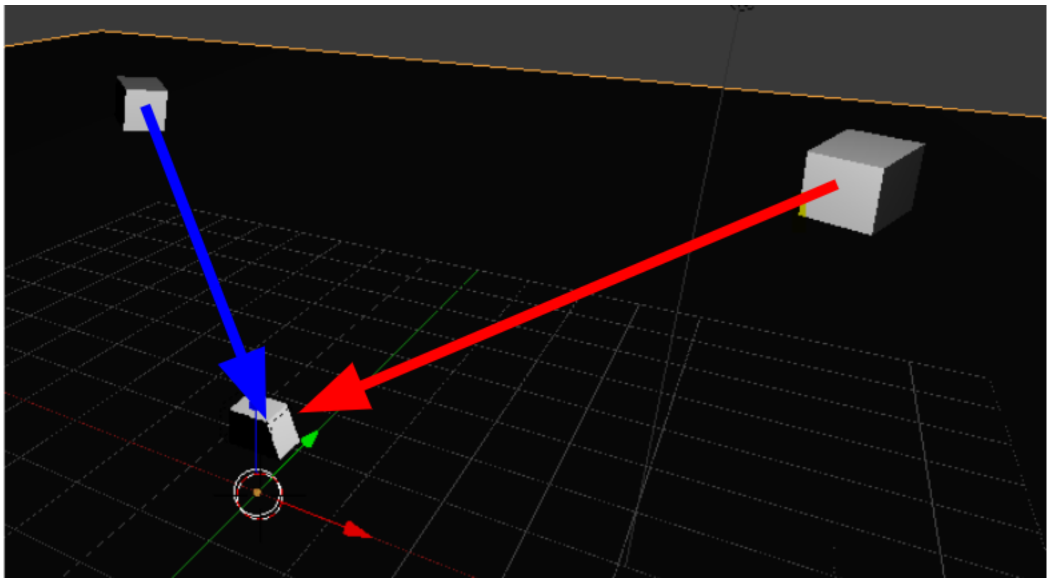


Figure 2.2: Representation of the intersection of the vectors coming from the Lighthouses and passing through a photodiode (extract of a first simulation test under Blender)

The imperfect intersection

Since it is very difficult for two non-colinear vectors to intersect in space, we have to define what is considered as their intersection. We have chosen here to use the middle of the segment perpendicular to the two vectors.

2.2.2 Other principle

Contrary to the principle stated above, it is possible to use only one Lightouse if we assume that the diodes are on a solid.

We know the distances between the photodiodes and a direction vector at the center of each diode from the Lighthouse. In addition, we seek the distances of each photodiode from the Lighthouse. The HIVE Tracker having 4 photodiodes, we can write 6 inter-distancial equations of the photodiodes, and have 4 unknowns.

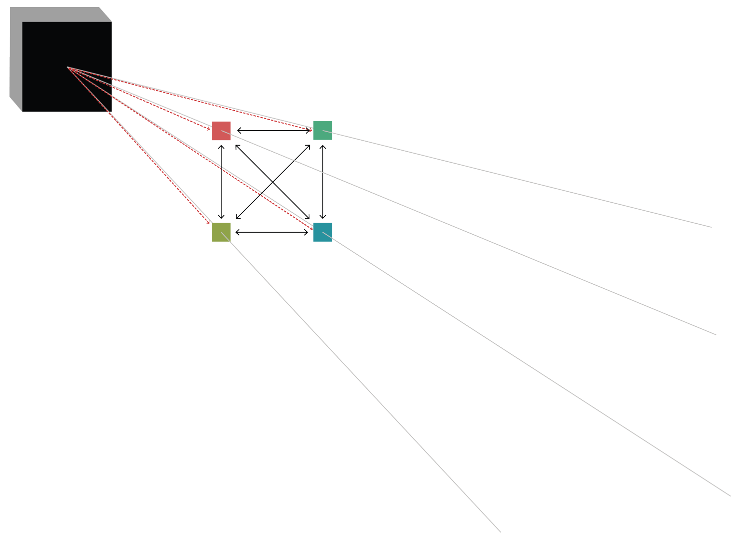


Figure 2.3: Single Lighthouse positioning

It is then necessary to solve a system of equations with 4 unknowns.

2.3 The simulation

Blender being an open source and multiplatform software, we have chosen to use it to realize a simulation using the principles previously stated. We will use its game engine, the Blender Game Engine (BGE) which is widely used to make physical simulations or visualizations.

Scanning in the simulation

The scan of the planes is done by imposing a rotation on a plane. When the signed distance separating the plane from the current object changes sign, we will consider that scan has reached the photodiode and keep this angle. With two angles and two lighthouses we have the operating principle of the real system.

How the simulation works

The final version of the simulation is made like a video game. A small blue vehicle can be moved with the numeric keypad in a living room. Its calculated position is symbolized by a green sphere. For the sake of understanding the working principle of the HTC Vive system, the rotation speed, which is actually 120Hz, is slowed down here. In addition, the scan planes are colored in translucent red.

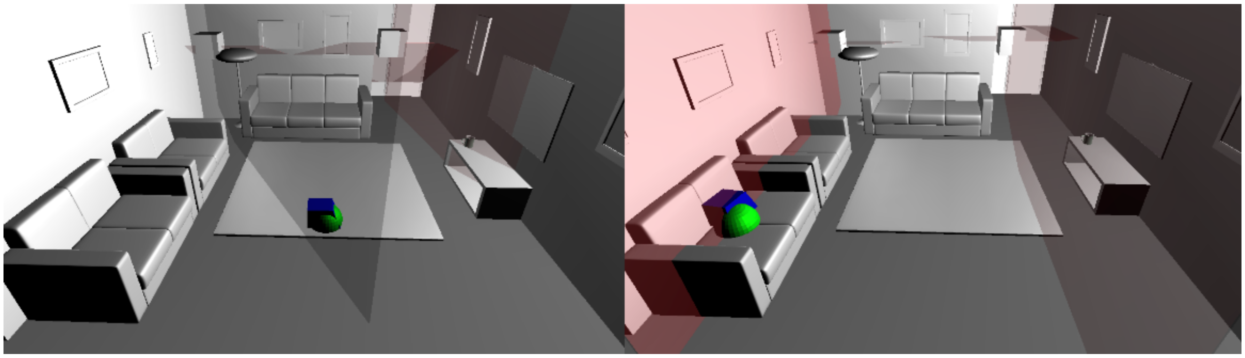


Figure 2.4: Start of the simulation on the left ; And after a move to the right

To better understand the simulation, the $LH\_Simu.blend$ file is available in the "Dev" branch of the sensor fusion repository: https://github.com/JulienMellet/Kalman-Filter/tree/Dev/Simulations

Simulation discussion

We notice that the position of the current object does not correspond exactly to the position of the calculated object. Several sources of errors can be at the origin of it: Imprecision due to the numerical calculation, too important speed of rotation and too weak number of FPS (Refreshment of the calculation per second).

Chapter 3

Sensor fusion

Since measurement sensors all have their advantages and disadvantages, the HIVE Tracker embeds different types. We will try later to get the best of each one by an intelligent fusion. There are different types of fusions, we will study two of them: the Kalman filter (KF) and the Madgwick fusion.

3.1 Sensors

3.1.1 Photodiodes

The measurement performed by the photodiodes is very stable over time (no drift), but is also noisy and has a rather low refresh rate. Moreover, the photodiodes are sensitive to occlusions.

The observed jitter requires a preprocessing of the measured data. The use of a simple low pass filter seems to be a solution. We will study two types.

3.1.2 The inertial unit

An inertial measurement unit (IMU) has three types of sensors: gyroscope, magnetometer and accelerometer. We will only deal here with the use of the accelerometer.

To obtain a position, the measured accelerations must be integrated twice with respect to time. However, as the noise around the measurement is not centered, the successive integrations cause the position to drift. A reset of the position must then take place regularly. In addition, the refresh rate of this sensor is higher than that of optical sensors.

3.2 The fusion of sensors

Knowing the qualities of each of the sensors we apply a preprocessing to the sensors: a low pass for the optical data and a double integration for the accelerometer measurements.

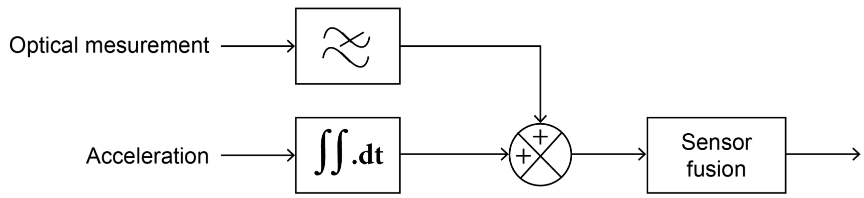


Figure 3.1: Block diagram of HIVE Tracker data processing

We can better understand the interest of preprocessing the measurements and then merging the data with the following schematic. This is a representation of typical data, without precise scale, where the tracker undergoes a step.

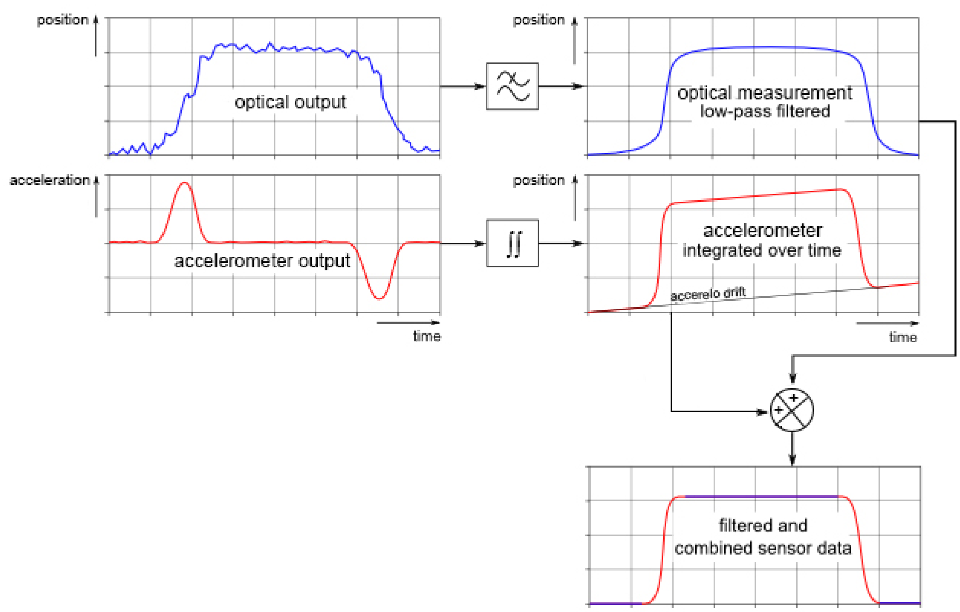


Figure 3.2: Schematic of a step experienced by the HIVE Tracker

3.2.1 Kalman Filter

The first sensor fusion that we will perform is the most common one: the Kalman filter. Assuming that each measurement follows a normal distribution, it is weighted by its standard deviation. The Kalman filter first estimates the position of the tracker using an initial state, then corrects it by weighting it according to the quality of each measurement. The Kalman filter thus allows to estimate the true position of the tracker.

Let's see on a 1 dimensional test how a difference in standard deviation can influence the result of the Kalman filter. We have here two different position measurements (x_light_house for the optical measurement, and $x_accelerometer$ for the measurement made from the IMU) where the units do not matter.

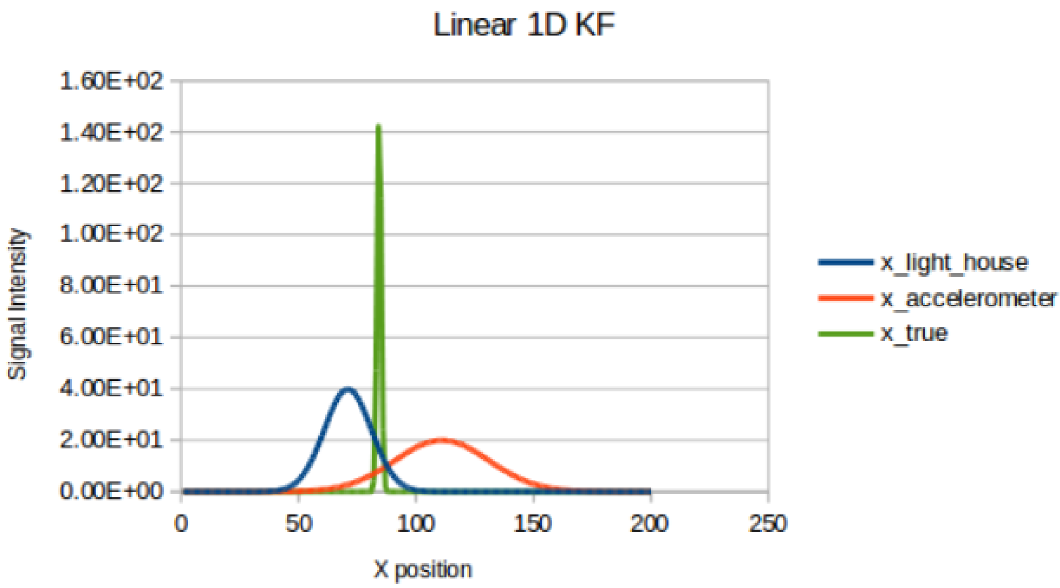


Figure 3.3: Value (*in green*) of the Kalman filter on one iteration from two measurements (*in orange and blue*)

We find the same results as in the literature [1]. The value calculated by the Kalman filter is more accurate than the two separate measurements; and this value is closer to the more accurate measurement.

The details of the use of the Kalman filter are explained in its implementation. The Kalman filter can sometimes require a lot of computing power contrary to some tricks found by Madgwick in his filter.

3.2.2 Madgwick Filter

The Madgwick filter was mainly designed to be used on gyro data. Here we try to adapt it for use on the tracker.

Two types of sensor fusion are proposed in Madgwick's white paper : fusion of accelerometer and magnetometer data by gradient descent, and fusion of the previous data and the gyroscope by a simple weighted average.

Gradient descent

The gradient descent allows to find the minimum of a function. Madgwick, in his publication, looks for the minimum of a function defined with the gravity vector (measured by the accelerometer) and the earth magnetic field (measured by the magnetometer). This function must be null, and a minimization of his result allows to find a very stable gravity vector. The gradient descent is interesting according to Madgwick because it requires little computing power.

Weighted average

A simplified version of the second fusion by Madgwick can be interpreted as a weighted average of the optical and accelerometer data.

Chapter 4

Implementation

4.1 Gimbal lock

The use of Euler angles in a first definition of the tracker rotations showed us a problem: the jump of the system for rotations where the angle with the vertical exceeds 90 degrees. This problem is known in aviation where it has been encountered, it is called gimbal locking. It is solved with the use of another system of rotation: the quaternions.

The quaternions are the generalization of the complex numbers in 4 dimensional space [3]. In our study it will be simply a matter of recovering the rotation sent by the tracker in the quaternion system. A conversion of the quaternion angles into Euler angles for the representation fixes the initial gimbal locking.

4.2 Enclosure design

A solid geometrical structure allows to refine the tracked positioning and orientation. Indeed, knowing the position of 3 points in space we can determine an oriented surface. Moreover, a solid whose relative position of photodiodes is known, allows to apply the principle of positioning explained in 3.2.2.

4.2.1 An enclosure for development

The first step is to create a modular structure. All the components of the tracker must be easily inserted and removed without damaging them. The distance separating the photodiodes from each other will be relatively important (a square of 4 cm side) for the development shell. This distance can be reduced in a second time, in a perspective of miniaturization.

The tracker and photodiodes fit easily into the housing and can be removed without damage. The various connection spaces are left free.

4.2.2 An optimized enclosure

To optimize the positioning of the diodes in space, we need to ensure that whatever the orientation of the tracker a maximum of diodes are visible. The tetrahedron, which maximizes the

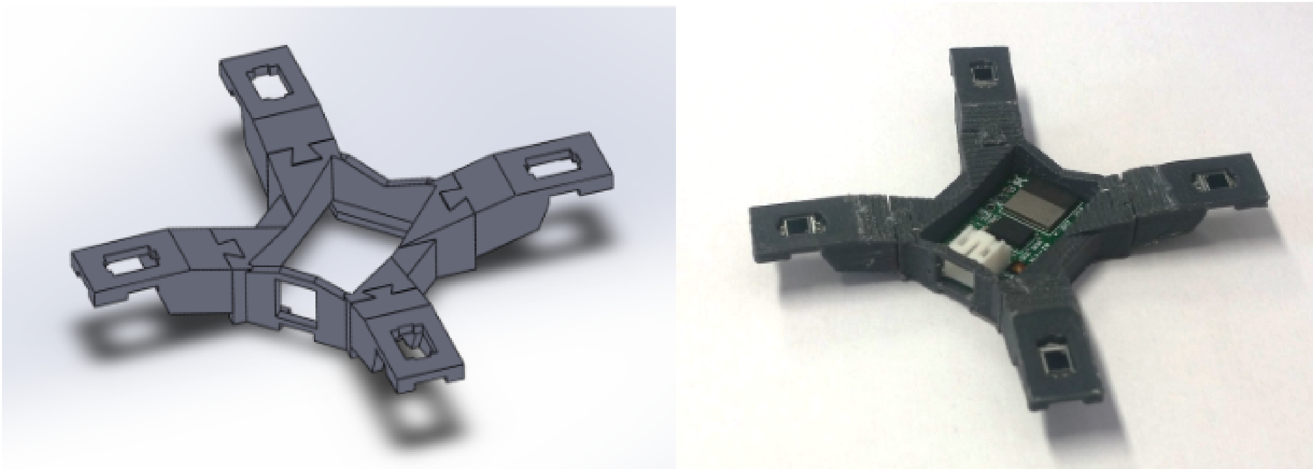


Figure 4.1: SolidWorks enclosure on the left; 3D printed ABS enclosure on the right

space between the diodes, ensures that at least one diode will be visible to the Lighthouses.

Using the previous design, but adjusting the position of the photodiodes we obtain the following hull.

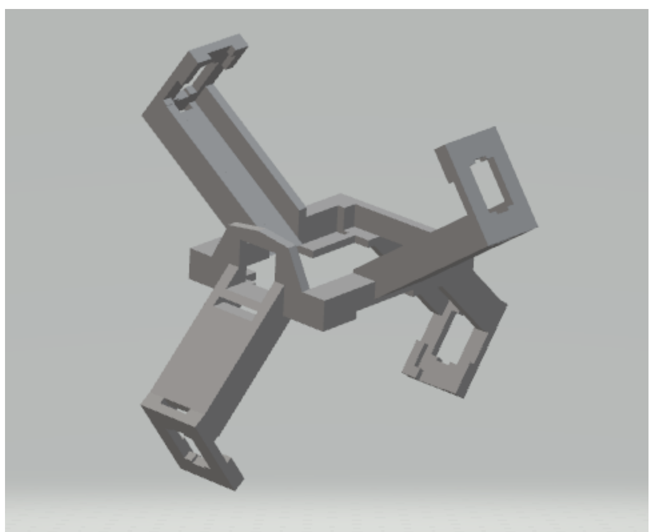


Figure 4.2: SolidWorks tetrahedral enclosure

This arrangement will not be tested in this research.

4.3 Position of the Lighthouses

To know the position of the photodiodes, it is necessary to know before the position of the Lighthouses and their orientations. It is possible to measure them, but this technique lacks precision, as we will see in the first experiment. It is however more interesting to use a second option, to recover the matrices of homogeneous transformations calculated by the HTC Vive system. We will detail this technique here.

4.3.1 Two frames

To recover the matrices of homogeneous transformations, we use a program developed by G. Lopes in Bonsai. After having connected the HTC Vive system in VR and made the initializations, this program allows to recover the 2 matrices of homogeneous transformations of the Lighthouses. These matrices are given in an OpenGL reference frame which is different from the one of Blender.

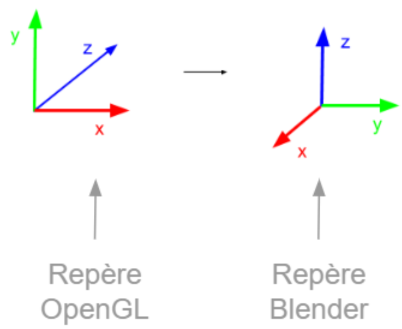


Figure 4.3: Frame difference between OpenGL and Blender

This difference was a rather complex problem to solve, since the application of a simple change of reference frame is not the solution.

4.3.2 Reference frame switch

To find out which change of reference point to apply we have made a Blender simulation. This idea was initiated by G. Lopes. The idea is to apply the homogeneous transformations to the vectors of a base, then to find virtually the position that the Lighthouses have really. Several tests have been done to find the right combination of changes to make.

Considering the initial homogeneous transformation matrix (where the translation vector is on the fourth row) here are the different steps to perform:

- Take the transpose of the homogeneous transformation matrix ;
- Apply the homogeneous transformation to the vector measured by a Lighthouse ;
- Invert the y and z components of the vector.

These changes allow us to define the local reference frame of a Lighthouse that would undergo these transformations. This reference frame is the following:

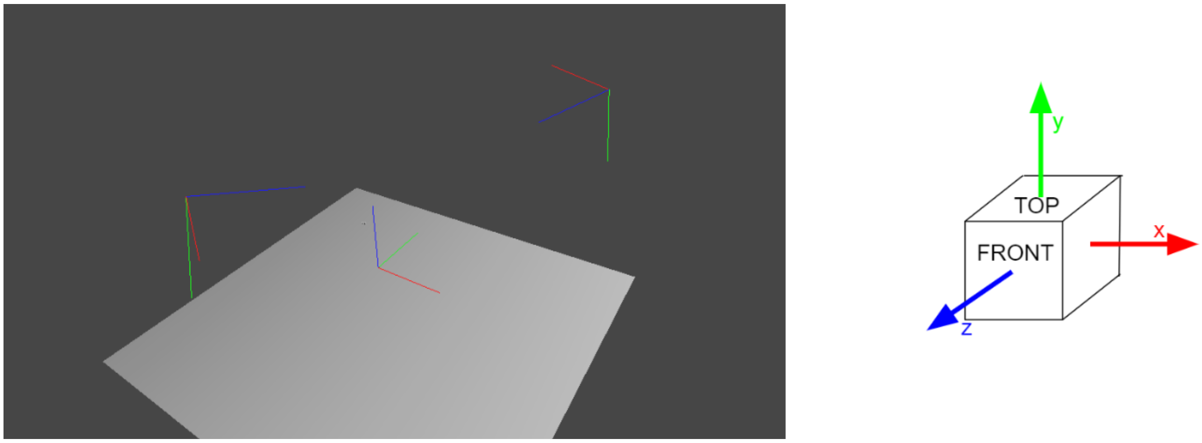


Figure 4.4: Left: Lighthouse landmarks (on the ends) after application of the homogeneous transformation matrices (to the central landmark); right: local landmark of a Lighthouse

4.4 Experimentation

Tests of the tracker on precisely known displacements can be used to characterize it.

4.4.1 The test bench

Theoretically, the HIVE Tracker can return a sub-millimeter position. To prove this, a system with sub-millimeter accuracy is needed.

A first phase of study allows to design a bench which embeds the same technology as the 3D printers: DC 12V power supply, driver for stepper motor, NEMA 17 stepper motor and Atmel microcontroller. A price of 170 euros of material for a precision estimated at a quarter of a millimeter. The purchase phase is interrupted when a similar system can be lent by the Mines-Telecom Institute.

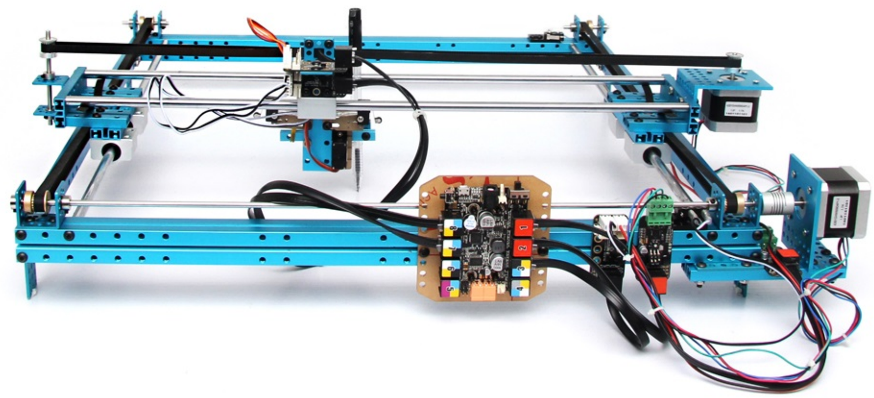


Figure 4.5: Testbed: MakerBot Plotter XY 2.0

The system not being updated since 2014 and not being necessarily adequate to our use, the engineer C. Honnet brings it some modifications. Thus the choice of the movements and their speeds are added to it.

4.4.2 First set of measures

This first experiment is part of the tracker characterization.

Working hypotheses

The position measured by the IMU is reset to 30Hz. The optical position is filtered by averaging the last 4 measurements.

Contrary to the recommendations, the Lighthouses look on the same side. Their locations and orientations were measured by hand. The measurements are summarized on the following diagram.

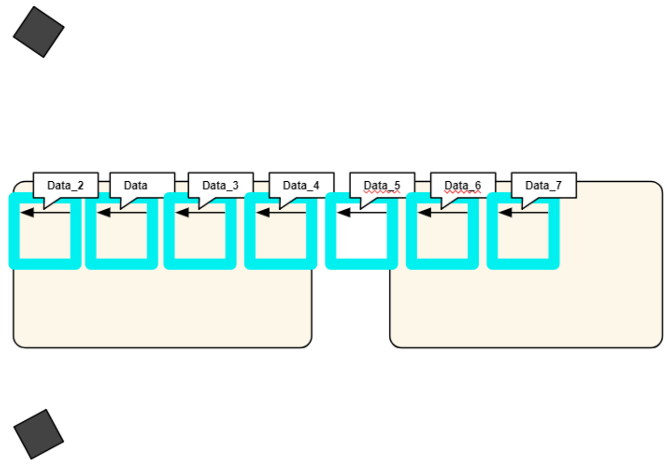


Figure 4.6: Layout of the first experiment (in black the Lighthouses; in beige the table where the Plotter rests; in blue the Plotter; the arrow corresponds to the measurements)

100 measurements that will be averaged over 30cm at each centimeter.

Observation of initial results

We can see that the virtual straight line is not respected. Moreover, the length of one centimeter does not correspond to the measurement made by the Tracker.

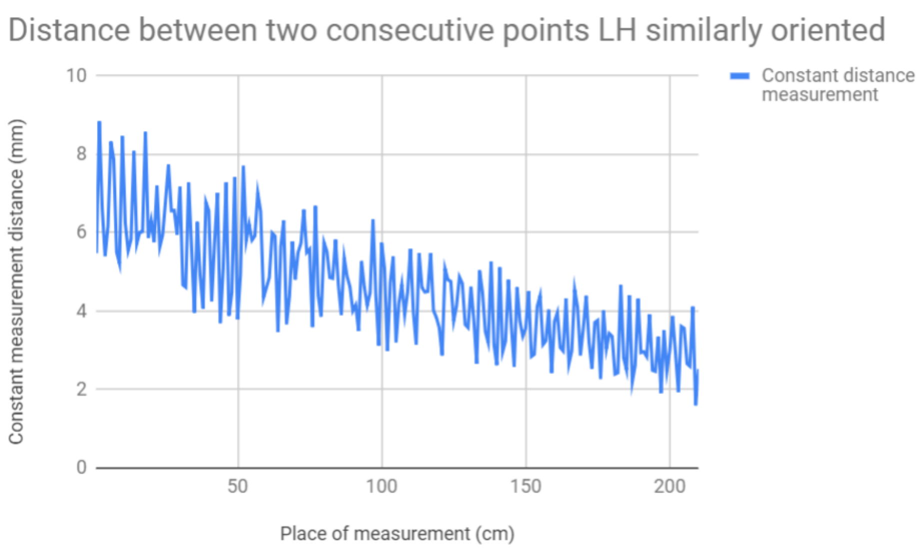


Figure 4.7: Experiment 1: Measurement in space of a displacement of 1cm

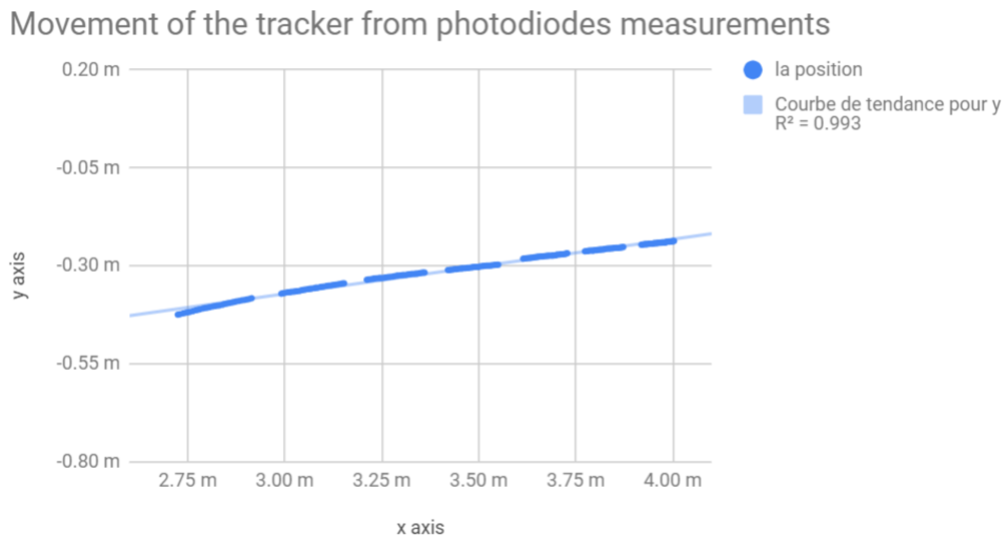


Figure 4.8: Experiment 1: Optical position of the tracker

It can be seen that the further the tracker is from the Lighthouses, the more inaccurate the measurement is. Furthermore, with the trend line, it can be seen that the straight line is almost respected. We can thus explain the measurement error as the result of the inaccuracy of the measurement of the Lighthouses. From now on we will try to use a more accurate location of the Lighthouses.

4.4.3 Second set of measures

New assumptions

The different low pass and sensor fusion filters will be compared.

To improve the accuracy of the measurements we will take here the location matrices of the Lighthouses calculated by the HTC Vive system. We place the Lighthouses so that they face each other.

We reduce the number of measurements to the extreme locations only, as can be seen on the following diagram.

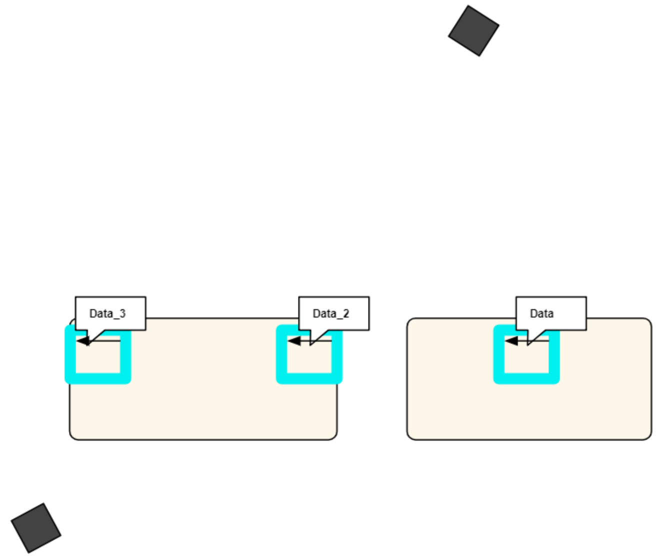


Figure 4.9: Layout of the second experiment (In black the Lighthouses; in beige the table where the Plotter rests; in blue the Plotter; the arrow corresponds to the measurements)

We will also do dynamic tests on sinusoids as a function of time. We will make different sinusoids, oscillating at different frequencies with different amplitudes.

Observation of the second results

The same study as before is performed with the change of positioning of the Lighthouses.

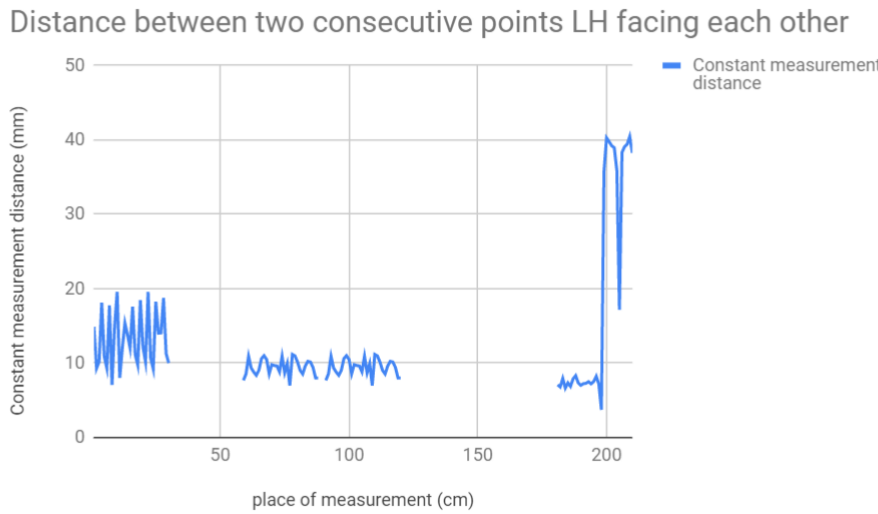


Figure 4.10: Experiment 2: Measurement in space of a displacement of 1cm

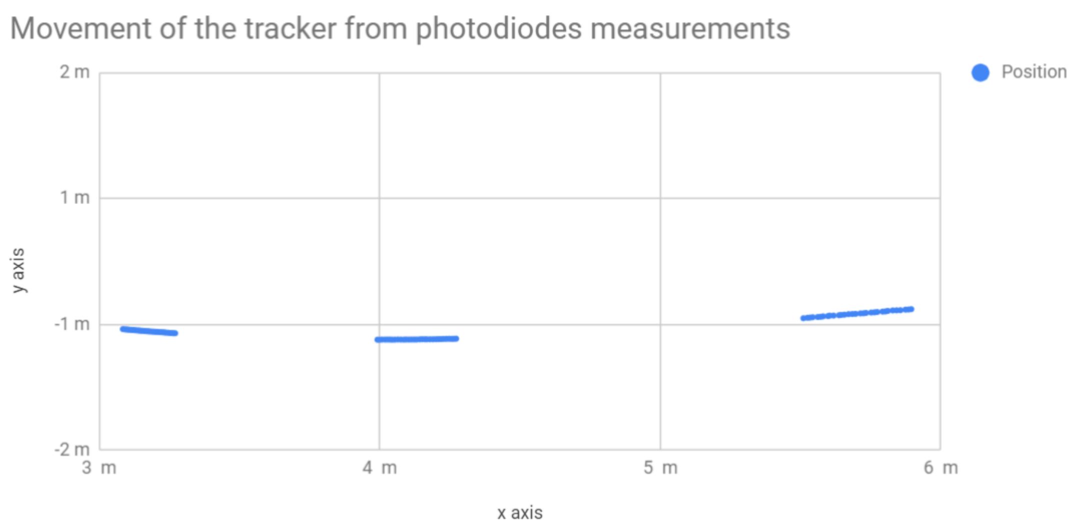


Figure 4.11: Experiment 2: Optical position of the tracker

We can see that with the new definition of the Lighthouses locations, the distance of 1 cm is better respected. However, this result is to be qualified for the extreme values where we can notice inconsistencies of measurements. Moreover, we can notice a slight curvature of the calculated space. Indeed, the measured data "data-2" show a slight curvature. An idea to understand this phenomenon and try to solve it will be proposed later.

4.4.4 Alternative excitation on the second series of measurements

The second set of measurements is more reliable than the first, so we will try and compare the filters on the measurements here. The excitation is of amplitude 5 cm and time period 1.8 s.

We place ourselves at the level of data-3.

Low pass filter on optical data

Here we compare the two low-pass filters of the optical data: average of the last 4 measurements and some percentages of the last measurement on the old ones.

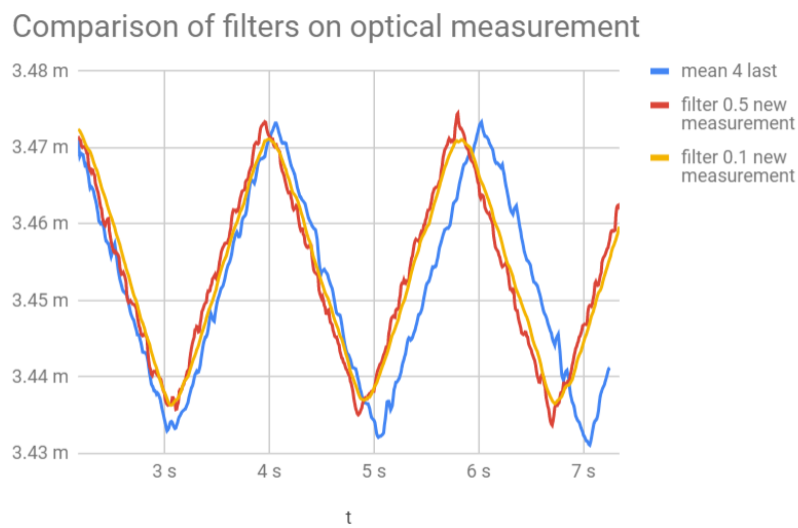


Figure 4.12: Experiment 2: Comparison of filters on optical data (in blue: average of the last 4; in red: the new measurement counts for 50 percent on the old ones; in yellow: the new measurement counts for 10 percent on the old ones)

Measured amplitudes:

- Average of the last 4 values: 40 cm ;
- Filter at 0.5: 35 cm ;
- Filter at 0.1 : 33 cm.

We first notice that the sinusoidal shape of the plotter should describe is not respected. But the analysis we will do will not take this into account. The shape remains constant in time but the amplitudes are not respected.

Kalman filter compared to Madgwick

We still compare measurements on a reciprocating motion. But here we look at the difference made between using the Kalman filter versus a filter that requires less computing power, the Madgwick technique. The optical data are filtered by using the average of the last 4 measurements.

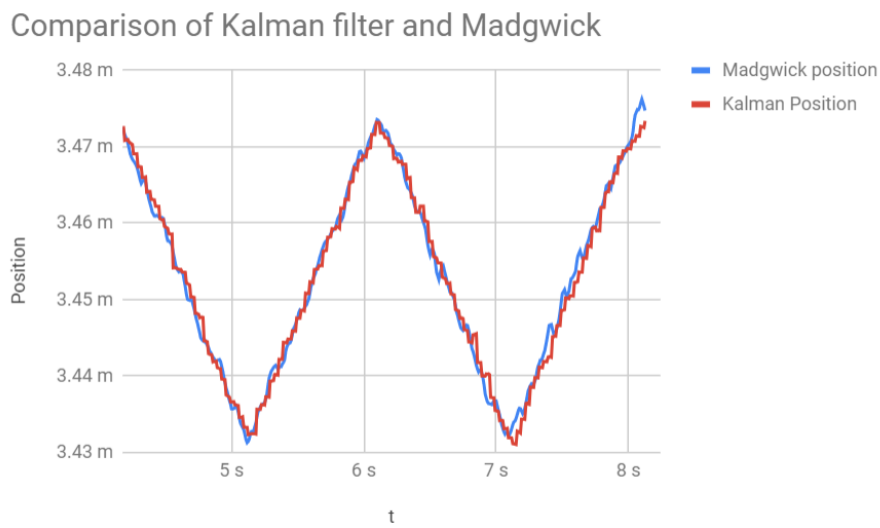


Figure 4.13: Experiment 2: Comparison of Kalman fusion and Madgwick fusion

We see that the Kalman data are jerky in steps. The Madgwick data are smoother. Both measures are noisy, and have the same amplitude and time periods.

4.4.5 Accelerometer data

The theoretical refresh rate of the accelerometer data is 4 times higher than that of the optical data. However, the drift of the accelerometer is such that very quickly the measured data are false. Let's observe the Kalaman filter acting statically for a 0.5 seconds drift of the accelerometer.

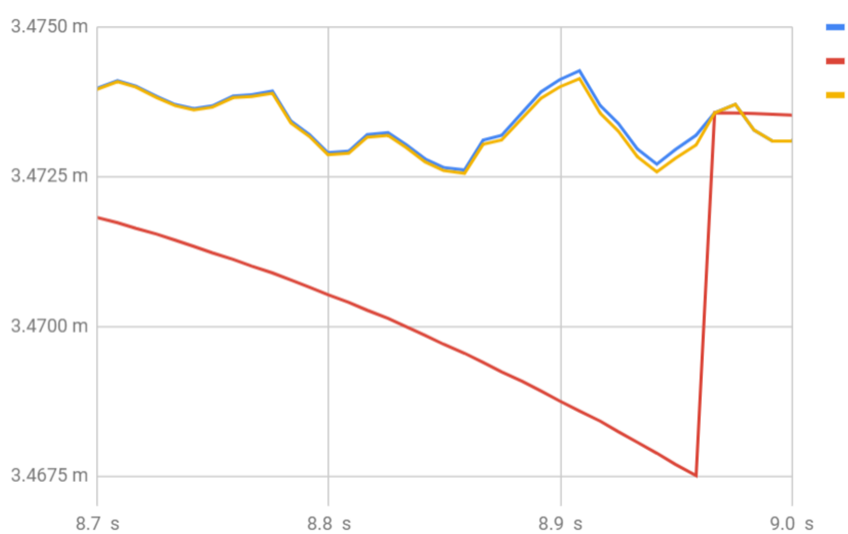


Figure 4.14: Experiment 2: optical measurements (blue), accelerometer measurements (red) and filtered data (yellow) as a function of static time

We notice that the accelerometer data pull the filtered data in their direction. We also see that the optical values are predominant for the Kalman filter.

Chapter 5

Discussion

5.1 Characterization

5.1.1 Characterization of the optical positioning

The tracker characterization shows that the position measurements with the Lighthouses depend on the location where they are made. Thus, for best accuracy, an intermediate distance to the Lighthouses should be chosen: not too close, not too far. This distance is about 2 meters from each Lighthouse.

5.1.2 Characterization of filters

Low-pass filter

The 0.1 filter seems to be the most suitable with the tests performed. Even if it adds a response time, the stability of the tracker is largely increased, and empirical tests show that this latency is slightly compensated by the refresh rate of the measurements. This filter is also very simple, and requires little computing power.

Sensor fusion

The Kalman and Madgwick filters have globally the same result. However the Kalman data being jerky, the Madgwick filter will be preferred. Indeed, its result seems smoother for a low computational power required.

Moreover, considering the drift of the accelerometer data, a reset of its position every 4 measurements seems to be sufficient not to impact too much the merged position.

5.2 Improvements

The whole course has brought to the fore some limitations. Here are some points that I think are interesting to notice.

5.2.1 Visualization

Even if Blender has the advantage of being free and multi-platform, its disadvantages of use are not the least.

At first, it is difficult to get to grips with Blender. This software is not intuitive and a long learning curve is required. Then its coordinate system is not the same as OpenGL, additional calculations are required at each step. Finally, the community using Blender's internal game engine (BGE) is in strong decline. So much so that it has been announced that the current BGE will be phased out.

All this argument shows that Unity is a better development platform for the HIVE Tracker. Indeed Unity can run on all OS (even on smartphones) in a free version; the coordinates are the same as on OpenGL; the community is very active.

5.2.2 Positioning

The Tracker single Lighthouse localization technique looks very promising.

Even if the first tests were not conclusive, it would first allow a localization of Lighthouses. It would thus free us from the complex initialization of the HTC Vive system, while having a good accuracy. Then, this localization would also be a means of resistance to obstructions. Because even with a hidden Lighthouse, the Tracker would still be visible.

A more intelligent optical data filter could be tested: the 1 Euro filter. This is a low-pass filter with a variable threshold. For the Kalman filter, an adjustment of the standard deviation of the measurements could be performed.

For the accelerometer, the drift seems to be constant, so we could subtract a coefficient to minimize. And using the orientation of the tracker would allow to refine the measurement of accelerations in space.

5.2.3 Space visualization

To understand the phenomenon of curvature of the calculated space, a study should be conducted. It would be to measure the position of the tracker by squaring the space. It would be a question of finding the zones of space where there are inconsistencies to then solve the problem. We could then either compensate for the discrepancies at each point, or redefine the positioning matrices of the Lighthouses. The first solution being tedious, the second would be preferred.

5.2.4 The enclosure

Two main improvements are to be added to the enclosures.

The first one is in the protection of the components of the card. Indeed, the first version trying to have a maximum of freedom showed a flaw. A plug of the link broke during the numerous uses of the tracker. It would then be a question of reinforcing the connectors in these zones solicited by light efforts.

The second improvement would come from the perspective of miniaturization of the tracker. It would be a question of reducing the space between the components as much as possible, while keeping the current practicality.

Conclusion

This internship allowed me to explore an engineering problem in the research world. Even if the problem I worked on is theoretically solved, it is difficult to implement and has many imperfections that can be improved. Some of the tasks turned out to be shorter than expected, and the whole study plan ended up taking longer to complete than the initial Gantt chart predicted.

I discovered how research is done internally in a laboratory, and how it is shared externally. I could see that the actual time spent working is rarely proportional to the amount of work done. Moreover, at the technical level I was able to learn some software such as Blender and Unity. I was able to deepen my knowledge in the management of code versions with GitHub. I was able to discover data fusion, quaternions and gradient descent. I was able to use my technical background. I also had to learn to popularize my ideas to better transmit them.

This internship also confirmed my interest in robotics research. Technical skills in mechanics, computer science and electronics are essential. And the sharing of new ideas based on previous work seems to me just as interesting.

Finally, even if the initial specifications were ambitious for a trainee, almost all of them have been achieved. This study has also highlighted the amount of work that could still be done to have a Tracker resistant to disturbances and easy to use. It is in this context that a proposal of continuity of my work in Master project was made to me. It would be to work for NeuroGears, a spin-off of the UCL whose direction will be taken by G. Lopes.

Bibliography

[1] John W. Dower *Probabilistic robotics 40.* 2005.

[2] Sinan Haliyo *Blender pour Simulation Physique*

[3] Andrew J. Hansen *Visualizing Quaternions* 2005.

[4] David Filiat *http://perso.ensta-paristech.fr/~filliat/Courses/index.html*

[5] *https://hivetracker.github.io/*

[6] Sebastian 0.H. Madgwick *An efficient orientation filter for inertial and inertial/magnetic sensor arrays* 2010.