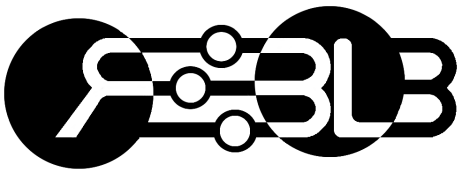

HapticPuncture Simulator manual

TM/ROBOLABO/2016

Developed by **ROBOLABO**

www.robolabo.etsit.upm.es

Authors:
Juana González-Bueno
Dr. Alvaro Gutiérrez Martín

July 15, 2018

Contents

0.1 Introduction

HapticPuncture Simulator is an innovative simulation system for epidural anesthesia training. Among its many advantages it is important to say that it is virtual-reality (VR) based, which means that can be used indefinitely. Moreover, it is a low-cost, open-source, customizable prototype that opens doors to new procedure simulators in multiple specialties of medicine.

This guide allows to any user to install the simulator downloaded from `www.github.com/robolabo`, and to start using the proposed simulator in **Ubuntu 14.04**. Furthermore, instructions about how to create new applications are detailed.

HapticPuncture Simulator is composed by two main modules: an haptic device and a virtual environment. In order to use this system, some requirements exist.

0.2 Materials

- A computer with Ubuntu 14.04
- An Arduino Due board
- A motor's board X-NUCLEO-IHM04A1 available online
- A voltage generator
- A DC motor MINIMOTOR 2842-012C
- An encoder HEDS 5540 A
- 3D printed structures, that can be found in the /3D-structures folder.
- Two Linear Shaft Rail Bars available online
- Two bearings available online
- A transmission wire
- A capstan
- A pulley

0.3 Installation

0.3.1 CHAI3D

First of all, HapticPuncture Simulator's virtual module is based on CHAI3D. It is therefore necessary to install this software platform on your computer.

To do that download the current CHAI3D software version from http://www.chai3d.org/download/releases. Choose the file named "chai3d-version-multiplatform.zip". For this simulator version 3.1.1 was used.

Unzip in the desired work folder.

Open the terminal and go into the unziped folder. Execute:

```
$ make
```

For this to work it is necessary to have previously installed:

- GLUT package:

```
$sudo apt-get install freeglut3 freeglut3-dev
$sudo apt-get install binutils-gold
```

- libusb-1.0-0-dev

```
$sudo apt-get install libusb-1.0-0-dev
```

- libasound2-dev

```
$sudo apt-get install libasound2-dev
```

- Moreover it is necessary to have a compiler installed. Build-essentials is enough:

```
$sudo apt-get install build-essentials
```

For more information, read the CHAI3D documentation in the /doc folder.

0.3.2 Arduino

As the haptic device uses an Arduino Due, it is necessary to install the Arduino IDE. To do that follow the next steps:

Download the desired version of Arduino for Linux from Arduino. The version used in this prototype is arduino-1.6.5-r5.

Unzip

Go to the folder using the terminal and execute:

```
$ cd arduino-version
$ ./arduino
```

Some libraries are required to use the code given in this project. For the purpose of this simulator, Encoder.h and DueTimer.h are needed. To install them download the library zip, from Encoder library and DueTimer library respectively. Then, open the Arduino IDE (explained just before) and select:

Program → Include Library → Add .ZIP library

and then select the zip forlder that contains the desired library. Close the program and execute again.

0.4 Start the system

The C++ code to run the virtual environment needs to be compiled. Firstly, copy the "07-HapticPuncture" directory from HapticPunture directory */Chai3D_code*

to
CHAI3D_directory */GEL/modules/GEL/examples/*.

Then, run in the command line:

```
$ cd <HapticPunture directory>/Chai3D_code/07-HapticPuncture
$ make
```

Next, to run HapticPuncture the following code scripts need to be run:

- Arduino_code/HapticPuncture_arduino.ino in arduino
- Chai3D_code/07-HapticPuncture/obj/release/lin-x86_64-ccHapticPuncture/HapticPuncture by typing in the terminal from lin-x86_64-ccHapticPuncture directory:

```
$ ./HapticPuncture
```

0.5 Customize your models

To either use different tissue models or to modify the parameters to perceive different haptic feedbacks, you will need to modify the file "HapticPuncture.cpp" inside Chai3D_code/07-HapticPuncture.

0.5.1 Using different models

First of all, you will need to add your new models inside the folder ressources/models/. Note each tissue has to be in a different file if you will want to assign different parameters to each of them. Note as well that the models have to be in ".stl" format, and shouldn't contain a high number of vertices, as the haptic rate will decrease.

Secondly, inside "HapticPuncture.cpp" you will need to modify the path to each of your models. Inside the model section, modify the code to load the model.

0.5.2 Modify your model's parameters