

Budapest University of Technology and Economics

Department of Telecommunications and Media Informatics

Robot position estimation using deep neural network with IMU data

Deep learning in practice based on Python and LUA

(BMEVITMAV45)

Homework

MIRA Team:

Balázs Nagy – o668s4

Márk Fekula -

Gergely Hunyady – FPZXIL

Budapest, 2018. 10. 11.

Table Of Content

[1. Project description 2](#_Toc526843273)

[1.1. Main goals 2](#_Toc526843274)

[2. Literature overview 3](#_Toc526843275)

[3. Data Collection 4](#_Toc526843276)

[4. Data Preparation 5](#_Toc526843277)

[4.1. Data understanding and visualization 5](#_Toc526843278)

[5. Modelling 6](#_Toc526843279)

[6. Hyperparameter tuning 7](#_Toc526843280)

[7. Summary 8](#_Toc526843281)

[7.1. Results 8](#_Toc526843282)

[7.2. Further development potentials 8](#_Toc526843283)

Table of Figures

**No table of figures entries found.**

# Project description

## Main goals

The aim of this study is to examine if it is possible to estimate a robot position from IMU (Inertial Measurement Unit) data. During our research we are using a Phidget Spatial sensor.

# Literature overview

# Data Collection

Collecting data is an essential part in deep learning era. In order to make enough measurement data we built up our own measurement setup which consist of three main parts. The client program can connect to the Motion Capture system to track and log the position and orientation of the sensor mounted on the robot. It also connect to the robot via wifi to control the movement of the robot. And the program connects the IMU (Inertial Measurement Unit) via USB to collect and log data from it. The sematic of MIRA (MoCap, IMU and Robotino Analyser) can be seen on Figure 2.

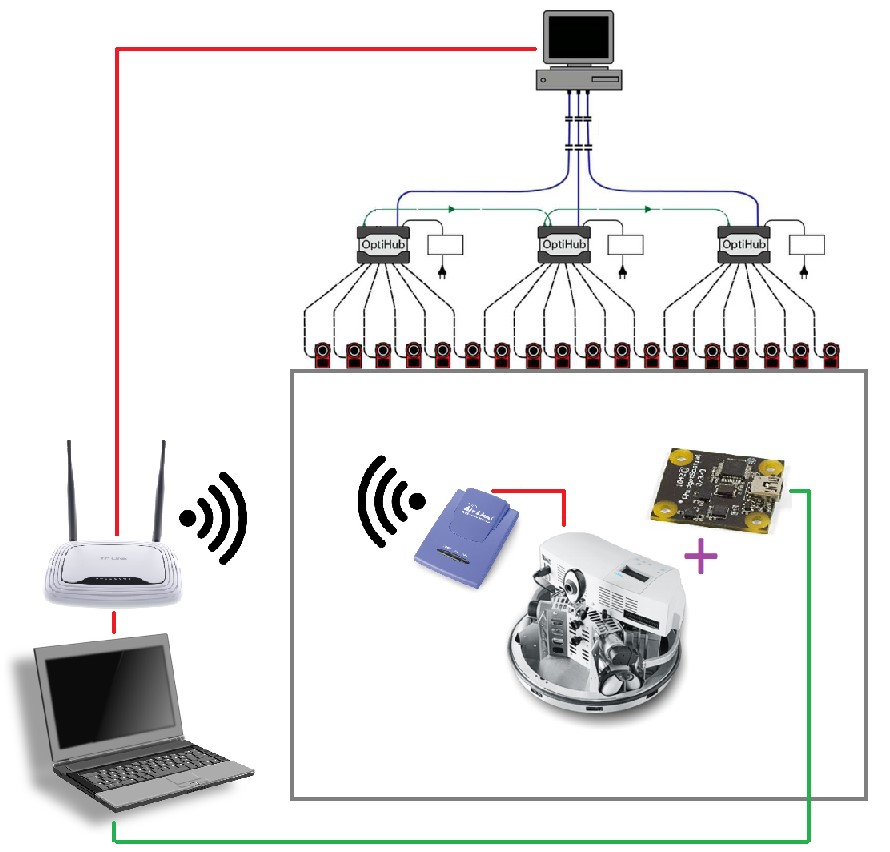


Figure : Sematic of MIRA

The following sections contain a detailed description about the three main parts of MIRA.

## IMU

A PhidgetSpatial 3/3/3 sensor, shown in Figure 2, had been used during our study.

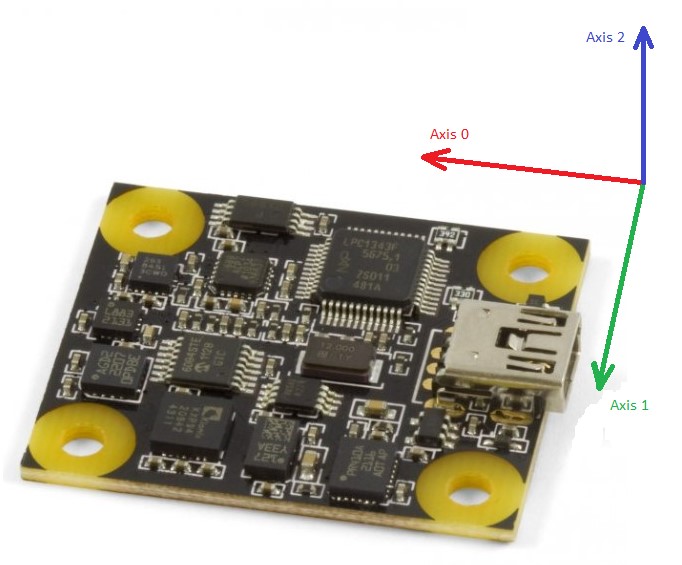


Figure : Phidget Spatial 3/3/3 sensor

The sensor contains two 3-axis accelerometers, two 3-axis gyroscopes and a 3-axis magnetometer. The exact parameters of the sensor can be found on the official website of the sensor [1]. The sensor is mounted on a carrier robot and plugged to the notebook via USB cable. The USB cable connection is necessary because the robot platform is not compatible with and it is the most secure way to collect data from it.

The data structure from the inertial measurement unit (IMU) is the following:

* acc0 – acceleration along the X axis [g]
* acc1 – acceleration along the Y axis [g]
* acc2 – acceleration along the Z axis [g]
* gyro0 – gyroscope speed around the X axis [°/s]
* gyro1 – gyroscope speed around the Y axis [°/s]
* gyro2 – gyroscope speed around the Z axis [°/s]
* mag0 – magnetic field strength along the X axis [G]
* mag1 – magnetic field strength along the Y axis [G]
* mag2 – magnetic field strength along the Z axis [G]

The different built in sensors have a different sampling speeds. The maximum sampling speed of the accelerometer and the gyroscope is 4 ms/sample, while the maximum sampling speed of the magnetometer is 8 ms/sample. The sampling time can be modified and in case the sampling speed is faster than the magnetometer a zero order hold filter is applied on the magneto data. In this way we eliminate the harmful effect of the corrupt measurement points.

## Robotino

However, the aim of this study to build up an algorithm which only uses data from the IMU, so it can be platform independent we used a Robotino during our project. Mounting the sensor on a real robot provides real data sets and in this way our algorithm will learn on a real-life use case scenario.

The Robotino -shown on Figure 2- is a small robot with omnidirectional driving system made by FESTO. The robot has its own programming language, but it is not compatible with other languages, so we had to write an API in LabVIEW. With the help of the LabVIEW code we gain access to every data produced by the Robotino, furthermore LabVIEW makes it possible to connect the robot to the Motion Capture system.



Figure : Robotino

# Data Preparation

## Data understanding and visualization

# Modelling

# Hyperparameter tuning

# Summary

## Results

## Further development potentials

# References

|  |  |
| --- | --- |
| [1] | P. Inc., “Phidgets,” 2017. [Online]. Available: https://www.phidgets.com/?&prodid=32. [Accessed 11 10 2018]. |