
WearLoc: A Wearable Indoor Localization Device

Lukas Gemein

gemeinl@cs.uni-freiburg.de

Jennifer Nist

nistj@cs.uni-freiburg.de

Rick Gelhausen

rick.gelhausen@gmail.com

David Speck

speckd@cs.uni-freiburg.de

Andre Biedenkapp

biedenka@cs.uni-freiburg.de

Introduction

Firefighters and other rescue teams often have to examine buildings or tunnels to find and save survivors of tragic events. Therefore they need to enter unknown environment where it can be hard to not lose the orientation. Even if the rescue team has a map of the building, it can be difficult to share the position of a located survivor with other members of the team, since ways can be blocked or impassible due to destructions. It would be desirable to create maps of the surroundings that are up-to-date, which could make the work for rescue teams much easier and safer.

Another area of application for a wearable mapping device would be the exploration of caves to simplify and secure the work of cave explorers.

In the lab course "Wearable Computing Systems" we created two versions of a wearable device to create live-maps. The main difference of the versions lie in the hardware we used. Our first version uses a Raspberry Pi and the RPL-IDAR laser scanner, which makes it a cheap and easy to rebuild system. In the second step we created a handier and even more powerful version using an Intel Edison and a Hokuyo laser scanner.

Hardware

Raspberry Pi

The Raspberry Pi 2 Model B features a 900MHz quad-core CPU and 1GB of RAM. We use the Raspberry Pi in our first setup to run the entire software for IMU and laser scanner as well as a SLAM approach. Since the Raspberry Pi lacks a display we send the results via Wi-Fi to an external device to visualize the map.

Intel Edison

The Intel Edison features a 500MHz CPU and 1GB of RAM, which we use in our second version. It also runs the software for IMU and laser scanner but sends the data to an external device which runs the SLAM approach and visualizes the data.

Laser Scanner Robo Peak (RPLIDAR)

The RPLIDAR is a 360 degree 2D laser scanner. The scan range is 0.2 to 6m with a scan rate of 5.5Hz. Due to its relatively large rotating part and the associate motor, the RPLIDAR is quite big for a wearable device, having a surface of about $70 \times 100 \text{ mm}^2$. We use it in our first setup mounted on a one-handed portable platform.

Laser Scanner Hokuyo (URG-04LX)

The Hokuyo URG-04LX laser scanner has a 240 degree measurement angle and offers a scan rate of 10Hz. The scan range is 0.02 to 4m. With its $40 \times 40 \text{ mm}^2$ surface it is ideal to be mounted on a GoPro camera holder, which allows several different wearable designs.

Grove IMU

The Grove IMU 9DOF v2.0 features a gyroscope, a magnetometer and an accelerometer. We use the Grove IMU in both our setups to measure the bearing of the laser scanners to track user rotation. Thus, it helps to better predict the location.

Software

Both the Raspberry Pi and the Intel Edison run the Robot Operating System (ROS), which is an open source and free to use software. Our project is based on a ROS package which implements an approach for indoor SLAM. The package is called *hector_slam*[4] and was implemented by Team Hector of the University of Darmstadt. We also use the already existing packages *i2c_imu*[3], *rplidar_ros*[5] and *urg_node*[2] and adapted them to our needs to conveniently use the IMU as well as the laser scanners. Additionally, we uploaded a binary blob[1] onto the IMU to directly convert the raw measurements to quaternions which were then forwarded to the SLAM implementation. A Wi-Fi network connection was set up to wirelessly transfer data between the wearable and the external device used for visualization (Figure 1).

Wearbale v1

In our first version we used the Raspberry Pi, the IMU and the RPLIDAR laser scanner. We chose to start with the Raspberry Pi, since it offers an easy start due to its numerous ports to connect other devices. The IMU was connected using the I2C GPIO pins of the Raspberry Pi whereas the laser scanner could easily be connected through a USB port. This fast setup allowed us to get familiar with the use of the IMU and the RPLIDAR, have a look at raw data and already start adapting the code. Finally, the Raspberry Pi runs all Software components itself, including the packages for IMU and RPLIDAR as well as the SLAM approach. The map is wirelessly transferred to a laptop which is used for visualization. All the hardware components are mounted onto a wooden plate which is screwed on a aluminium table leg for the purpose of holding the device (Figure 2). A powerbank with 2.1A powers the system. Since the Raspberry Pi and the laser scanner are relatively big this version is not quite as handy which is why we created a

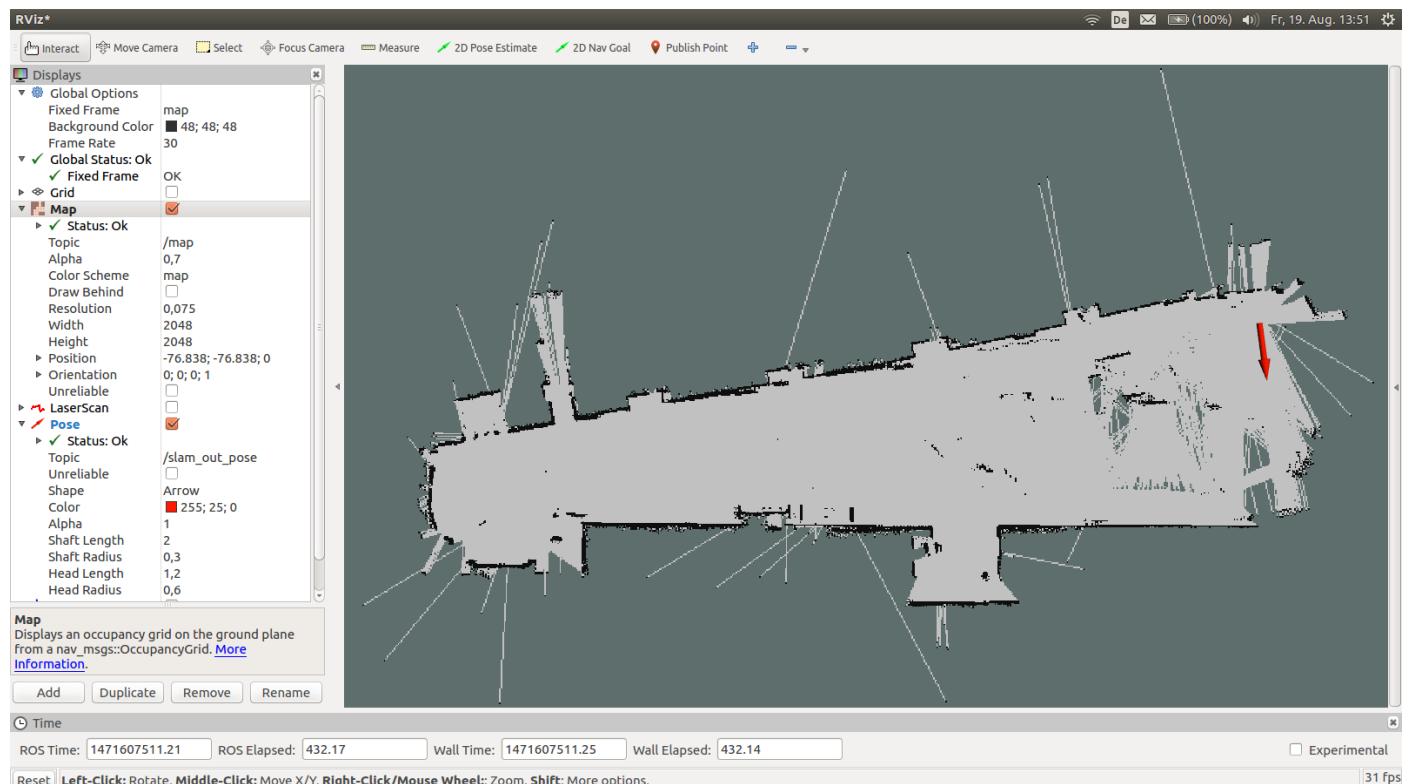


Figure 1: Live visualization of a SLAM run in rviz.



Figure 2: Wearable v1 featuring the Raspberry Pi and the RPLIDAR.

second version. Note, that apart from small adjustments the Raspberry Pi and the Intel Edison are interchangeable.

Wearable v2

The Intel Edison does not offer GPIO pins or USB ports to directly connect our sensors which is why we designed a circuit board that holds the Intel Edison and offers the appropriate ports. The making of the board took some time, which is why we already started developing the software using the Raspberry Pi (Wearable v1). Since the Hokuyo laser scanner is very small, it is better suited for a wearable device. In contrast to the Raspberry Pi, the Intel Edison runs a lightweight version of ROS which was built from source. Due to memory limitations we only set up the packages `i2c_imu` and `urg_node` necessary for the sensors. The SLAM approach as well as the visualization run on a laptop that receives the sensor data wirelessly. The Hokuyo laser scanner is small enough to be mounted on a 3D printed platform that can be attached to GoPro camera holders. The IMU is attached to the bottom of the platform. As there is an enormous amount of different GoPro camera holders,

this allows us to attach the laser scanner and the IMU in many different ways e.g. as a head mounted gear similar to a helmet lamp which leaves both hands free for other tasks. This system is powered by a 1A powerbank. We also created a 3D printed box, s.t. the Intel Edison can safely be transported and used in the pockets of the wearer.



Figure 3: Wearable v2 featuring the Intel Edison and the Hokuyo.

Results

The presented wearable devices yield good mapping results. Because of its higher scan range of 5.5m, the RPLIDAR outperforms the Hokuyo with regard to the SLAM results. It returns more reference points which helps to better estimate the location. Both the RPLIDAR and the Hokuyo laser scanner have a low scan rate leading to bad results during fast movements. In our opinion, the results can be improved significantly with a

more powerful scanner that overcomes the drawbacks of low scan rate and range. Nevertheless, sticking to the laser scanner limits the presented wearable devices can be used for simultaneous localization and mapping as seen in Figure 4. It shows a map that was generated with the first version of our wearable device and afterwards matched to the groundplan of the building. The map fits the groundplan very well and realistically reflects the building. Thus, both provided wearable SLAM devices can be used to construct meaningful representations of the environment.

REFERENCES

1. Forde C. 2014. mpu9250 quaternion blob. Binary blob. (2014). Retrieved August 25, 2016 from <https://github.com/Snowda/MPU9250>.
2. Rockey C. 2013. urg_node. ROS package. (2013). Retrieved August 25, 2016 from http://wiki.ros.org/urg_node.
3. Eskesen J. 2014. i2c_imu. ROS package. (2014). Retrieved August 25, 2016 from https://github.com/jeskesen/i2c_imu.
4. Meyer J. Kohlbrecher S. 2012. hector_slam. ROS package. (2012). Retrieved Mai 30, 2016 from http://wiki.ros.org/hector_slam.
5. Robopeak. 2014. rplidar. ROS package. (2014). Retrieved August 25, 2016 from <http://wiki.ros.org/rplidar>.



Figure 4: A groundplan of building 51 (Technical Faculty, University of Freiburg) combined with a map generated by the wearable device (v1).