Simulation of a Mobile Robot Localization based on Hierarchical Sensor Fusion

Carlos Eduardo Magrin[1,2], **Eduardo Todt**[1]

[1]Laboratory of Vision, Robotics and Imaging (VRI)
Department of Computer Science – Federal University of Paraná (UFPR)
Curitiba, Brazil

[2]Laboratory of Robotics and Embedded Systems (ROSIE)
Department of Mechatronics – UNISOCIESC
Curitiba, Brazil

carlos.magrin@unisociesc.com.br, todt@inf.ufpr.br

Abstract. *This article presents a hierarchical sensor fusion (HSF) method applied in virtual robot experimentation platform (V-REP) for indoor mobile robot localization. Robot operating system (ROS) framework is used for interface with V-REP. The proposal is to simulate a robot equipped with low-cost sensors such as eight sonars and digital compass, also using the received signal strength (RSS) from wireless networks available in the environment. The HSF method based on fingerprint kNN can determine the robot localization in different grid sizes. This work was useful for modeled a robot in a virtual scene and segmentation the project in nodes for supporting a sensor fusion method and solve the mobile robot localization problems in indoor environments.*

1. Introduction

In mobile robotics, many sensors are used for perception the indoor environment, but each one has different features about the perception. As Tzafestas [Tzafestas 2014] described, sensor fusion is the process of merging data from multiple sensors in order to reduce the amount of uncertainty that may be obtained with the use of just one sensor.

The localization problems are classified by Siegwart, Nourbakhsh, and Scaramuzza [Siegwart et al. 2011] in three types, which are: *position tracking* - the robot current location is updated based on the knowledge of its previous position; *global localization* - assumes that the robot previous location is unknown; and the *kidnapped robot* - similar to the global localization problem, but the robot realizes having been kidnapped. The main goal of this paper is to solve these three localization types of problems in indoor environments, applying a hierarchical sensor fusion (HSF) method, based on fingerprint kNN (k-Nearest-Neighbor). The research was performed with a virtual mobile robot, based on simulation information gathered from low-cost sensors, specifically multiple sonars, a compass, and a wireless network received signal strength (RSS) sensor.

Other previous works already used fingerprint kNN method for sensor fusion in indoor localization. Torteeka and Chundi [Torteeka and Chundi 2014] proposed a simulation model using dead reckoning and RSS value based on fingerprint technique with fuzzy kNN. [Magrin and Todt 2016] developed a mobile robotic platform with eight sonars, placed in an octagon pattern, with invariance to orientation, using RSS information acquired from access points (AP's) available in the environment, performing the location estimation based on fingerprint kNN and fuzzy features weighting.

Sensor fusion in the mobile robotic field includes others methods for combining information, such as: weighted average, Kalman filtering, particle filters, Bayes estimation, Dempster-Shafer method, Rough set, neural networks, fuzzy logic, behavior-based algorithms, rule-based techniques, and hybrid fusion, as described in [Khaleghi et al. 2013], [Zhao and Luo 2008], [Kam and Kalata 1997].

The organization of the paper is as follows. After the introduction, fundamental concepts of fingerprint kNN, ROS, and V-REP are presented, just as introduces a real mobile robot platform and its corresponding model in the virtual platform V-REP. Furthermore, a hierarchical sensor fusion with fingerprint kNN is presented in section III. Section IV shows simulation results aiming to validate the method. Finally, section V concludes the paper.

2. Frameworks and Robot Model

In this section, we presented the concepts of localization using fingerprint technique, frameworks services with the Robot Operating System, and Virtual Robot Platform based on a distributed control. The mobile robot model used for simulation is presented for indoor localization using sensor fusion.

2.1. Fingerprint kNN

Localization using fingerprint technique has the goal of finding the relationship of a place in a map, and the localization of a mobile robot, based on a comparison of signatures correspondent to each location. This technique basically has two main steps: *training* and *positioning*. In the training, the robot collects data from sensors in each position and save them in a training database. In the positioning step, the robot does the sensors reading and matching with the training database, determining the localization in a map [Torteeka and Chundi 2014], [Kunemund et al. 2009].

The relevant research on the positioning using fingerprint technique is reported in numerous works, e.g., [Magrin and Todt 2016], [Torteeka and Chundi 2014], [Yang and Zhang 2014], [Liang et al. 2012], [Kunemund et al. 2009]. While the kNN (k-Nearest-Neighbor) matching algorithm was chosen for being one of the most widely used, in this field the most commonly used metric in measuring the distance between a new data vector, and those of the training database is the Euclidean distance, as in (1), described by [Mitchell 1997], [Webb and Copsey 2011].

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{1}$$

2.2. Robot Operating System (ROS)

Robot Operating System (ROS) is an open source framework for writing robot software, it simplifies the task of dealing with a wide variety of robotic platforms. As described in [Quigley et al. 2015] approximately 80 commercially available robots are supported, and we can find more than 1850 academic papers that mention ROS. Although ROS is not an operating system, it provides services designed for hardware abstraction, low-level control, high-end capabilities, as well as packets with tools for debugging, data visualization and interface simulation. ROS is represented by a service called *roscore* that provides connection information to *nodes* so that they can transmit messages among them, through a common topic to *publish* and *subscribe* data.

2.3. Virtual Robot Experimentation Platform (V-REP)

The robot simulator V-REP is based on a distributed control architecture. This framework can be controlled via a particular embedded script and a ROS node. One of the most important aspects is the flexibility, portability, and scalability of the simulation model. V-REP implements a ROS node via publishers and subscribers, and it's directly enabled from within V-REP, via an embedded script command. In virtual environments, including the most popular mobile robots, such as Khepera III, KUKA YouBot, NAO, and Pioneer 3-DX, but designing the mobile robot model and simulation scene, V-REP supports CAD data formats, is of great relevance for some projects of real robots and environments. The following scene objects are supported in V-REP: joints, shapes, proximity sensors, vision sensors, force sensors, cameras, lights, paths, and dummies. The functionalities in simulation models make easier the programmers task and reduce the deployment complexity for the users [Rohmer et al. 2013].

2.4. Mobile Robot Model

For this work, a mobile robotic virtual platform was modeled with V-REP based on a mobile robot developed by Daniel Deda, OCTO ROBOT, for use as a tool for teaching, as described in [Magrin and Deda 2016]. The real OCTO robot, Figure 1, has eight sonars positioned in an octagonal pattern, that in addition to ensuring 45° displacement between each sonar, can correspond to an orientation from cardinal and ordinal, resulting in a reading distance of sonar that can be oriented by a compass.

Figura 1. Mobile robot platform - OCTO robot by Daniel Deda [Magrin and Deda 2016].

In order to introduce the V-REP functionalities into the OCTO robot model, Figure 2. The OCTO robot was modeled importing CAD file for a body, adding motors for the wheels, selecting proximity sensors, and included objects for representing the access points (AP's) in the scene, for simulated received signal strength (RSS) from a wireless network.

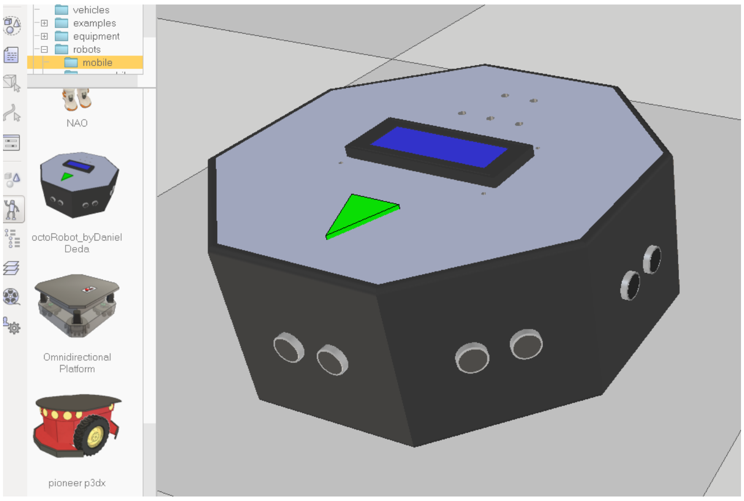

Figura 2. OCTO robot model in virtual platform V-REP.

3. Indoor Localization using Sensor Fusion

Sensor fusion is presented in [Zhao and Luo 2008], [Kam and Kalata 1997], [Luo and Kay 1989], considering that many autonomous mobile robot platforms are assembled with the goal of studying applications of multiple sensors and techniques to combine their information.

In this section, a hierarchical sensor fusion with fingerprint kNN is presented, to solve the problem of location of a mobile robotic platform model equipped with sonars octagon, compass and RSS. A sensor fusion block diagram (Figure 3) shows the localization process while different steps are needed to obtain an estimation of the absolute robot pose in the map. The localization process has three main steps: *perception*; *preprocessing* and *hierarchical fusion* that results localization grid in the map. This paper does not have the goal of covering the robot odometry, or robot tracking, but its absolute localization in a map, represented in grid form.

The hierarchical sensor fusion method, Figure 3, is proposed for simulated indoor localization of an autonomous mobile robot and follows these steps: perception, preprocessing, and fusion levels.

3.1. Perception

The simulation of data acquisition from OCTO robot includes the following sensors: sonars octagon, compass, and RSS from AP's.

The sonars have as main purpose the environment perception in $360°$, sonars position in an octagonal pattern (Figure 2), corresponding to an orientation cardinal and ordinal, avoiding measurement error by *crosstalk* and resulting in a reading distance of sonar oriented by compass.

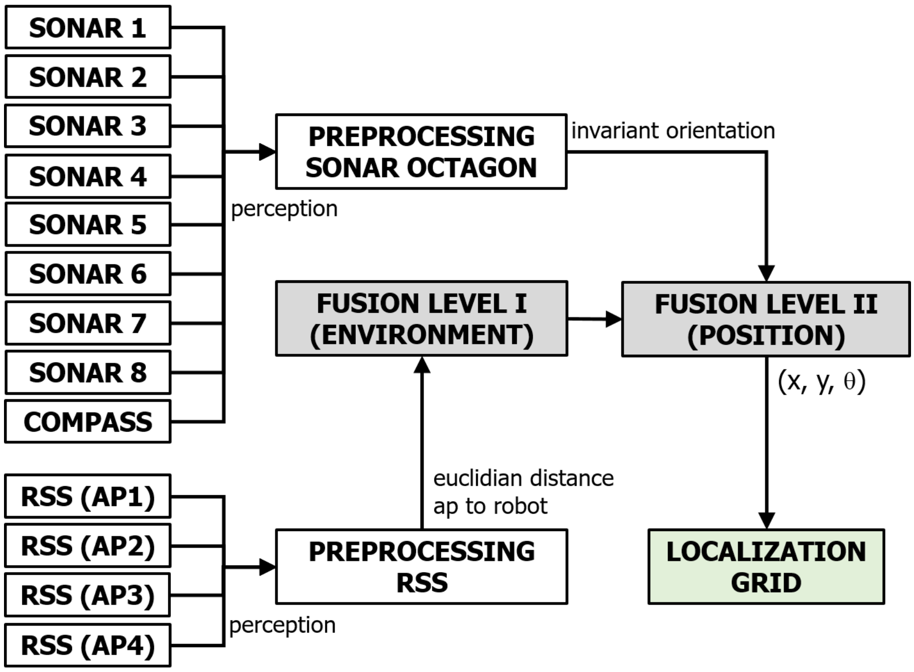

Figura 3. Hierarchical sensor fusion method block diagram.

The sonars simulation used the same features as the low-cost real sensor, ultrasonic ranging module HC-SR04, which provides measurements within a range from 2 cm to 400 cm, and $15°$ measuring angle. In some cases, it is not possible to simulate a real sensor, as well as the magnetoresistive sensor (digital compass HMC5883L), designed to measure the direction and the magnitude of Earth's magnetic fields, and get received signal strength information from access point in wireless network. In the following section, sensor preprocessing step is shown.

3.2. Preprocessing

Orientation invariance is an important aspect of this work, and the compass is necessary in order to obtain rotation invariance on the sonar measurements obtained with the sonar octagon. In V-REP platform the orientation measurement was simulated from the *yaw* orientation taken from the robot pose in the scene.

The RSS information obtained from AP's, was values of signal strength (dBm), for fingerprint features just was need the distance the robot to each AP. For simulated RSS data was get the robot and AP's *pose* in the scene. In the RSS information, was get the robot and AP's *pose* in the scene (Figure 4), and simulated the attenuation from received signal strength, calculated the Euclidian distance from four AP's to the robot.

3.3. Hierarchical sensor fusion

The sensor fusion method proposed in this work is based on fingerprint technique, kNN matching algorithm with the preprocessing data from multiple sonars, compass and four APs in the scene, resulting in the position grid (x, y) from the autonomous mobile robot. The hierarchical sensor fusion process in Figure 5 defines the grid position in matching levels: sensor fusion level I (larger grid - environment) and sensor fusion level II (smaller grid - position).

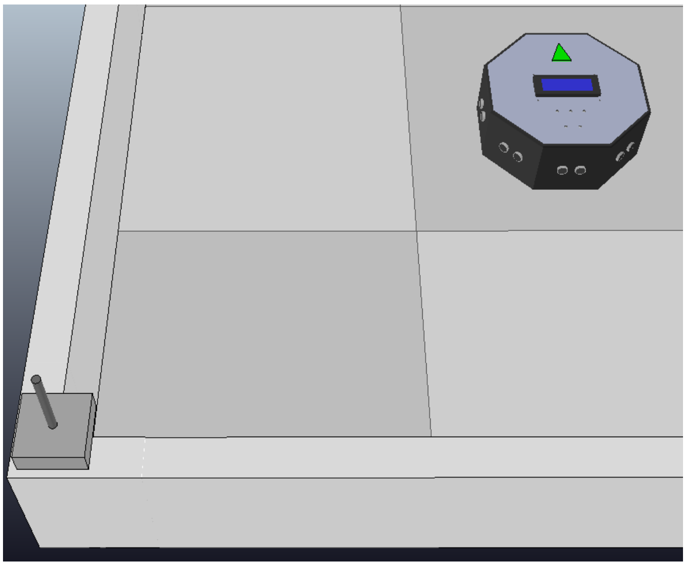

Figura 4. Simulated distance AP to robot - in each corner of the scene.

In the following, the steps from hierarchical sensor fusion are fingerprint maps and positioning levels.

3.3.1. Fingerprint maps

The fingerprinting map construction, or training step, is realized with the perception and preprocessing steps, sorting each measurement with the position on the map. In the fingerprinting map, each information belongs to one position attribute vector, representing a received signal strength from AP's (Euclidian distance, in cm, AP to Robot), and a distance, in cm, obtained from the sonars octagon oriented by the compass (invariant orientation).

3.3.2. Positioning levels

The hierarchical fusion steps (Figure 5) is the result of each position level: sensor fusion level I and sensor fusion level II, belongs to fingerprinting map (training) realized *a priori* step in the matching process. Each positioning step applied a fingerprint technique used kNN matching algorithm. The fusion algorithm for matching between fingerprinting map vector and the features vector uses Euclidian distance (1), applied with metric matching. The first positioning step *larger grid* (sensor fusion level I) is the results of matching between training base (fingerprinting map) and the RSS features (AP1toRobot, AP2toRobot, AP3toRobot, AP4toRobot) from the vector. Dividing the fingerprinting map in a map that represents the position of a *larger grid*, generally an environment, such as a room, a lab or a hall. The second positioning step *smaller grid* (sensor fusion - level II) is the result of matching between the fingerprinting map from *larger grid* (sensor fusion - level I) and sonar features (sonarN, sonarNE, sonarE, sonarSE, sonarS, sonarSW, sonarW, sonarNW) from the vector. The result of the hierarchical fusion step through each positioning level reduces uncertainty about the robot position, in first step results in a *larger grid*

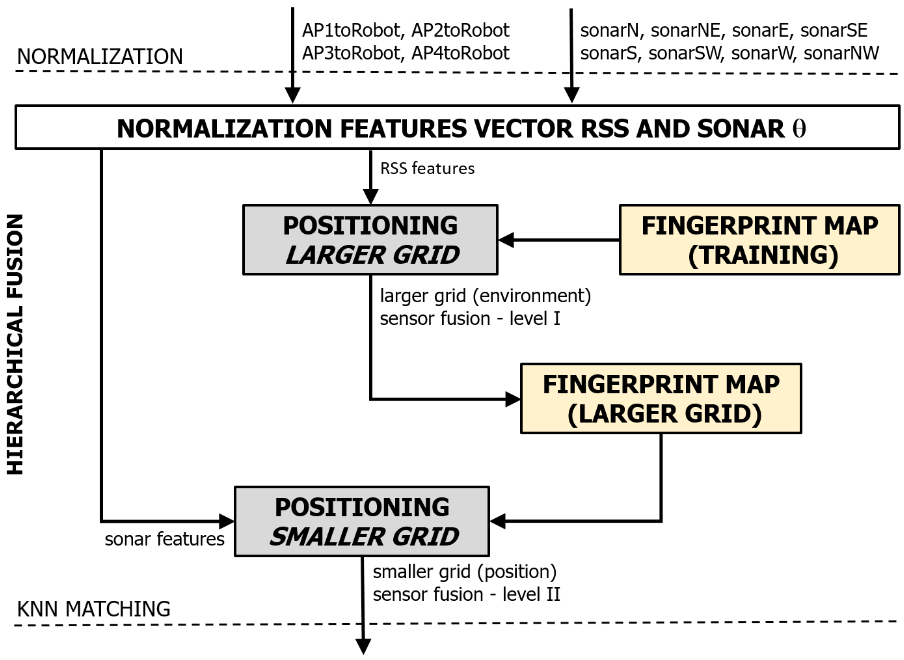

Figura 5. Hierarchical fusion block diagram, based on kNN matching algorithm.

position, and then in a second step the absolute robot position (x, y) in a *smaller grid*.

3.4. Localization grid

The HSF process, as shown in the previous sections, performs the perception of the environment with eight sonars oriented by compass and four APs; a preprocessor step is necessary for normalizing the data acquisition (simulation). The hierarchical fusion uses a two positioning level, segmentation, responsible for environment location (*larger grid*) restricted the fingerprinting map database, built in training step and the position level, where results in absolute location for the autonomous mobile robot.

4. Simulation and Results

The sensor fusion method based on fingerprint kNN started with simulation the environment (building the scene in the map) and the mobile robot tests model, followed by data collection for training database, mobile robot teleoperation and path planning, and navigation for positioning step. However, for kNN validation was generated recognition rate in different grid spaces. Figure 6 shows the structure of ROS *nodes* and topics to *publisher* and *subscriber* data with V-REP.

4.1. Simulation model

The research presented in [Magrin and Todt 2016] analyzed the training database features of fingerprinting map in a real environment and chosen empirically different types of space for training database. For simulated the scene in V-REP was chosen different types of the grids sizes, considering each grid size 50x50 cm, such as: *grid 6x4* and *grid 4x4* - represents a room space, *grid 10x2* - represents a hall space, and *grid 5x4* - two rooms space with the same size in the scene.

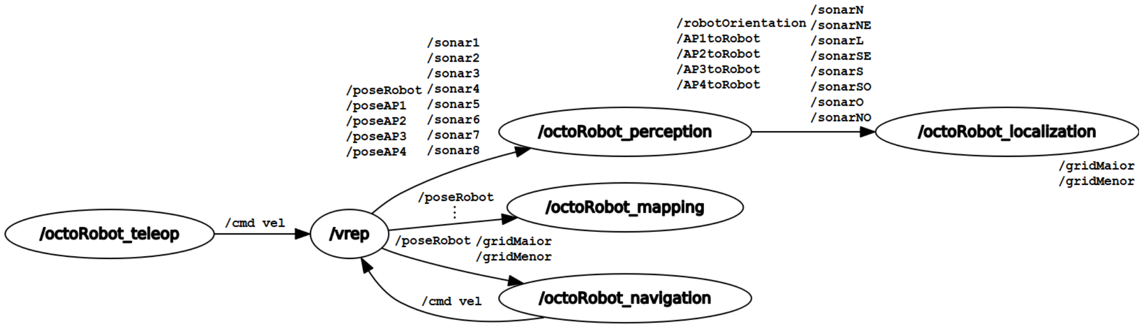

Figura 6. Structure of ROS nodes (plugin *rqt_graph*), topics to publisher and subscriber.

Mobile robot model used the same features of the real sensors, but errors were not included as a parameter in this simulation model. For simulation, the sensors measurements first were chosen the magnetic north pole of the Earth in the scene, for after measuring the sensors data of sonars invariant orientation, reading is always performed in cardinal and ordinal directions, and RSS (simulated by the distance between AP to Robot).

4.2. Training step

Planning a training step is of great relevance for building the fingerprinting map because the data collected will be the database of required training for algorithm matching.

For the data collected in the scene was positioning the OCTO robot in each grid of the map, followed your angular movement in 90°, and the robot always in the center of the grid (Figure 7). The features vector collected in each positioning for fingerprint map, are *sonarN, sonarNE, sonarE, sonarSE, sonarS, sonarSW, sonarW, sonarNW, AP1toRobot, AP2toRobot, AP3toRobot, AP4toRobot, gridLarger, gridSmaller*, the last two vectors represents the position of the robot in the scene *larger grid* (environment in the map) and *smaller grid* (position (x, y) in the map).

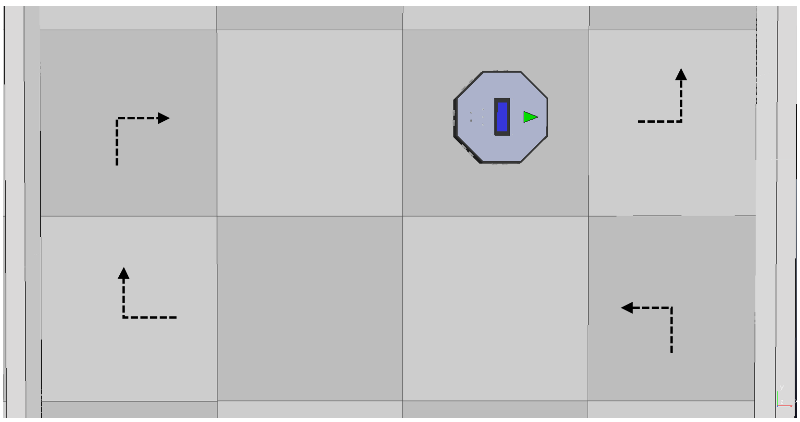

Figura 7. Robot in the center of the grid, and followed your angular movement.

4.3. Validation

The validation methodology of kNN matching algorithm consists in generating the recognition rate, using the database training (fingerprint map). As the hierarchical fusion step uses the matching algorithm into two levels, the validation results will be presented as *larger grid* and *smaller grid*.

For the validation was used mobile robot teleoperation and placed the robot in each positioning, localization the robot in *larger grid* and *smaller grid*. The table 1 shows the recognition rate between fingerprint map and robot positioning through teleoperation, the recognition rate represents 100% in all *larger grid*, and bigger than 90% in *smaller grid*. The localization errors presented are caused by the sonar measurement, fusion level II, when the measuring angle is close to the environment doors.

Tabela 1. Recognition rate. Positioning in different grids.

Grid	**6x4**	**4x4**	**10x2**	**5x4**	**5x4**
Larger	100%	100%	100%	100%	100%
Smaller	92%	100%	95%	95%	90%

4.4. Positioning

In the positioning step, was developed the path planning (navigation) for the virtual OCTO robot with ROS interface. The robot moves in the scene (point A to point B) using proportional control and showing your localization in the map.

5. Discussions and Conclusions

This work has shown through simulation that it's possible to successfully execute the localization in two coordinated levels: the first level used the RSS to find the environment room or specific area, whereas the second level of fusion, used sonars octagon in order to determine the position of the robot on the map (x, y, θ) in a finer resolution.

Finally, the development of this project with the framework ROS allowed segmentation in nodes for each application, as shown in Figure 6, such as teleoperation, perception, mapping, navigation, and localization. This work solves the mobile robot localization problem in indoor environments using the most widely used algorithm for matching, kNN, as mentioned in Section 2, but in a future work other sensor fusion methods using the same mobile robotic virtual platform can be performed, as well other types of sensors can be included to improve the perception of the scene.

Referências

Kam, M. and Kalata, P. (1997). Sensor Fusion for Mobile Robot Navigation. *Proceedings of the IEEE*, 85(1):108–119.

Khaleghi, B., Khamis, A., Karray, F. O., and Razavi, S. N. (2013). Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, 14(1):28–44.

Kunemund, F., Lategahn, J., and Rohrig, C. (2009). WLAN Mobile Robot Localization with Sensor Fusion. In *2009 IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, number September, pages 649–654. IEEE.

Liang, X., Gou, X., and Liu, Y. (2012). Fingerprint-Based Location Positoning using Improved KNN. In *2012 3rd IEEE International Conference on Network Infrastructure and Digital Content*, pages 57–61. IEEE.

Luo, R. and Kay, M. (1989). Multisensor Integration and Fusion in Intelligent Systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(5):901–931.

Magrin, C. E. S. and Deda, D. (2016). Robô Móvel Microcontrolado como uma Ferramenta de Ensino. *Revista de Extensão e Iniciação Científica UNISOCIESC-REIS*, 3:64–71.

Magrin, C. E. S. and Todt, E. (2016). Hierarchical Sensor Fusion Method Based on Fingerprint kNN and Fuzzy Features Weighting for Indoor Localization of a Mobile Robot Platform. In *2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)*, pages 305–310. IEEE.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

Quigley, M., Gerkey, B., and Smart, W. D. (2015). *Programming Robots with ROS*. O'Reilly, first edition.

Rohmer, E., Singh, S. P. N., and Freese, M. (2013). V-REP : A versatile and scalable robot simulation framework. *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on Intelligent Robots and Systems*, (November).

Siegwart, R., Nourbakhsh, I. R., and Scaramuzza, D. (2011). *Introduction to Autonomous Mobile Robots*. MIT Press, second edition.

Torteeka, P. and Chundi, X. (2014). Indoor Positioning Based on Wi-Fi Fingerprint Technique Using Fuzzy K-Nearest Neighbor. *Proceedings of 2014 11th International Bhurban Conference on Applied Sciences & Technology (IBCAST) Islamabad, Pakistan, 14th - 18th January, 2014*, pages 461–465.

Tzafestas, S. G. (2014). *Introduction to Mobile Robot Control*. Elsevier.

Webb, A. and Copsey, K. (2011). *Statistical Pattern Recognition*. Wiley, 3 edition.

Yang, R. and Zhang, H. (2014). RSSI-Based Fingerprint Positioning System for Indoor Wireless Network. In *Intelligent Computing in Smart Grid and Electrical Vehicles. Volume 463 of the series Communications in Computer and Information Science*, pages 313–319.

Zhao, X. and Luo, Q. (2008). Survey on Robot Multi-sensor Information Fusion Technology. *2008 7th World Congress on Intelligent Control and Automation*, pages 5019–5023.