Route Learning and Reproduction in a Tour-Guide Robot

Víctor Alvarez-Santos[1], A. Canedo-Rodriguez[1], Roberto Iglesias[1], Xosé M. Pardo[1], and C.V. Regueiro[2]

[1] Centro Singular de Investigacion en Tecnoloxias da Informacion
Universidade de Santiago de Compostela, Spain
victor.alvarez@usc.es

[2] Department of Electronics and Systems
Universidade da Coruña, Spain

Abstract. Route learning and reproduction in tour-guide robots is usually performed with the help of an expert in robotics. In this paper we describe a novel approach to these tasks, which reduces the intervention of an expert to a minimum. First, the robot is able to learn routes while following a human acting as a route instructor. Then, anyone can easily ask the robot to reproduce a route using various hand gestures. In order to achieve an accurate route learning and reproduction we use a novel localization algorithm, which is able to combine various sources of information to obtain the robot's pose. Moreover, the path planning and obstacle avoidance used to navigate while reproducing routes are also described in this article. Finally, we show through several trajectories how the robot is able to learn and reproduce routes.

Keywords: tour-guide robot, route learning, route reproduction, human following.

1 Introduction

In the coming years, personal service robots are expected to become a common element in most homes or offices, playing an important role as appliances, servants and assistants; they will be our helpers and elder-care companions. These robots will need to be capable of acquiring a sufficient understanding of the environment, being aware of different situations, as well as establishing a successful communication with humans in order to be able to cooperate with them.

We are currently building a general purpose tour-guide robot, which learns routes from an instructor, and then shows these routes to the visitors of the event where the robot operates. The development of the control software for this robot involves many open challenges such as robot localization, human identification, and robot navigation in crowded and challenging environments.

In this work, we describe the route learning and reproduction processes in our robot. The route learning takes place while following an instructor, who only needs to move in the environment showing the route to the robot. The reproduction of

J.M. Ferrández Vicente et al. (Eds.): IWINAC 2013, Part II, LNCS 7931, pp. 112–121, 2013.

a route by the robot is performed on user's demand. During the reproduction, the robot travels to the origin of the route, and starts to mimic the route learnt from the instructor in the past. We want to remark that other service robots like a robotic wheelchair will also benefit from this ability: the wheelchair could learn a route in a hospital while following a nurse, and then be able to travel back to a patient's room.

2 Related Work

In the late nineties, the first well-known tour-guide robots (*Rhino* [1] and *Minerva* [2]) had no online route learning abilities. In fact, points of interest and the rest of information which make up a route were introduced in the robot by an expert. In these robots, the users could select the exhibit that they wanted to visit using a touch-screen located at the robot. This strategy for route recording and route learning, with minor variations, became a common element in the tour-guide robots that were developed afterwards. One of them was *RoboX*[3], which was designed for long time operation in a public exposition where the routes were also precoded in the robot by experts. These routes were shown to visitors as soon as anyone approached the robot. Other tour-guide robots [4] randomly choose a visitor and guide him to an exhibit, which was selected based on information from RFID tags, which are used to detect which exhibit has not been visited yet by the human. Urbano [5] is another tour-guide robot, which requires a manual creation of a navigation graph, and a database of objects and locations. This, allows the robot to dynamically generate routes that will be shown to a visitor depending on which category is the user classified on. Robotinho [6] was one of the first humanoid tour-guide robots, and it could give tours to people who show interest in the robot. These tours were also manually introduced in a pre-operational stage of the robot.

A recent tour-guide robot [7] allows the user to select which exhibit he wants to visit using speech recognition. It can also identify which part of the exhibit needs further explanation by recognizing pointing gestures. Despite of these improvements in route management, this tour-guide robot still requires a pre-operational stage in which an expert introduces location and other information about the exhibits available. Another recent tour-guide robot [8] states that new routes can be created on the fly, although no details about the process are given. Finally, one of the latest tour-guide robots that has been developed [9] explores different ways of reproducing routes: it can keep track of the human which is following him and wait for him while reproducing the route. However, the definition of the routes is still done by an expert, and the same ahppens with the selection of which route should be reproduced.

3 System Overview

The latest version of the robot that we use as prototype for our tour-guide robot can be seen in Fig 1 (left). It consists of a Pioneer P3DX robot base, a laser

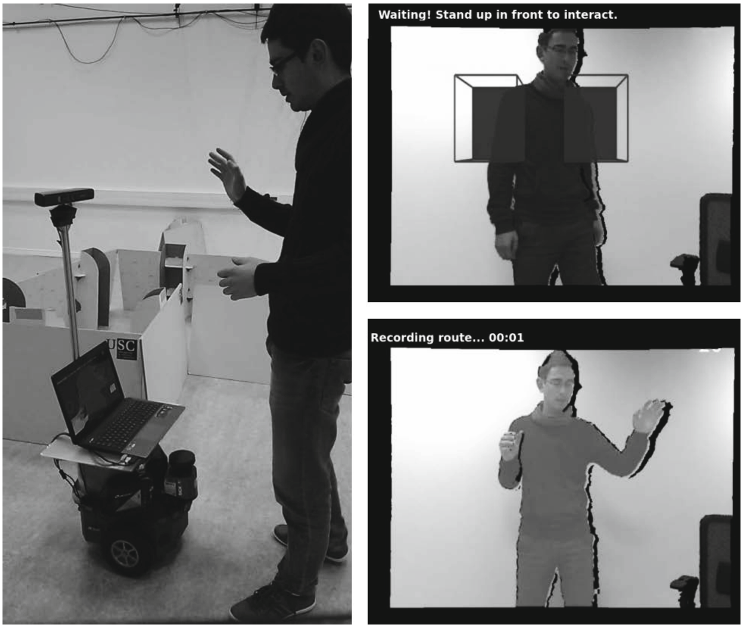

Fig. 1. Left: picture of our prototype of tour-guide robot interacting with a human. Right: captures of the robot's user interface with augmented reality.

range finder, and a range camera which is located at the top of the robot. Its main processing unit is a laptop.

An instructor can teach routes to the robot by commanding the robot to follow him. The instructor can be anyone from the staff of the event where the robot will operate, and this person will not need to be an expert in robotic. The instructor can also inform the robot about the *points of interest* along the route. Thus, every time that the instructor reaches one of those *points of interest* he will be able to record an explanatory *voice messages*.

On the other hand, the visitors of an event can ask the robot to show them those routes. The robot will search and travel to the first point of the route, and start to mimic the full route. When the robot arrives at any *point of interest*, it will reproduce the corresponding *voice message*, which has been previously recorded by the instructor.

We have developed an interface which provides visual feedback to the users so that they know whether they are being properly detected by the robot. On the other hand, this interface also uses augmented reality (virtual buttons) so that either the instructor or the visitor will be able to use hand gestures to start recording a route or voice messages along it (instructor), or select and reproduce a route (visitor).

In order to successfully accomplish the tasks of route recording and route reproduction, we need a robust robot localization that we have developed. As we will see in the next section, the use of this localization strategy makes us consider two different stages: the *deployment stage* and the *operational stage*.

4 The Deployment Stage

In the location system that we use in our robot [12] we combine information from different sources, which have to be previously set up when the robot arrives to a new environment. This process is performed by an expert in a short period of time.

Therefore, the goal of this stage is to generate what we call the *model* of each sensor. By *model* we understand a function that is able to determine the probability of a sensor reading provided a robot position. These models will be used to localize the robot in the environment considering the sensor readings at each instant (section 5.1). For this reason, we need to move the robot around the environment collecting data: odometry, laser signatures and signal strength of each Wi-Fi access point (AP).

Firstly, we use this data to build a laser-based map of the environment. Using this map, we can pre-compute the expected laser signature $\boldsymbol{l_e}$ (with N_L readings) from each possible pose s in this map and compare it with the actual laser signature $\boldsymbol{l_t}$. This allows us to approximate the laser *sensor model* by Eq. 1:

$$p(z^l|s) = \left[\sqrt{1 - \frac{\sqrt{\sum_{i=1}^{N_L} l_t^i \cdot l_e^i(s)}}{N\sqrt{\sum_{i=1}^{N_L} l_e^i(s) \sum_{i=1}^{N_L} l_t^i}}} \right] \left[\frac{1}{N_L} \sum_{i=1}^{N_L} max\left(1 - \frac{|l_e^i(s) - l_t^i|}{max_{LD}}, 0\right) \right] \quad (1)$$

The first term of the previous equation calculates the Hellinger distance which estimates shape similarity. To take scale into account, the second term computes the average difference among each pair of range measurements $(l_t^i, l_e^i(s))$. The parameter max_{LD} (maximum laser difference) indicates the maximum allowed difference among each pair of laser ranges.

Secondly, we use [12] the signal strength of the Wi-Fi AP and the ϵ-Support Vector Regression technique with Radial Basis Function kernels (ϵ-SVR-RBF [16]) to provide an estimate $z^w = (x^w, y^w)$ of the robot position from the signal strength of the audible Wi-Fi APs. The prediction error of the ϵ-SVR-RBF can be approximated by a zero mean Laplace distribution [16], and therefore the *sensor model* of our Wi-Fi location system can be approximated by Eq. 2:

$$p(z^w|s) = \left[\frac{1}{2\sigma_x^w} e^{-\frac{|x^w - x|}{\sigma_x^w}} \right] \left[\frac{1}{2\sigma_y^w} e^{-\frac{|x^y - y|}{\sigma_y^w}} \right] \quad (2)$$

where the x^w and y^w are the coordinates corresponding to the position of the robot that has been estimated using the signal strength of the Wi-Fi APs.

5 The Operational Stage

Using the information recorded during the deployment stage, we will be able to localize the robot at any instant, because of this, during the operational stage any instructor can teach routes to the robot, or any visitor can demand the reproduction of these routes. Both route learning and reproduction involve several critical tasks which need to be properly addressed: robot location in the environment, the detection, identification and tracking of the instructor, the path planning of the routes and a safe navigation of the robot in a crowded environment. In the next sections, we will describe the most relevant asoects of each of these tasks.

5.1 Robot Localization

As we have already introduced in section 4, our localization algorithm [12] is able to combine information from different sources: a laser scanner, a compass, and a Wi-Fi positioning system.

To this extent, we fuse the information from all the data sources by means of a particle filter [17]. The most relevant contribution of this filter lays on the weighting of the particles p^i using the N sensors available. Each particle p_i consists of a estimated position of the robot s_i, and a weight $\hat{\omega}^i$. The weight of each particle is re-computed taking into account the meassurements of the available sensors:

$$\hat{\omega}^i = \prod_{k=1}^{N} p(z^k|s^i) \quad \forall\, i \in \boldsymbol{P} \tag{3}$$

It is important to notice that in order to compute the weight of the particles, we need to use the models $p(z|s)$ of each sensor obtained in the deployment stage (Eq. 1 and Eq. 2).

This solution [12] is a indoor localization robust to sensor failures or sensor with different data acquisition rates, because it allows us to keep updating the particle filter even if not all the sensors are available.

5.2 Route Learning

Route learning in our tour-guide robot is performed under the command of an instructor who moves along the desired route while being followed by the robot. That instructor does not have to be an expert in robotics, thus, anyone can teach new routes to our robot by simply letting the robot follow him.

Therefore, the robot will follow the instructor, logging its pose every metre (*way-points* from now on). Every time that the robot logs a new *way-point*, it will make a sound to let the user know that the route is being recorded. In addition, during the route teaching, the instructor can stop at any *point of interest* and record a voice message.

It is obvious that the performance of the person following behaviour is critical for this task, that is why in the past [11] we have presented a study of several colour and texture features, which are used to distinguish the instructor from the rest of the people which moving around the robot. In order to select which ones would perform best, we evaluated 27 visual characteristics, and selected eight of them based on which features shared the lowest mutual information. The features that we have selected are: the HSV colour space, the second derivatives in both image axis, the Canny edge image, the Centre-symmetric LBP, and the MPEG-7 edge histogram. In addition, we use an online feature weighting procedure [11], which is able to increase the weight of those features which are more discriminant at each moment. The weight of each feature f is dynamically updated based on the feature's discrimination power calculated with the Hellinger distance d_f:

$$d_f = \sqrt{1 - \sum_{i=1}^{B} \sqrt{h_1(i) h_2(i)}} \tag{4}$$

where h_1 is the normalized distribution of the feature in the instructor, and h_2 the normalized distribution of the same feature in the rest of the people present in the scene where the robot operates, and B the number of bins in which we have discretised the distributions. This allows us to obtain a visual distance between humans detected in the image and a visual model of the instructor, this visual distance combined with the physical distance (predicted position vs actual position) can be used to obtain a probability of a human to be the instructor that the robot is actually following.

Moreover, we have developed our own human detector [10] which is able to perform pixel-level segmentation of humans from the background. This point is quite important because it has significantly improved the feature extraction, and thus, the accuracy of the visual distance between humans and the instructor.

We have evaluated the person following ability in many real world scenarios with people of all ages. One of those scenarios was the Domus museum (A Coruña, Spain), which presents many challenges: strong illumination changes, an uneven floor and a crowded environment.

The advantage of this route learning procedure is that it is not performed on the deployment stage, which would increase the robot's deployment time, and also that anyone can be followed regardless of his knowledge about the robot.

5.3 Route Reproduction

Route reproduction is a critical behaviour of the robot.

When the robot is about to reproduce a route, it picks the first point from the route and tries to reach it. For this task, we use a global planner based on Dijkstra's algorithm [13] in order to find the shortest path between the current robot's pose and the first point of the route. Once the robot has calculated its global path towards the goal, the local planner is responsible for safely moving the robot towards the environment. Thus, local navigation is performed using the

Dynamic Window Approach (DWA) [14], which consists in forward-simulating various trajectories which were generated by sampling the linear and angular velocities of the robot. Then, each simulated trajectory is scored based on some parameters, such as distance to the goal and proximity to obstacles. The trajectory with the highest score is chosen, and the velocities which generated that trajectory are executed in the robot's base. This algorithm, allows us to continuously evaluate the trajectory, and thus avoid obstacles such as persons walking in the robot's path.

Once the robot has reached the first point in the route, the robot will pick the next *way-point* and navigate towards it. This is done until there last point is reached or the visitor stops the robot. Moreover, the *points of interest* are treated in the same way, with the difference that the robot stops in that point for a while, until the voice message is played back.

In order to speed up route reproduction, we are flexible when it comes to decide whether the robot has reached a *way-point* or not. Therefore, we have set a distance of 0.3m as acceptable, as well as a difference in the yaw goal of 0.4 radians. Moreover, whenever there is a point within the route that is unreachable the robot can skip it and travel to the next one. The robot will be making bell sounds every time it reaches a point and a error sound if it skips a point.

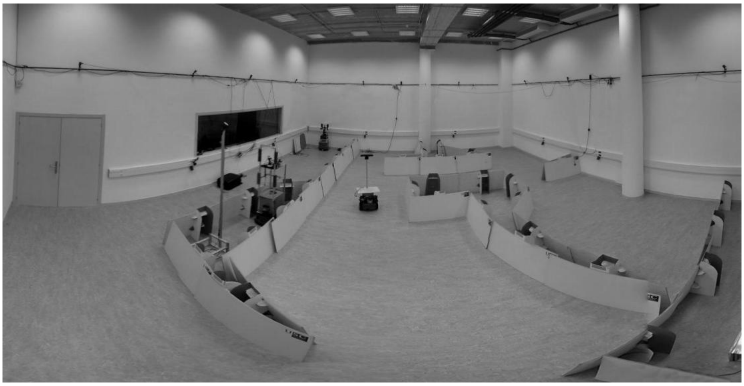

Fig. 2. $100m^2$ robotics laboratory where the experiments were conducted. We prepared an environment with 5 rooms, and recorded and reproduced several routes.

In order to test the route learning and reproduction of our robot, we have recorded two routes in our robotics laboratory (Fig. 2), where we have settled an indoor environment with five rooms. We have chosen this laboratory as the first test-bed for route-management in our robot because it is similar in size ($100m^2$) and space arrangement (five rooms) with most home or offices nowadays. The routes that we used for testing are illustrated in Fig. 3 where we can see the robot's trajectory during route recording (drawn in a dark-grey and dotted line), and the trajectory (drawn in a light-grey and continuous line) that the robot

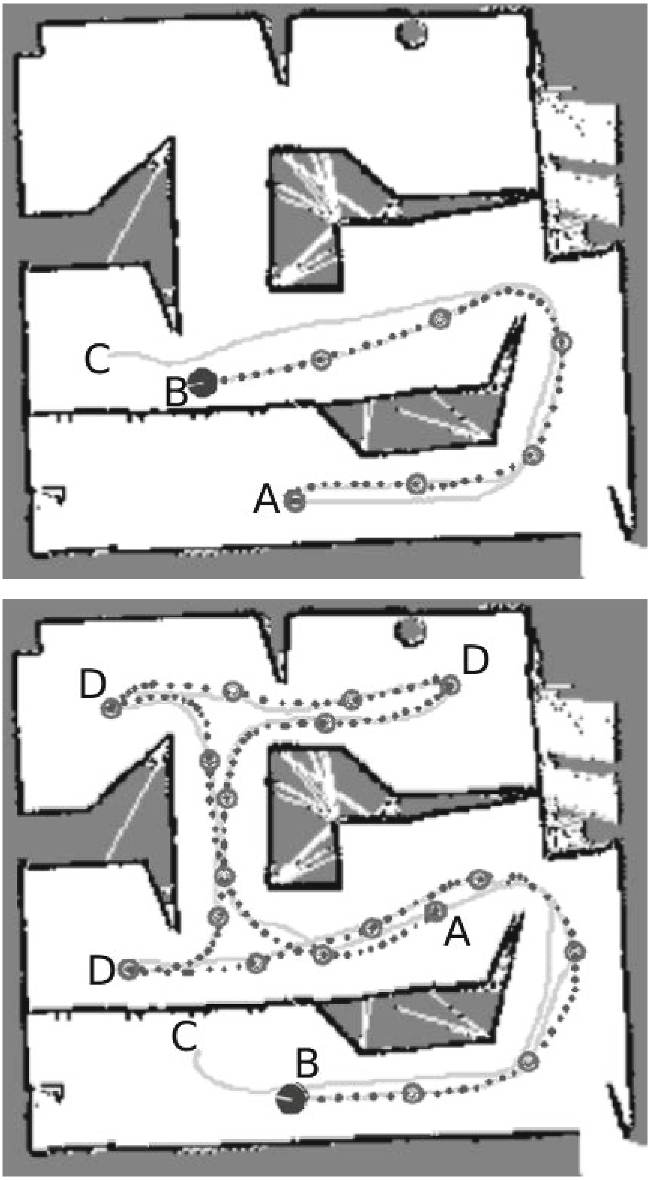

Fig. 3. Trajectories for recording and reproduction of two routes. The dotted trajectory represents the robot's trajectory while recording the route, and the circles are the recorded way-points. The light-grey lines are the robot's trajectories while reproducing these routes. The points marked with the letter A are the first points recorded in the routes, the points marked with the letter B are the end point of the routes, the points marked with the letter C are the location of the robot when it was asked to reproduce the route, and the points marked with the letter D are the locations of *points of interest* specified by the instructor while recording the route.

performed when it reproduced the route. In these examples we can see how the robot has to find a valid path and to travel from its initial pose (C in Fig. 3) to the initial point of the route (A in Fig. 3). This is not a trivial issue since the distance between both points can be several dozens of metres, and the path between those points might cross several rooms and corridors. Once the robot has travelled to the first point within the route, it moves from one *way-point* to the next one in the route (illustrated with dark-grey circles in Fig. 3). When the robot reaches a point of interest, it plays the voice message recorded by the instructor. Finally, when the robot finishes the reproduction of the route, it plays a bell sound and waited for new commands from its nearby users.

The first route (Fig. 3, top) was recorded in 56 seconds, and it is 15.3m long. In this route no points of interest were recorded by the instructor, but it is interesting in order to notice how fast can anyone teach a route path to the robot in a home-like environment with narrow corridors. In this experiment, the robot had to travel 29.2m in order to find the first point of the route and mimic the complete route path.

The second route (Fig. 3, bottom) was recorded in 155 seconds, and it is 40.8m long. When reproducing this route the robot travelled 52.8metres. In this route, the instructor recorded three voice messages at the points of interest. These messages were very short: around 5 seconds each one. In this way we can isolate the time used for route management (path teaching and gesture interaction) from that of the message itself: with our tour-guide robot a complex route can be recorded in few time by non experts.

6 Conclusions and Future Work

In this work we have described the route learning and reproduction behaviours in our tour-guide robot. Moreover, we have presented a critical element for those behaviours: a multi-sensor fusion algorithm for robot localization. Moreover, we have successfully tested the route management through two routes that were successfully recorded and played back in a real world environment.

Like we did in the past [11] [10], we plan to test the route management and robot localization in challenging scenarios such as the Domus museum (A Coruña, Spain), with which we have been collaborating during the development of this tour-guide robot. In those tests, we will adhere to the ISO/IEC-9126 for evaluating the quality of the route management with real users. In addition, we also plan to improve the human-robot interaction that takes place while reproducing a route by including a pan camera which can rotate and monitor the user to whom the robot is showing the route.

Acknowledgements. This work was funded by the research projects TIN2009-07737, TIN2012-32262, the grant BES-2010-040813 FPI-MICINN, and by the Galician Government (Consolidation of Competitive Research Groups, Xunta de Galicia ref. 2010/6).

References

1. Buhmann, J., Burgard, W., Cremers, A.B., Fox, D., Hofmann, T., Schneider, F.E., Strikos, J., Thrun, S.: The mobile robot Rhino. AI Magazine 16(2), 31 (1995)
2. Thrun, S., Bennewitz, M., Burgard, W., Cremers, A.B., Dellaert, F., Fox, D., Hahnel, D., Rosenberg, C., Roy, N., Schulte, J., et al.: MINERVA: A second-generation museum tour-guide robot. In: Proceedings of IEEE International Conference on Robotics and Automation, vol. 3 (1999)
3. Jensen, B., Froidevaux, G., Greppin, X., Lorotte, A., Mayor, L., Meisser, M., et al.: The interactive autonomous mobile system RoboX. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, pp. 1221–1227 (2002)
4. Shiomi, M., Kanda, T., Ishiguro, H., Hagita, N.: Interactive humanoid robots for a science museum. In: Proceedings of the 1st ACM SIGCHI/SIGART Conference on Human-Robot Interaction, pp. 305–312 (2006)
5. Rodriguez-Losada, D., Matia, F., Galan, R., Hernando, M., Montero, J.M., Lucas, J.M.: Urbano, an interactive mobile tour-guide robot. In: Seok, H. (ed.) Advances in Service Robotics, pp. 229–252. In-Teh (2008)
6. Faber, F., Bennewitz, M., Eppner, C., Gorog, A., et al.: The humanoid museum tour guide Robotinho. In: The 18th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2009, pp. 891–896 (2009)
7. Avilés, H., Alvarado-González, M., Venegas, E., Rascón, C., Meza, I.V., Pineda, L.: Development of a Tour–Guide Robot Using Dialogue Models and a Cognitive Architecture. In: Kuri-Morales, A., Simari, G.R. (eds.) IBERAMIA 2010. LNCS, vol. 6433, pp. 512–521. Springer, Heidelberg (2010)
8. Yelamarthi, K., Sherbrook, S., Beckwith, J., Williams, M., Lefief, R.: An RFID Based Autonomous Indoor Tour Guide Robot. In: IEEE 55th International Midwest Symposium on Circuits and Systems, MWSCAS, August 5-8, pp. 562–565 (2012)
9. Bueno, D.R., Viruete, E., Montano, L.: An autonomous tour guide robot in a next generation smart museum. In: 5th International Symposium on Ubiquitous Computing and Ambient Intelligence (2011)
10. Alvarez-Santos, V., Iglesias, R., Pardo, X.M., Regueiro, C.V., Canedo-Rodriguez, A.: Gesture-based interaction with voice feedback for a tour-guide robot. Submitted to Journal of Visual Communication and Image Representations (2013)
11. Alvarez-Santos, V., Pardo, X.M., Iglesias, R., Canedo-Rodriguez, A., Regueiro, C.V.: Feature analysis for human recognition and discrimination: Application to a person following behaviour in a mobile robot. Robotics and Autonomous Systems 60(8), 1021–1036 (2012)
12. Canedo-Rodriguez, A., Alvarez-Santos, V., Santos-Saavedra, D., Gamallo, C., Fernandez-Delgado, M., Iglesias, R., Regueiro, C.V.: Robust multi-sensor system for mobile robot localization. In: Ferrández, J.M., Álvarez, J.R., de la Paz, F., Javier Toledo, F. (eds.) IWINAC 2013, Part II. LNCS, vol. 7931, pp. 92–101. Springer, Heidelberg (2013)
13. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numerische Mathematik 1(1), 269–271 (1959)
14. Fox, D., Burgard, W., Thrun, S.: The dynamic window approach to collision avoidance. Robotics & Automation Magazine 4(1), 23–33 (1997)
15. Grisetti, G., Stachniss, C., Burgard, W.: Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters. IEEE Transactions on Robotics 23(1), 34–46 (2007)
16. Chang, C.-C., Lin, C.-J.: LIBSVM: A library for support vector machines. ACM Trans. Intell. Syst. Technol. 2(3) (2011)
17. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics. MIT Press (2003)