Segway Robot Loomo As A Tour Guide

Johanna Liebl
Chair for Data Processing, Technical University of Munich
johanna.liebl@tum.de

***Abstract*—In this paper we use the off-the-shelf mobile robot Loomo to implement a robot tour guide for an art gallery. The robot's task is to locate the exhibits, in which the visitors are interested in, and to then provide an interactive tour. Therefore, we introduce an exploration algorithm to search for the images along the walls. Furthermore, we use Oriented FAST and Rotated BRIEF (ORB) feature detection, feature mapping and homography to identify the images. Finally, we use the Robot Operating System (ROS) navigation stack to mark the positions of the images and to provide the navigation for a tour along the found exhibits.**

***Keywords*—Service robot, tour guide robot, exploration, feature detection**

I. Introduction

While robots have already a widespread use in industry settings, they yet have to enter the everyday lives of most people. Thus interacting with a robot is for most a novelty, making it an interesting and exciting experience. This can be used to add a new level of entertainment to information providing systems by designing them as interactive robots. Guide robots have already been successfully deployed in museums [1]–[5], shopping malls [6], shops [7] and even in train stations [8]. There they have been used to explain exhibits, to give directions or to entertain visitors.

Yet the vast majority of these guide robots are highly customized, unique robots and only in use by a single research team. This limits the use cases of these guide robots to a few research projects and prevents especially smaller museums from using guide robots at all. Therefore, to allow a more widespread use of guide robots, an easily accessible, affordable and user-friendly guide robot is needed. Ideally, this robot would be given images of the current exhibits in a museum, locate these exhibits in the showrooms by itself and then provide the visitors with information about these exhibits, e.g. by giving an interactive tour. To provide a first step in this direction, this paper introduces a prototype guide robot which uses the off-the-shelf Segway robot platform Loomo [9]. This prototype is designed to locate the exhibits in an art gallery and to provide a short tour for visitors. This work is based on previous work on the robot platform Loomo by the company iteratec [10], where this research internship was done.

The remainder of this paper is organized as follows: In Sec. II a short overview of the state of the art in guide robots is given. In Sec. III the problem statement which is addressed in this paper is formulated. This is followed in Sec. IV by the methodology used in this paper to allow our prototype to search for images, to detect images and to provide a tour. Sec. V contains the results of the experiments conducted. In Sec. VI follows the discussion of these experiments. Finally, a short conclusion is given in Sec. VII.

II. State of the art

A. Related Work

Deploying social robots as tour guides has been an active research topic for quite some time. An overview of what has already been achieved can be found in [4]. Most of the early work, as in [1], [2], focused on the localization of the robot, path planning, as well as object avoidance. This addressed the challenge of locating the robot at the correct position of the map of the building, generally referred to as Simultaneous Localization And Mapping (SLAM) as described in [11], and finding a path between the different exhibits while ensuring not to collide with the visitors or other objects. As the work on SLAM and path-planning progressed, the focus of the research on guide robots shifted to the social components of the interaction between the guide robot and the visitors. Recent works include the correct detection of visitors [6] as well as natural and human-like interaction with the robot, as in [3], [5].

Most of the above mentioned work assumes that the positions of the exhibits are previously known to the robot. As not every museum can be assumed to have the coordinates of each of exhibits readily available, it might be necessary to first find the exhibits. A work that addresses the challenge of first finding the exhibits is [12]. There QR codes are used such that the guide robot can detect the exhibits in the room.

Most of the social robots used in the works mentioned above are highly specialized robots that have been designed uniquely for these projects. Therefore, the challenge remains to combine their abilities in an affordable, easy to use robot, which museums can use on a daily basis.

B. Previous work on project

This paper expands the work of the company iteratec, who used the robotic platform Segway Loomo for previous projects. Loomo is a two wheeled, self-balancing robot that is equipped with multiple cameras as well as infra-red and ultra-sonic sensors, as shown in Fig. 1. The operating system is Andriod based.

The previous work on the project includes the integration of ROS, an open source robotic operation system, which allows the use of existing packages to solve problems like path-planning. An overview of ROS can be found here [13]. Especially the navigation stack provided by ROS, which provides SLAM and path-planning, has been made available.

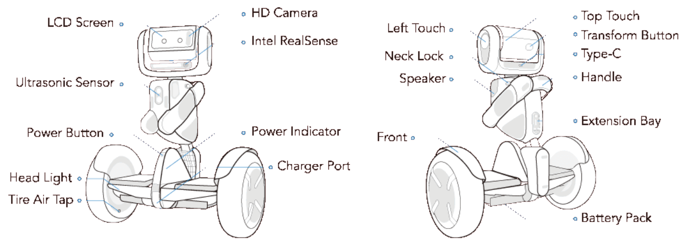


Fig. 1: The overview of the Segway Loomo robot [9]

Additionally a RPLIDAR laser scanner was added to the front of Loomo and connected to ROS with a Raspberry Pi. This is the only customization added to the platform and was needed in order for Loomo to be able to use the navigation stack. Furthermore, a state machine was designed which allows the robot to follow different distinct behaviours. Lastly relevant to the work of this paper is the integration of the Android text-to-speech engine, which allows Loomo to synthesize speech from text. The project has been made publicly available and can be found here [14].

III. Problem statement

To come closer to the goal of an affordable, user-friendly guide robot, this paper aims to use the readily available Segway Loomo robot with the previously discussed work as a base. It is then expanded to allow the robot to become a guide in an art gallery, as this is the setting addressed in this paper. It is assumed that the map of the gallery is known, e.g. because SLAM was used to record it before hand. Additionally the setting assumes a closed, level, indoor area. Furthermore, it is assumed that the exhibits only consist of 2D pictures of which digital copies are available for the robot's usage. Moreover, it is assumed that the pictures are exhibited on the outer walls of the room. Additionally, it is assumed that the human interaction partners follow the robot's suggestions without the need for the robot to check for their compliance. This leaves three main challenges this paper aims to solve: First, as art galleries change the exhibits and their positions regularly, the robot is supposed to autonomously search for the relevant exhibits in the showrooms. In the next step, the robot needs to correctly detect the exhibits and mark their positions. Finally, it is then supposed to give a tour through the art gallery. This leads to the following research questions:

- How can the robot explore the showrooms while ensuring to search all the outer walls?
- How can the robot detect the relevant exhibits?
- How can the position of the images be remembered?
- How can Loomo provide an entertaining tour?

IV. Methodology

This work is a conceptual work, where the aim is to provide an prototype such that the above mentioned gallery setting can be solved. Thus the following section introduces the methodology used to achieve this goal.

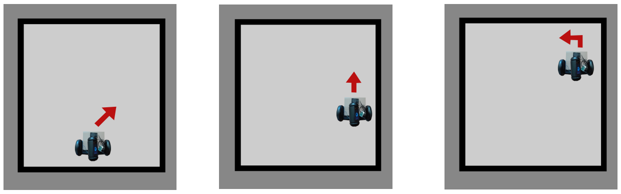
A. Exploration

The first problem to address is the exploration, i.e. the way in which the robot searches for the exhibits. As mentioned above, the images are hung on the outer walls of the room and the map of the room is known in beforehand. Thus an exploration algorithm needs to be found that searches along all the outer walls. As a first attempt we tested the ROS frontier exploration package [15]. Frontier exploration marks the areas which the robot has already explored and then continues to search in the next available free space on the border to the marked area. This package also allows to limit the area to explore to smaller subareas of the whole map. The challenge there is to define the search areas along the walls without choosing all the border points by hand. To overcome this difficulty we tried to determine the corners of the rooms by using corner detection on the provided maps. These points could then be used to define the relevant search areas. But it still proved impossible to ensure that the robot searches along the outer walls, as it is not possible to force the path planning to keep a consistent distance and orientation with regards to the wall. This is a necessary constraint to be able to detect the images with the means provided by Loomo.

Thus the final exploration algorithm we used is a simple wall follower algorithm based on [16]. It consists of three different movement choices, as depicted in Fig. 2, and uses the RPLIDAR laser scanner to detect the walls. The laser scan is split into three areas: the area to the right of Loomo, the area in front and the area to the left. If Loomo detects no walls to its right, it will turn to the right, as shown in Fig. 2a. If it detects a wall to its right side which is closer then a manually set threshold, it will move forward along the wall, as shown in Fig. 2b. Lastly, if Loomo detects a wall to its right and to its front, it turns to the left, as shown in Fig. 2c. To additionally ensure that Loomo is not too close to the wall to detect the images, it moves away from the wall if it is too close. This way Loomo is able to follow the right outer wall of a room at a consistent distance and orientation. The disadvantage of this approach is that it does not use the map provided beforehand, in which the path planner usually allows to mark areas as off-limits and thus ensures that it is possible to keep the robot from entering sensitive areas. Another disadvantage is that this exploration algorithm fails if there are obstacles in the way between Loomo's starting point and the outer wall. Nevertheless it proved to work the most reliable among those tested.

B. Image Detection

To introduce as few additional components as possible, we decided to use Loomo's inbuild cameras to detect the images in the showrooms. Loomo has two cameras, one is a InterRealSense Depth camera at the top of its neck, and the other is a HD camera with a wide-angle lens, as can be seen in Fig. 1. As only the HD camera can be turned along with Loomo's head display to face upwards in order to see images hung above Loomo, we used this camera in this work. As there is currently no support for the HD camera in the SDK provided by Segway Robotics, we wrote our own access for it.



(a) No walls (b) Wall to the right (c) Wall to the right and the front

Fig. 2: The three different cases Loomo can face when exploring along the outer walls of the showroom.

Even though most Deep Learning approaches achieve better results detecting images than Feature Detection approaches, as shown in [17], they generally require both large numbers of training samples as well as longer periods of training time. This cannot be combined with the assumption of the art gallery setting, in which the exhibits can easily and often be replaced. Therefore, we used the ORB Feature Detection and Feature Matching of OpenCV [18] to detected the exhibits in Loomo's camera stream instead.

ORB was first introduced in [19] as an alternative to the popular feature detector algorithms Speeded Up Robust Features (SURF) and Scale-Invariant Feature Transform (SIFT). It detects corners in images by using the Features From Accelerated Segment Test (FAST) keypoint detector. The FAST corner detector algorithm uses a circle of 16 pixels around the center pixel to find corners. If a certain number of these pixels are above an intensity threshold, a corner is detected for the center pixel. This algorithm is then expanded in ORB to allow searching for keypoints in different scales and rotations. It then computes an oriented version of the Binary Robust Independent Elementary Features (BRIEF) feature vectors for all keypoints. This allows ORB to perform feature detection with a similar accuracy to SIFT or SURF, but at an order of magnitude faster.

Thus the search for images is performed in this work as follows: First we perform ORB feature detection on digital copies of all the exhibits which Loomo is supposed to find in the art gallery, and save the information about their keypoints and feature vectors. Then for each frame from Loomo's camera stream ORB feature detection is performed as well. In the next step we use a brute force matcher to compare each of the feature vectors from the search images to each of the feature vectors from the camera frame. The best matches for each feature vector are then saved. To ensure the matches are good matches, we use the Lowe's ratio test. This test compares how well the best match is in comparison to second best match. If a keypoint from the search images can uniquely and clearly be matched to the corresponding keypoint in the camera frame, this match should be much better than any other match using the same seach image keypoint. Thus by ensuring that only matches are taken in consideration where the first match of a keypoint is better than the second by a fixed ratio, it is possible to discard weak matches.

We then use the good matches to find a homography between the search images and the frame from the camera stream. This homography describes how the one image can be fit into the other by using rotation and translation. If a homography can be found that fits a minimum number of good matches, the image is detected. To prevent the images from being occasionally wrongly detected, they have to be detected a minimum number of times in a time window to be considered found. An example of how the algorithm recognizes an image in the furnished room can be seen in Fig. 3. In Fig. 3a the digital copy of the exhibit is shown which Loomo is supposed to find. In Fig. 3b a frame from Loomo's camera stream is shown in which the exhibit was detected. The green frame around the detected exhibit shows how the digital copy is rotated and translated by the homography in the camera frame.

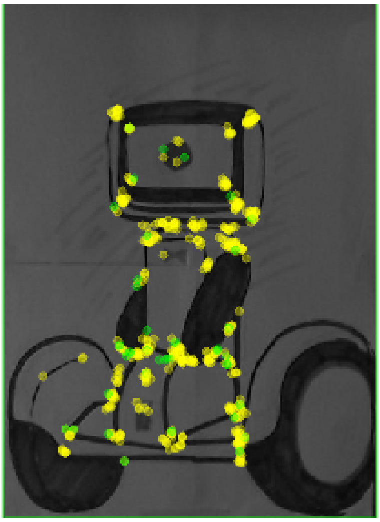
C. Giving The Tour

The tour consists of two main parts: The first is the navigation from one exhibit to another, the second is the interaction with the visitor. To be able to perform the first part, Loomo has to know the location of the exhibits on the map which it has been given beforehand. Thus when Loomo finds an image during the exploration phase, Loomo needs to remember its location. To do so we used the ROS navigation stack. Part of the stack is a localization algorithm, which is based on the adaptive Monte Carlo localization approach [20] to track the pose of a robot against a known map by using particle filters. This algorithm publishes the estimated position of the robot continuously, thus the current position of the robot is known when an image is detected and can be marked. Once all images have been found, Loomo thus has a list of all the images it needs to find along with the position and orientation it had when it found them. These can then be used with another part of the ROS navigation stack, the move base node.

This part of the stack we use for path planning. It allows to set a goal on the map, which it then tries to find a path to. As this node uses global path planning to find a path from the robots current position to the goal marked on the map, as well as local path planning to avoid moving obstacles, it is sufficient for Loomo's navigation during the tour.

The interaction with the visitor is speech based. We use the Android text to speech engine to let Loomo speak with the visitors. The communication consists of speeches with two different purposes. The first is to guide the human visitors through the gallery. It consists of requests to inform Loomo when to start the tour, to follow Loomo to the next exhibit, as well as to inform Loomo when to continue the tour after arriving at a exhibit. The only input the visitor can give at this point is her consent, expressed through touching Loomo's head display. As an alternative we tried to use a speech recognition system to allow the visitor to answer Loomo by using natural language, but it proved to be too unreliable.

The second part of Loomo's interaction with the visitors has the purpose of offering entertainment. This is done by Loomo offering its opinion on the exhibits on arrival. This is generally a positive sentiment combined with praise for the artist, such as



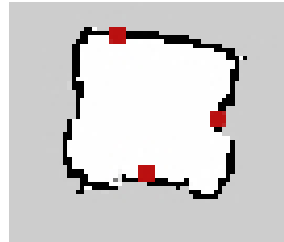
(a) The image of an exhibit.

(b) The found exhibit on the wall

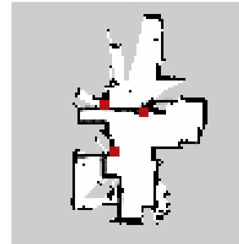
Fig. 3: The left image shows the image which Loomo is given to find the real exhibit. The right image shows a frame from Loomo's camera stream, in which the found exhibit is outlined in green. The white and yellow dots on both images mark the keypoints found by the ORB detector.

for the composition of the image or the colour selection. The phrase Loomo uses for a specific exhibit is chosen randomly from an assortment of predefined sentences. This is done to give the visitor the feeling of companionship when admiring the artworks as well as to amuse the visitor by presenting the robot guide as an art connoisseur.

This could be extended in a real-world-setting to offer more information about the exhibits, e.g. by playing the audio from an audio guide.



(a) Room without obstacles



(b) Room with obstacles

Fig. 4: The maps of the two different rooms in which the system was tested. The red dots mark some of the tested positions of the exhibits.

V. Experiments

We tested the system in two different environments, shown in Fig. 4. The purpose of the tests was to obtain a first estimate of the overall performance of the system. The first room in which the system was tested in, is a mostly empty, square room, as shown in Fig. 4a. The second room shown in Fig. 4b is a furnished, highly decorated room with the north most wall holding a mirror. In both rooms the same three exhibits were hung. They are images of three different levels of difficulty, in the following from the easiest to detect to the hardest: The first is an A2 picture without glossy finish and with high contrasts. The second is an A3 picture with glossy finish and high contrasts. The last is an A3 image with glossy finish and low contrasts. To test the system's capabilities, they were hung at different positions and heights in the two different rooms.

In the first test the exploration algorithm was assessed. Therefore, we started the system in different positions in both rooms. In the room without obstacles, the exploration succeeded constantly. In the furnished room the exploration faced two struggles. The first was a table which the laser scan could not detect as the laser scanner only scans a little above the ground. Thus it detected the legs of the table, but not the protruding tabletop. The second challenge was the mirror on the wall, as the laser scanner occasionally returned faulty results while scanning it.

In the second series of tests we applied the exploration algorithm to test the system's ability to find the images on the walls. There during the first test the images were hung below eye level in the middle of walls. Here Loomo found all three images reliable in both rooms, only occasionally needing an additional round in the furnished room. In the second test the images were hung at eye level. Here Loomo struggled occasionally to find the A3 image with low contrast, especially in the furnished room. In the next test the images were hung again at eye level, but also close to corners. In the room without obstacles Loomo found all three images reliably, in the furnished room only the A2 image with high contrast was found consistently when hung close to corners. In the third test we changed the source of the search images. Before, the images provided for Loomo where frontal shots of the images hanging at Loomo's height, which we took from the HD camera's stream. In this test the images were instead taken with a cellphone camera. Here the results were generally worse. The images were still found, but especially the two A3 images were found far less reliably.

In the last test, we assessed the tour. Here Loomo returned after finding all images to its starting point, asking for the user's permission to start the tour. Once started, Loomo performed the tour as described in Sec. IV-C. In the room without obstacles, this was done without difficulties. Occasionally the path planner took a little longer to find a suited path, but Loomo always reached its goals. In the furnished room, Loomo struggled more to reach all positions. Especially when moving to or from corners, the path planner sometimes failed to find a path.

VI. Discussion

The experiments show that it is possible to use the robotic platform Loomo as a tour guide in an art gallery. Nevertheless, they also show shortcomings in the current design. First, regarding the exploration algorithm, the greatest drawback is that the laser scanner can only be mounted on Loomo's bottom part, thus it only scans the area above the ground for obstacles. In cases where there are things protruding into the open space, like a tabletop, Loomo fails to detect them and can crash into them. A similar issue occurs when the laser scanner scans highly reflective surfaces like mirrors or transparent ones like glass. These surfaces can lead to faulty measurements and thus to undesired behaviours. As this forces the showrooms to be adjusted accordingly, e.g. by placing objects in Loomo's path to stop it from reaching certain areas, this is a great drawback. Future work could hence focus on checking the provided map for fictional walls while driving, such that these could be used to stop Loomo from entering an area.

The results from the experiments regarding the image detection show that images that are around the size of A2 can be found reliably, while the result of the detection of images below that size depend on the specific picture. The higher the contrasts in an image are, the easier it is for Loomo to find meaningful keypoints in it. These then improve the recognition. Another factor is the surface of the images. The glossy images were more reflecting, which lead to artificial highlights on top of the image. These made it more difficult to detect the correct keypoints. Thus this limits which images can currently be used together with the prototype. Finally, regarding the obtainment of the search images, the worse performance of cellphone photos is so far hard to quantify. Further trials with a wider range of pictures would be necessary to test how the results could be improved. Additionally, a solution to easily add new search images needs to be found.
The results from the tour show that Loomo is able to maneuver successfully, as long as there is enough space. As most art galleries are designed to allow for larger crowds of people to move around, this should be sufficient for Loomo as well. The only challenge might be moving through the crowd, especially when the way to the next exhibit is blocked by visitors. Thus another mechanism might be necessary, in order to ask the visitors to move aside, and then to check for success.

How the interaction between Loomo and the visitors is perceived by non robotic experts has not been tested yet, thus further research might be necessary to improve the intuitiveness of the interaction.

VII. Conclusion

In this paper we presented a prototype guide robot based on the robot platform Segway Loomo. We proved that it can be used to find exhibits in an art gallery autonomously by performing exploration based on a wall follower algorithm and ORB feature detection. Furthermore, we showed that the robot guide is able to give a tour for visitors. This shows that future work on social guide robots could focus more on using available, off-the-shelf robots like Loomo instead of custom made ones, so that they can be used by a larger number of galleries and museums.

References

[1] S. Thrun, M. Bennewitz, W. Burgard, A. B. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte *et al.*, "Minerva: A second-generation museum tour-guide robot," in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 3. IEEE, 1999.

[2] W. Burgard, A. B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun, "Experiences with an interactive museum tour-guide robot," *Artificial Intelligence*, vol. 114, no. 1-2, pp. 3–55, 1999.

[3] Y. Kuno, K. Sadazuka, M. Kawashima, K. Yamazaki, A. Yamazaki, and H. Kuzuoka, "Museum guide robot based on sociological interaction analysis," in *Proceedings of the Conference on Human Factors in Computing Systems*, 2007, pp. 1191–1194.

[4] I. R. Nourbakhsh, C. Kunz, and T. Willeke, "The mobot museum robot installations: A five year experiment," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)*, vol. 4. IEEE, 2003, pp. 3636–3641.

[5] T. Iio, S. Satake, T. Kanda, K. Hayashi, F. Ferreri, and N. Hagita, "Human-like guide robot that proactively explains exhibits," *International Journal of Social Robotics*, pp. 1–18, 2019.

[6] T. Kanda, M. Shiomi, Z. Miyashita, H. Ishiguro, and N. Hagita, "An affective guide robot in a shopping mall," in *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*, 2009, pp. 173–180.

[7] H.-M. Gross, H.-J. Boehme, C. Schröter, S. Müller, A. König, C. Martin, M. Merten, and A. Bley, "Shopbot: Progress in developing an interactive mobile shopping assistant for everyday use," in *2008 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, 2008, pp. 3471–3478.

[8] M. Shiomi, D. Sakamoto, T. Kanda, C. T. Ishi, H. Ishiguro, and N. Hagita, "A semi-autonomous communication robota field trial at a train station," in *2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2008, pp. 303–310.

[9] (Last accessed on 02.06.2020) Segway loomo overview. [Online]. Available: https://developer.segwayrobotics.com/developer/documents/segway-robot-overview.html

[10] (Last accessed on 25.06.2020) iteratec. [Online]. Available: https://www.iteratec.com/

[11] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part i," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[12] S. J. Lee, J. Lim, G. Tewolde, and J. Kwon, "Autonomous tour guide robot by using ultrasonic range sensors and QR code recognition in indoor environment," in *IEEE International Conference on Electro/Information Technology*. IEEE, 2014, pp. 410–415.

[13] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source robot operating system," in *IEEE International Conference on Robotics and Automation workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.

[14] (Last accessed on 02.06.2020) Loomo project iteratec. [Online]. Available: https://github.com/iteratec/Loomo-Ros-App

[15] P. Bovbel. (Last accessed on 02.06.2020) Ros frontier exploration. [Online]. Available: http://wiki.ros.org/frontier_exploration

[16] M. Arruda. (Last accessed on 02.06.2020) Wall follower algorithm. [Online]. Available: https://www.theconstructsim.com/wall-follower-algorithm/

[17] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *International journal of computer vision*, vol. 128, no. 2, pp. 261–318, 2020.

[18] (Last accessed on 25.06.2020) OpenCV. [Online]. Available: https://opencv.org/

[19] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to sift or surf," in *2011 International conference on computer vision*. Ieee, 2011, pp. 2564–2571.

[20] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, "Monte carlo localization: Efficient position estimation for mobile robots," *Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence Conference*, vol. 1999, no. 343-349, pp. 2–2, 1999.