FAST VISION-BASED LOCALIZATION FOR OUTDOOR ROBOTS USING A COMBINATION OF GLOBAL IMAGE FEATURES

Christian Weiss * **Andreas Masselli** * **Andreas Zell** *

* *Department of Computer Science, University of Tübingen, Sand 1, 72076 Tübingen, Germany*

Abstract: In this paper, we present a geometrical localization method based on a combination of global image features. Our method represents each image by two feature vectors. The first feature vector is a *Weighted Gradient Orientation Histogram* (WGOH). The second feature vector is a *Weighted Grid Integral Invariant* (WGII) feature vector based on Integral Invariants. For localization, we use a particle filter which updates the weights of the particles based on image similarities calculated from the two feature vectors. We evaluate our approach on outdoor images of two different areas and under varying illumination and compare it to a SIFT-based approach. The comparison shows that the SIFT approach is slightly more exact than our method, but our method is more than four times faster than the SIFT approach and allows a localization frequency of more than 2 Hz.

Keywords: Outdoor Localization, Integral Invariants, WGOH, SIFT

1. INTRODUCTION

Outdoor localization of mobile robots is a difficult task for many reasons. Some range sensors, like 2D laser range finders, are not well suited for outdoor localization because of the cluttered environment. GPS signals often are not available because the GPS satellites are occluded by buildings or trees. Due to these problems, vision has become the most widely used sensor in outdoor localization. A serious problem for vision are illumination changes, because illumination in outdoor environments is highly dependent on the weather (sunny, cloudy, ...) and on the time of day.

An algorithm which can deal with changing illumination to a relatively high extent is the *Scale Invariant Feature Transform* (SIFT) developed by Lowe (Lowe, 2004). SIFT computes descriptors for local interest points within the image. These local features are more dependent on structure than on illumination and are very distinctive. However, as the number of features per image is large (about 420 for our 320×240 pixel images on average), matching images is very time-consuming. Approaches that use SIFT for indoor localization are, for example, (Se *et al.*, 2001), (Tamimi and Zell, 2005). Outdoor localization using SIFT was presented in (Barfoot, 2005). There also exist methods that replace the gradient histogram features of the SIFT approach, for example by *Local Integral Invariants* (Tamimi *et al.*, 2006). Bradley *et al.* use a technique inspired by SIFT for topological outdoor localization, the so-called *Weighted Gradient Orientation Histograms* (WGOH) (Bradley *et al.*, 2005).

Another group of vision-based localization methods are the appearance-based methods, which compute global features for images. Well-known methods for indoor localization are based on PCA (Artac *et al.*, 2002), (Jogan *et al.*, 2003), (Jogan

et al., 2002) or on *Integral Invariant* features (Wolf *et al.*, 2005). The main advantage of global methods over local techniques is that image similarities can be computed much faster. However, in general, global methods are more sensitive to illumination changes than local methods. There are also methods that combine global and local techniques (Artač and Leonardis, 2004). An overview of global and local features for mobile robot localization can be found in (Tamimi, 2006).

The new method presented in this paper represents every image by two global feature vectors. The first feature vector is the WGOH also used by Bradley *et al.*. The second feature vector is a new one based on Integral Invariants. Similar to Bradley *et al.*, we subdivide the image into a grid of subimages. On each subimage, we compute an 8-bin histogram of Integral Invariants, weighted by the distance of each point to the center of the subimage. We call these new features the *Weighted Grid Integral Invariants* (WGII).

In the training phase, we collect images at known positions. For each image, we extract the two feature vectors, which we store together with the image position. In the localization phase, the robot moves around in the same area and takes new images. To localize the robot, we use a particle filter which updates the weights of the particles according to the image similarities calculated from the feature vectors of the current test image and the training images.

In experiments on outdoor images, we compare our method to a SIFT-based approach. The results show that in general, the SIFT approach is slightly more exact. On the other hand, our method is more than 4 times faster than SIFT.

The rest of the paper is organized as follows. Section 2 describes the WGII and WGOH features in more detail and Section 3 explains the particle filter. Section 4 presents the experimental results and Section 5 concludes the paper.

2. IMAGE FEATURE EXTRACTION

This section describes the WGII and WGOH features we use to represent an image. Additionally, we shortly explain the SIFT technique we compare our method to.

2.1 WGOH

The Weighted Gradient Orientation Histograms for outdoor localization were presented by Bradley *et al.* (Bradley *et al.*, 2005). The WGOHs were inspired by SIFT features (Lowe, 2004) and are similar to features presented by Kosecka and Li (Kosecka and Li, 2004). Bradley *et al.* first split the image into a 4×4 grid of subimages. On each subimage, they calculate an 8-bin histogram of gradient orientations, weighted by the magnitude of the gradient at each point and by the distance to the center of the subimage. In our implementation of WGOHs, we use a 2D gaussian for weighting, where the mean corresponds to the center of the subimage and the standard deviations correspond to 0.5 times the width and the height of the subimage, respectively. We took these parameters from SIFT, where a gaussian with half the width of the descriptor window is used for weighting. The 16 histograms are concatenated to a 1×128 feature vector, which is normalized subsequently. To reduce the dependency on particular regions or some strong gradients, the elements of the feature vector are thresholded to 0.2, and the feature vector is renormalized.

We use the WGOH features in our method, because Bradley *et al.* obtain good result using these features for topological localization in outdoor environments. The features are also relatively robust to illumination changes in Bradley's experiments.

2.2 WGII

Global Integral Invariant features are features which are invariant to euclidean motion, i.e. rotation and translation, and to some extent robust to illumination changes. The key idea is to apply all possible translations (t_0, t_1) and rotations r to the image and to calculate the features by averaging over all the transformed versions of the image. For an image $\mathbf{I}$ with $N_0 \times N_1$ pixels,

$$F(\mathbf{I}) = \frac{1}{RN_0N_1} \cdot \sum_{t_0=0}^{N_0-1} \sum_{t_1=0}^{N_1-1} \sum_{r=0}^{R-1} f\left(g\left(t_0, t_1, \phi = 2\pi\frac{r}{R}\right)\mathbf{I}\right) \quad (1)$$

computes the Global Integral Invariant feature $F(\mathbf{I})$ of image $\mathbf{I}$. Here, R is the number of different rotation angles, g is an element of the group of euclidean motions, $g\mathbf{I}$ is the image $\mathbf{I}$ transformed by g, and f is a kernel function. The kernel function involves the local neighborhood of a pixel in the calculation. For example, the *monomial kernel* multiplies the intensities of two neighborhood pixels that lie on circles with certain radius around the center pixel and that have a certain phase shift. In our method, we use the *relational kernel*, which is invariant to strict illumination changes. For two pixel coordinates $p_1 = (x_1, y_1)$ and $p_2 = (x_2, y_2)$, the relational kernel is given by

$$f(\mathbf{I}) = rel\left(\mathbf{I}(x_1, y_1) - \mathbf{I}(x_2, y_2)\right), \quad (2)$$

with the ramp function

$$rel(\gamma) = \begin{cases} 1 & \text{if } \gamma < -\epsilon, \\ \dfrac{\epsilon - \gamma}{2\epsilon} & \text{if } -\epsilon \leq \gamma \leq \epsilon, \\ 0 & \text{if } \epsilon < \gamma. \end{cases} \quad (3)$$

To create a more distinctive feature for an image, one can compute a histogram of the Integral Invariants evaluated at each pixel instead of representing the image by a single number. It is also possible to use more than one kernel and to form a multidimensional histogram as feature for an image. A more detailed description of Global Integral Invariants can be found in (Siggelkow, 2002).

Experimentally, we found that ordinary Global Integral Invariant features are not distictive enough for outdoor localization, even when using 3 kernels to form multidimensional histograms. Thus, we adopt the technique of Bradley *et al.* to calculate individual histograms on a grid of subimages. We also calculate weighted histograms such that Integral Invariant features for pixels near the center of a subimage get a higher weight than pixels near the borders of subimages, because the pixels near the borders are more likely to fall into another subimage under image translations or rotations.

Thus, we first compute the Integral Invariants for each pixel of the image. We then split the image into a 4×4 grid of subimages. We use a 4×4 grid, because coarser grids led to decreased performance in our experiments, and finer grids did not further improve performance. On each subimage, we calculate a weighted 8-bin histogram of Integral Invariant features. For weighting, we use a 2D gaussian with mean at the center of the subimage and with standard deviations equal to 0.25 times the width and the height of the subimage, respectively. Then we concatenate the 16 histograms and normalize the resulting vector to get the final 1×128 WGII feature vector for the image.

In the relational kernel, we use the pixel coordinates $p_1 = (10, 0)$ and $p_2 = (0, 20)$. We set the parameter ϵ to 0.098 and the number of rotation angles to $R = 10$. We chose these kernel parameters, because experimentally, they led to the best results.

2.3 Image Matching

To calculate the similarity between two images $\mathbf{Q}$ and $\mathbf{D}$, we compare their feature histograms $\mathbf{q}$ and $\mathbf{d}$ using normalized histogram intersection

$$\bigcap_{\mathbf{norm}} (\mathbf{q}, \mathbf{d}) = \frac{\sum_{k \in \{0,1,\ldots,m-1\}} \min(q_k, d_k)}{\sum_{k \in \{0,1,\ldots,m-1\}} q_k}, \quad (4)$$

where m is the number of histogram bins.

2.4 SIFT

For comparison to our method, we use a localization approach based on SIFT (Lowe, 2004). In this approach, the most similar training image to a test image is the one which contains the highest number of local features that can be matched to the local features of the test image.

To speed up the SIFT-based localization, we use an additional step that reduces the number of features of each image. The idea is to delete "noisy" SIFT-features, which are likely not to appear in more than one image. In the training phase, we match each training image to the two neighboring training images. We only keep the features that can be matched to a feature of at least one of the two neighboring images. In the localization phase, we only keep the features of the test image that can be matched to a feature of the previous test image. Depending on the dataset, this technique removes about 50 to 80% of the features, and matching images is accelerated by a mean factor of about 5 without loss of accuracy.

3. PARTICLE FILTER

For localization, we use a particle filter (Thrun *et al.*, 2000). We update the weights of the particles based on image similarities calculated from the WGOH and WGII features, and accordingly based on SIFT in the method we use for comparison.

Particle filters represent the belief $Bel(x)$ of the robot about its position by a set of m particles. In our case, each particle consists of a pose (x, y) together with a non-negative *importance factor*, which determines the weight or importance of the particle. The estimated pose of the robot is given by the weighted mean of the particles. For global robot localization, the inital particles are randomly distributed over the robot's universe. All importance factors are set to $\frac{1}{m}$. The particles are updated for each test image using 3 steps:

(1) Draw m random particles $x_{t-1}^{(i)}$ from $Bel(x_{t-1})$ according to the importance factors p_{t-1} at time $t-1$.
(2) Update the sample $x_{t-1}^{(i)}$ to sample $x_t^{(j)}$ according to an action u_{t-1}. As we do not use a motion model, for example from odometry, we randomly update the particle according to a gaussian distribution centered at the position of the particle and with a standard deviation of 4 m. Additionally, we move each particle a short distance d in the direction to the nearest training image, where d corresponds to 0.2 times the distance of the particle to the nearest training image.

Fig. 1. Our RWI ATRV-JR robot "Arthur".

(3) Weight the sample $x_t^{(j)}$ by the likelihood of the sample $x_t^{(j)}$ given the measurement y_t. To assign new weights, we first search the nearest training image to each particle. In the case of SIFT, we perform a standard SIFT match between the test image and the chosen training image, and the score of the match becomes the new weight of the particle. In our new method, we match the test image to a training image by comparing the WGII and WGOH feature vectors individually, using normalized histogram intersection. We set the new weight of the corresponding particle to the product of the WGII and WGOH matches. We additionally multiply the new weight by a factor that decreases with the distance of the particle to its nearest training image. In our new method, we then potentiate the new weight by 20, because the differences between the matching scores are all low (but anyhow distinctive at that low level). This way the difference between the matches becomes clearer.

After the third step, we normalize the importance factors and calculate the estimated position. Before repeating the three steps for the next test image, we replace the worst 5% of the particles by randomly generated new ones. This way the robot can recover its position if the position was lost or if the robot was kidnapped.

To speed up the calculation of the weights, we save for each particle the matching result to the test image. If another particle has the same nearest training image, we do not have to recalculate the match, but can use the saved value. In the case of SIFT, this method speeds up the estimation of a new position by a factor of about 5. For our new method, we only get a slight speedup.

4. EXPERIMENTAL RESULTS

In our experiments, we use images collected by our RWI ATRV-JR outdoor robot (Fig. 1). We moved the robot around using a constant velocity of about 0.6 m/s. Once per second, we took a 320×240 pixel grayscale image with the left camera of the robot's stereo camera system. The robot is also equipped with a Differential GPS (DGPS) system, which we used to get the position of each image. However, due to occlusion by trees and buildings, the GPS path sometimes significantly deviated from the real position or contained gaps. As we always moved the robot on a smooth trajectory, we corrected some wrong or missing GPS values manually by linearly interpolating between the positions before and after the error or gap.

We collected two datasets that differ in the type of environment. Dataset 1 consists of six rounds around a big building. Each of the rounds is about 260 m long and is represented by about 400 images. We collected three of the rounds under sunny conditions, but there are also some short sections (about 5 to 10 s long) during which the sun was covered by clouds. We collected the other three rounds about six weeks later on a cloudy day. The images of dataset 1 contain many artifical objects like buildings, streets and cars. Additionally, there are some dynamic objects like cars and people passing by.

We acquired dataset 2 in a different area mostly containing vegetation like grass, bushes and trees. We recorded two rounds in the early afternoon, in which the sun was shining brightly. In the evening, we collected the third and fourth round. The sun was covered by clouds and it was starting to get dark. Each round of dataset 2 is about 220 m long and consists of about 350 images. Fig. 3 a) and b) show the GPS ground truth data of dataset 1 and 2. Fig. 2 shows example images of dataset 1 and 2 under different illumination.

For evaluation, we calculated the error of all possible training/test combinations of rounds using $m = 300$ particles. Additionally, we repeated each experiment n times, where n is the number of test images. For each of these experiments, we used a different test image as starting image for the localization. Then we calculated the mean error of all experiments that are similar, e.g. all experiments in which we used the sunny images of dataset 1 for training and the cloudy images for testing.

The columns WGOH, WGII2 and WGII1 of Tab. 1 show the localization errors when representing images by a single feature vector, i.e. by WGOH, by WGII with a two-dimensional histogram calculated from two kernels and by WGII with one kernel. The table shows that WGOH features produce large errors on dataset 2 under changing illumination. The WGII features calculated using two kernel functions perform well on these images, but have problems on images

Table 1. Mean localization errors ± standard deviation (m)

dataset	training im.	test im.	WGOH	WGII2	WGII1	WGII + WGOH	SIFT
dataset 1	sunny	sunny	3.04 ± 1.06	3.53 ± 1.22	4.17 ± 1.18	3.15 ± 1.20	2.15 ± 0.29
	cloudy	cloudy	1.85 ± 0.29	2.18 ± 0.71	2.37 ± 0.38	1.60 ± 0.26	2.06 ± 0.56
dataset 2	sunny	sunny	1.65 ± 0.16	1.48 ± 0.11	1.82 ± 0.13	1.09 ± 0.06	1.78 ± 0.05
	cloudy	cloudy	2.38 ± 0.89	3.79 ± 0.51	3.95 ± 0.72	2.07 ± 0.76	2.10 ± 0.14
dataset 1	sunny	cloudy	3.69 ± 0.62	5.97 ± 1.68	9.38 ± 1.70	3.95 ± 0.54	3.27 ± 0.27
	cloudy	sunny	3.64 ± 0.85	6.10 ± 1.96	8.12 ± 2.10	3.85 ± 0.79	2.52 ± 0.17
dataset 2	sunny	cloudy	9.44 ± 6.25	3.82 ± 0.44	5.13 ± 0.55	3.64 ± 0.80	2.88 ± 0.20
	cloudy	sunny	6.28 ± 1.65	3.65 ± 0.01	5.18 ± 0.80	3.45 ± 0.32	2.74 ± 0.24

Fig. 2. Example images. a) Dataset 1, sunny. b) Dataset 1, cloudy. c) Dataset 2, sunny. d) Dataset 2, cloudy.

of dataset 1 under changing illumination. The results for the WGII features computed from one kernel show a similar characteristic, except that the errors are larger.

Column WGII + WGOH of Tab. 1 shows the results for the combined method. Here, we use WGII features computed using one kernel, because the errors are only about 0.2 m larger than when using WGII features computed from two kernels, and WGII1 is about twice as fast as WGII2. In all cases, the error of the combined method is lower or only slightly worse than the error of the better one of the single features. Especially in the difficult cases, i.e. the experiments with changing illumination, the combined method is much more robust than using only a single feature vector.

Fig. 3 presents a comparison between our new method and the SIFT approach. The errors of both approaches decrease rapidly during the first few iterations of the particle filter. In the experiments with constant illumination, the mean errors of WGII + WGOH and SIFT are relatively similar. In experiments using changing illumination conditions, the errors of both methods increase, where the increase for the WGII + WGOH features is more significant than for SIFT. Thus, SIFT seems to be more robust against illumination changes, but our method also performs well.

The main advantage of our method over the SIFT approach is its speed, illustrated by Fig. 4. Localization of one image using SIFT takes 1.696 s on our robot, which is equipped with a 1.8

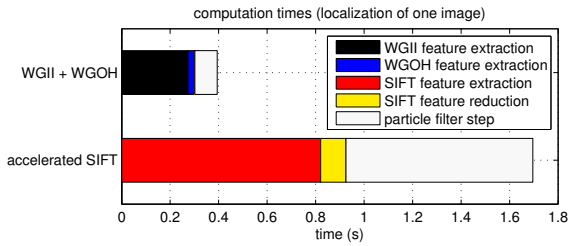

Fig. 4. Time for localization of one test image.

Ghz Pentium M Processor and 1 GB of RAM. This time is composed of 0.821 s for SIFT feature extraction, 0.104 s to reduce the number of SIFT features and 0.771 s for a particle filter step. In contrast, average localization of one test image using our WGII + WGOH method is possible in 0.394 s, where computation of the WGII feature vector takes 0.274 s, computation of the WGOH feature vector takes 0.028 s, and one particle filter step takes 0.093 s.

These numbers show that there is a trade-off between accuracy and speed. The mean localization error of our method is about 1.14 times the error for the SIFT approach on average (and maximally about 1.5 times the error of SIFT). On the other hand, localization using our approach is more than 4 times faster than using SIFT.

5. CONCLUSION

We presented a new method for vision-based localization for outdoor mobile robots. We represent images by two global 1×128 image feature vectors, the new Weighted Grid Integral Invariant (WGII) vector, which is based on Integral Invariants, and a Weighted Gradient Orientation Histogram (WGOH). We use these feature vectors to update the weights of a particle filter.

A comparison to a SIFT-based approach showed that the error created by our method is about 1.14 times the error of the SIFT approach on average. However, localization of one test image using our method is possible in about 0.4 s, which is more than four times faster than the SIFT approach.

For the future, we plan a combination between our method and SIFT, which selects the method based on how sure the robot is about its position.

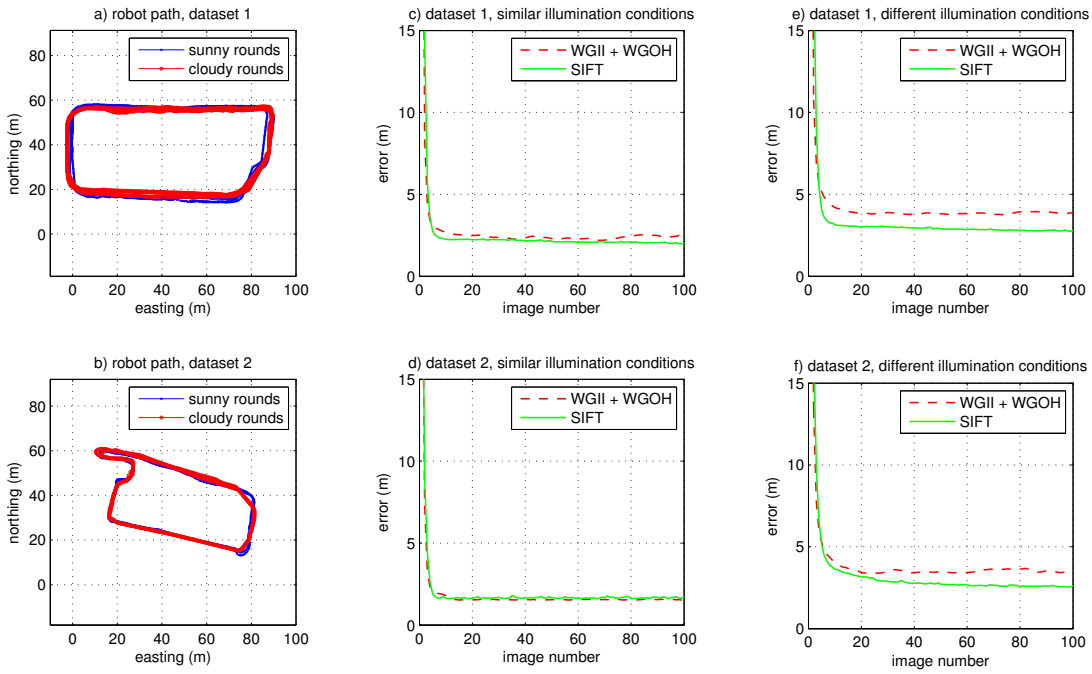

Fig. 3. a-b) GPS data. c-f) Mean errors for particle filter experiments. There is no significant change after image 100. The mean initial error is about 36 m for dataset 1 and about 26 m for dataset 2.

REFERENCES

Artac, M., M. Jogan and A. Leonardis (2002). Mobile robot localization using an incremental eigenspace model. In: *Proc. IEEE/RSJ Intl. Conf. on Robotics and Automation (ICRA 2002)*. Washington, D. C.. pp. 1025–1030.

Artač, M. and A. Leonardis (2004). Outdoor mobile robot localisation using global and local features. In: *Proc. 9th Computer Vision Winter Workshop (CVWW)*. Piran. pp. 175–184.

Barfoot, T.D. (2005). Online visual motion estimation using fastslam with sift features. In: *Proc. IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS 2005)*. Edmonton, Canada. pp. 3076 – 3082.

Bradley, D.M., R. Patel, N. Vandapel and S.M. Thayer (2005). Real-time image-based topological localization in large outdoor environments. In: *Proc. IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS 2005)*. Edmonton, Canada. pp. 3062 – 3069.

Jogan, M., A. Leonardis, H. Wildenauer and H. Bischof (2002). Mobile robot localization under varying illumination. In: *Proc. Intl. Conf. on Pattern Recognition (ICPR02)*. Quebec, Canada. pp. 741–744.

Jogan, M., M. Artac, D. Skocaj and A. Leonardis (2003). A framework for robust and incremental self-localization. In: *Proc. 3rd Intl. Conf. on Computer Vision Systems (ICVS 2003)*. Graz, Austria. pp. 460–469.

Kosecka, J. and F. Li (2004). Vision based topological markov localization. In: *Proc. Intl. Conf. on Robotics and Automation (ICRA)*. New Orleans, LA, USA. pp. 1481–1486.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *Int. Journal of Computer Vision* **60**(2), 91–110.

Se, S., D. Lowe and J. Little (2001). Local and global localization for mobile robots using visual landmarks. In: *Proc. IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*. Maui, Hawaii. pp. 414 – 420.

Siggelkow, S. (2002). Feature Histograms for Content-Based Image Retrieval. PhD thesis. Institute for Computer Science, University of Freiburg. Freiburg, Germany.

Tamimi, H. (2006). Vision-based Features for Mobile Robot Localization. PhD thesis. University of Tübingen. Tübingen, Germany.

Tamimi, H., A. Halawani, H. Burkhardt and A. Zell (2006). Appearance-based localization of mobile robots using local integral invariants. In: *Proc. 9th Intl. Conf. on Intelligent Autonomous Systems (IAS-9)*. Tokyo, Japan. pp. 181–188.

Tamimi, H. and A. Zell (2005). Global robot localization using iterative scale invariant feature transform. In: *36th Intl. Symposium on Robotics (ISR 2005)*. Tokyo, Japan.

Thrun, S., D. Fox, W. Burgard and F. Dellaert (2000). Robust monte carlo localization for mobile robots. *Artificial Intelligence* **128**(1-2), 99–141.

Wolf, J., W. Burgard and H. Burkhardt (2005). Robust vision-based localization by combining an image retrieval system with monte carlo localization. *IEEE Transactions on Robotics* **21**(2), 208–216.