Particle Filter Localization on Continuous Occupancy Maps

Alberto Y. Hata[1], Denis F. Wolf[1], and Fabio T. Ramos[2]

[1]Mobile Robotics Laboratory, University of São Paulo
[2]School of Information Technologies, University of Sydney
{hata,denis}@icmc.usp.br,fabio.ramos@sydney.edu.au

Abstract. Occupancy grid maps have been widely used for robot localization. Despite the popularity, this representation has some limitations, such as requirement of discretization of the environment, assumption of independence between grid cells and necessity of dense sensor data. Suppressing these limitations can improve the localization performance, but requires a different representation of the environment. Gaussian process occupancy map (GPOM) is a novel representation based on Gaussian Process that enables the construction of continuous maps (i.e. without discretization) using few laser measurements. This paper addresses a new localization method that uses GPOM to estimate the robot pose in areas not directly observed during mapping and generally provides higher accuracy compared to occupancy grid maps localization. Specifically, we devised a novel likelihood model based on the multivariate normal probability density function and adapted the particle filter localization method to work with GPOM. Experiments showed localization errors more than three times lower in comparison with particle filter localization using occupancy grid maps.

Keywords: Gaussian Process Occupancy Maps, Gaussian Process, Particle Filter Localization, Occupancy Grid Map, Sparse Laser Sensor Data.

1 Introduction

Localization is an important problem in mobile robotics because of its direct impact on other robot tasks, such as path planning and environment mapping. Most of the localization methods use a map of the environment as a reference to correct the robot position. The basic idea is to find the position that maximizes the correspondence between the current sensor data and the map. The occupancy grid maps (OGMs) have been widely applied to this task due to its relative simplicity to build and speed [10].

Despite the popularity of OGMs, they have some drawbacks. First, this representation requires the discretization of the environment, thus the localization precision is limited to the chosen map resolution. Second, independence between grid cells is assumed. Therefore, the occupancy information of neighboring cells is not taken into account during the construction of the map. As a result, for a detailed map, OGMs require dense sensor measurements and intense exploration of the environment.

To overcome the limitations of OGM, O'Callaghan and Ramos [6] proposed a novel mapping method named Gaussian Process Occupancy Map (GPOM) that uses Gaussian Process (GP) to enable continuous representation of the environment using a dependent model of the sensor measurements. With GPs, the occupancy status of any place in the environment can be estimated; it is possible to reconstruct non-observed areas, even from highly sparse sensor data.

In order to relax the requirement of a dense, accurate and high resolution maps for localization, in this paper, we propose the use of GPOM representation for robot localization. Another advantage of our approach is the possibility to use low-resolution sensors which reduces the size of datasets allowing for lower cost range sensors. As the existing localization methods are designed to work with discrete maps, here we modify the popular Particle Filter localization (PFL) to support continuous occupancy maps (GPOMs) instead. Specifically, a novel observation likelihood model is proposed to update the particles. The main advantage of our approach is the use of occupancy uncertainty information returned by the GP to incorporate the degree of accuracy of the map in the likelihood model. In this way, the association of GPOM with PFL makes the localization more robust to noisy measurements and enables an accurate localization in areas not observed in detail during mapping.

Few studies have explored the use of alternative map representations for robot localization. For instance, Yang and Wang [10] proposed OGM maps that include the probability of each cell containing a moving obstacle. Then, this map was used for localization in dynamic environments.Despite the lack of works covering localization in continuous occupancy maps, there are others that apply GP learning for localization. In [2], the authors used GP for SLAM problem based on WiFi signal. [4] proposed the integration of GP with Bayesian filters for blimp tracking. The work of [7] simulated a dense sensor measurement by training a GP model from sparse laser data. In this, a new GP model is trained for each new scan. A similar approach was adopted in [1], but cameras were used to extract the features to build the GP model.

Differently from previous works, we are applying GPs to generate a more detailed representation of the environment and enable the robot to precisely localize at any location of the map with a reduced number of laser beams.

2 Gaussian Process Occupancy Maps (GPOMs)

A GP is a multivariate Gaussian distribution modeling the function space of a dataset $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^{N}$, where $\mathbf{x}_i \in \mathbb{R}^D$ and $y_i \in \mathbb{R}$. The model is fully specified by a mean function $\mu(\mathbf{x})$, and a covariance function $k(\mathbf{x}, \mathbf{x}')$. In the mapping context, $\mathbf{x}_i$ and y_i corresponds respectively to a laser beam and its occupancy status.

In GPs, given a query data point $\mathbf{x}_*$ and the training dataset $\mathcal{D}$, the posterior y_* is also a Gaussian:

$$p(y_*|\mathcal{D}, \mathbf{x}_*) = \mathcal{N}(\mu(\mathbf{x}_*), \sigma(\mathbf{x}_*)). \tag{1}$$

The mean and the variance of the posterior are represented respectively by $\mu(\mathbf{x}_*)$ and $\sigma(\mathbf{x}_*)$, which are obtained by:

$$\mu(\mathbf{x}_*) = k(\mathbf{x}_*, \mathbf{x})^T [k(\mathbf{x}, \mathbf{x}) + \sigma_n^2 I]^{-1} y, \tag{2}$$

$$\sigma(\mathbf{x}_*) = k(\mathbf{x}_*, \mathbf{x}_*) - k(\mathbf{x}_*, \mathbf{x})[k(\mathbf{x}, \mathbf{x}) + \sigma_n^2 I]^{-1} k(\mathbf{x}_*, \mathbf{x}), \tag{3}$$

where σ_n is the global noise value. The Gaussian Process of $\mathbf{x}_*$ is denoted as $\mathcal{GP}(\mu(\mathbf{x}_*), \sigma(\mathbf{x}_*))$.

For GPOM, generally the Matérn 3/2 covariance function is employed [6] because it is less smooth compared to the squared exponential covariance function, which is more convenient to represent the occupancy values in space:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f \left(1 + \frac{\sqrt{3}\,|\mathbf{x} - \mathbf{x}'|}{l}\right) \exp\left(-\frac{\sqrt{3}\,|\mathbf{x} - \mathbf{x}'|}{l}\right), \tag{4}$$

where σ_f and l are respectively, the signal variance and length-scale of the covariance function.

The main advantage of GP is its capacity of learning hyper-parameters for both mean and covariance functions by maximizing the marginal likelihood of the data. The hyperparameters of the covariance function, given by $\theta = \{\sigma_f, l\}$ are obtained through the maximization of the log marginal likelihood function [8]:

$$\log p(y|\mathbf{x}, \theta) = \frac{1}{2} y^T K^{-1} y - \frac{1}{2} \log |K| - \frac{n}{2} \log 2\pi, \tag{5}$$

where K is the covariance matrix of the training dataset $\mathcal{D}$ with size n.

GPs were originally proposed to solve regression problems. Therefore, adaptations in the original formulation were made for classification problems, as in the case of GPOM. For GP classification, the mean value resulted from regression is squashed into $[0, 1]$ interval using a sigmoid function. For this, the probabilistic least squares function [8] is used:

$$p(occupancy|\mathcal{D}, \mathbf{x}_*) = \Phi\left(\frac{\alpha\mu(\mathbf{x}_*) + \beta}{1 + \alpha^2 \sigma(\mathbf{x}_*)^2}\right), \tag{6}$$

where Φ is the cumulative Gaussian distribution and the parameters α and β are obtained by leave-one-out cross-validation.

3 GPOMs for Large Environments

Considering the $\mathcal{O}(n^3)$ cost of the GP prediction, the construction of GPOMs for large areas involving millions of sensor measurements is computationally unfeasible. The major bottleneck in the Gaussian Processes regression is associated to the matrix inversion K^{-1}, which involves the solution of a large linear system. In order to reduce the complexity, Cholesky decomposition can be applied to update the inverted matrix as new measurements are obtained. This reduces the matrix inversion complexity from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$.

Despite reducing the GP complexity, depending on the amount of measurements, Cholesky decomposition alone can be insufficient for suitable computation times. Therefore, two other approaches are also employed: (a) reduced training dataset and (b) GP committees. These methods are described in the following sections.

3.1 Information Theoretic Compression of Training Data

When gathering sensor data there are generally redundant measurements that could be eliminated to speed up the computation. Unnecessary data can be discarded by adapting the information-theoretic compression of laser data method proposed in [5]. The idea behind is to evaluate the mutual information between the measurement and the existing dataset. Only laser readings that reduce the uncertainty about the environment the most are kept in the dataset.

3.2 Mixture of Gaussian Processes

Instead of generating a single map, the environment can be split into smaller regions and then produce a set of GPOMs. The strategy of using several GPs is named mixture of GPs and its application for mapping was proposed in [3].

The first step of the mixture of GPOMs is to cluster the data (training dataset) according to a distance measure. Here, k-means clustering was employed. Given the maximum number of measurements s that each GP expert must handle, the number of clusters is set as $k \geq \frac{n}{s}$. After clustering, each measurement will be associated with a centroid $\{c_{i=1\cdots k}\}$ and a GP expert $\{\varepsilon_{i=1\cdots k}\}$.

When building a dense map, a set of test points $D = \{d_{j=1\cdots m}\}$ must be evaluated. A gating network evaluates which expert should be chosen to infer the occupancy of test points. For a test point d_j, we associate the expert ε_i whose corresponding cluster c_i is the closest to d_j.

4 Particle Filter Localization for GPOM

After generating the GPOM, particle filter localization (PFL) uses the map information to estimate the robot position. The standard PFL algorithm starts by randomly distributing particles over the environment. The particle set is represented by $\mathbf{S}_k = \left\{\mathbf{s}_k^i\right\}_{i=1}^n$, where n is the number of particles and k represents the time stamp. Each particle $\mathbf{s}_k^i$ stores the position $\mathbf{x}_k^i = \{x_k^i, y_k^i, \theta_k^i\}$ and the importance weight w_k^i of the particle. For each PFL iteration, $\mathbf{S}_k$ is updated through an auxiliary particle set $\mathbf{S}'_k$. The following steps comprise the particle set update:

1. Perform motion update from the probability $p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{u}_k)$, where $\mathbf{u}$ is the robot motion.

2. Compute the measurement update using the probability $p(\mathbf{z}_k|\mathbf{x}_k^i, m)$, which corresponds to the particle weight w_k^i and also known as measurement likelihood function
$$w_k^i = p(\mathbf{z}_k|\mathbf{x}_k^i, m), \tag{7}$$
where, $\mathbf{z}$ and m correspond to the sensor measurement and the environment map, respectively.
3. Choose randomly with replacement n particles from $\mathbf{S}'_k$. Particles of higher weights have higher probability to be selected. The chosen particles replace the current $\mathbf{S}_k$ set. The particle with the highest weight in $\mathbf{S}_k$ is chosen to represent the robot position.

The second step of PFL is the most crucial, because it directly affects the localization robustness. Here we devise a new likelihood function to use together with GPOM.

In our measurement likelihood function, occupancy and geometric information are used. To calculate the occupancy, first the laser end point pose is calculated from the laser range and the particle's position:

$$\mathbf{p}_k^i = \{x_k^i + r_k^j \cos(a_k^j + \theta_k^i); y_k^i + r_k^j \sin(a_k^j + \theta_k^i)\}, \tag{8}$$

where r_k^j and a_k^j denote the measurement $\mathbf{z}_k$ range distances and beam angles, respectively, and j denotes the beam index.

From a particle $\mathbf{s}_k^i$, occupancy mean $\boldsymbol{\mu_o}$ and variance $\boldsymbol{\sigma_o}$ is obtained from each laser beam end point through GP prediction:

$$\boldsymbol{occupancy}_k^j = \mathcal{GP}(\boldsymbol{\mu_o}(\mathbf{p}_k^i), \boldsymbol{\sigma_o}(\mathbf{p}_k^i)). \tag{9}$$

However, using just the occupancy information can easily lead to ambiguous positions. Therefore, it is also necessary to include geometric information of the laser measurements into the likelihood model. We used the distance $\boldsymbol{\mu_r}$ between the particle to the first point along the laser beam direction which has occupancy mean higher than 0.99. Essentially, a laser measurement is simulated in the particles' pose to determine how far the particle is to the obstacle. The variance of the distance $\boldsymbol{\sigma_r}$ is given by the squared difference from the range distance returned by the sensor and $\boldsymbol{\mu_r}$.

Thus, the following array of mean and variance values are obtained from a single particle $\mathbf{s}_k^i$, given the map and measurement information:

$$\boldsymbol{\mu}_k^i = \begin{bmatrix} \boldsymbol{\mu}_o^i \\ \boldsymbol{\mu}_r^i \end{bmatrix}_k \qquad \boldsymbol{\sigma}_k^i = \begin{bmatrix} \boldsymbol{\sigma}_o^i \\ \boldsymbol{\sigma}_r^i \end{bmatrix}_k, \tag{10}$$

where $\boldsymbol{\mu}_r^i$ and $\boldsymbol{\sigma}_r^i$ represent respectively the mean and variance of range measurement error of the particle $\mathbf{s}_k^i$.

To calculate the observation likelihood, the normal distribution of the current laser measurement is compared with the normal distribution obtained from the occupancy and geometric information of a particle. This is done by connecting the measurement and the particle information in the multivariate normal probability density function:

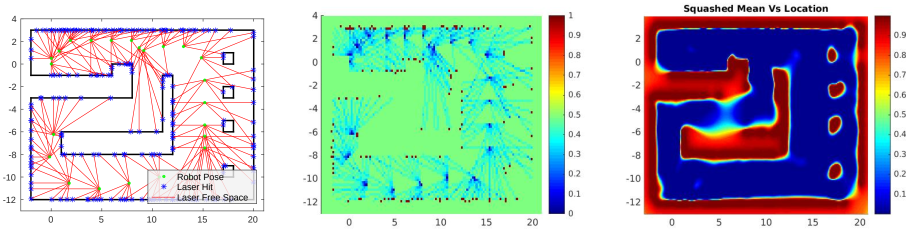
$$p(\mathbf{z}_k|\mathbf{x}_k^i, m, \boldsymbol{\mu}_k^i, \boldsymbol{\Sigma}_k^i) = \frac{1}{2\pi^{\frac{N}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp^{\frac{1}{2}(z_k-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(z_k-\boldsymbol{\mu})}, \tag{11}$$

where, $\boldsymbol{\Sigma}$ is the covariance matrix with $\boldsymbol{\sigma}$ as the diagonal values and N is the $\boldsymbol{\mu}$ array size. For numerical stability, $\log(p(\mathbf{z}_k|\mathbf{x}_k^i, m))$ is used to update the particle weight w_k^i. Here, z_k is an array formed by the beam end point occupancy values (sequence of ones) and the ranges returned by the laser sensor (r_k^j).

5 Results

For the validation experiments, we used laser data collected in simulated and in real environments. In the simulated scenario, we tested the global localization performance in maps with areas that was intentionally not observed and using sparse laser readings. Experiments with real robots and noisy laser measurements in larger areas were conducted using publicly available datasets from Freiburg[1] and Rawseeds[2] repositories. From the Freiburg repository we used the Seattle and Belgioioso datasets, while from Rawseeds we used the Bicocca dataset. In all experiments, distinct measurement sets were used for mapping and localization tasks.

In the mapping stage, OGMs were generated with 0.10 m resolution cell which can provide fine details of the scene. We used standard GPOM for all scenarios except the Bicocca dataset which used the strategy for large maps. Here the combination of PFL with OGM is named OGM-PFL and the combination of PFL

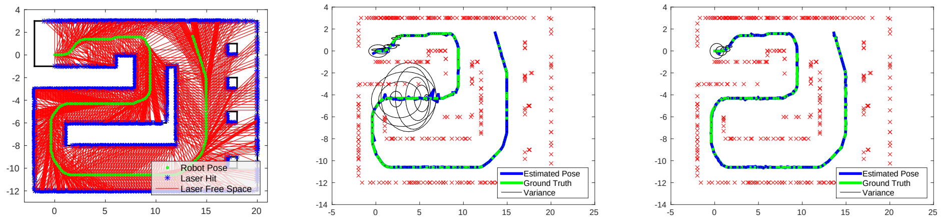


(a) Mapping dataset (b) OGM (0.10 m res.) (c) GPOM

Fig. 1. Sparse laser dataset (a) used to build the OGM (b) and GPOM (c) representations. Few measurements were intentionally gathered from the central corridor. GPOM could recreate the corridor area and reconstruct objects of the scenario (e.g. boxes on the right side), as opposite to OGM.

[1] `http://www2.informatik.uni-freiburg.de/~stachnis/datasets/`

[2] `http://www.rawseeds.org/home/category/benchmarking-toolkit/datasets/`



(a) Localization dataset (b) OGM-PFL 0.10 m (c) GPOM-PFL

Fig. 2. Estimated poses (blue line) compared to ground truth (green line). Particles' variance is represented by the ellipses. The OGM-PFL (b) variance increases around the corridor, while the GPOM-PFL (c) variance stay low.

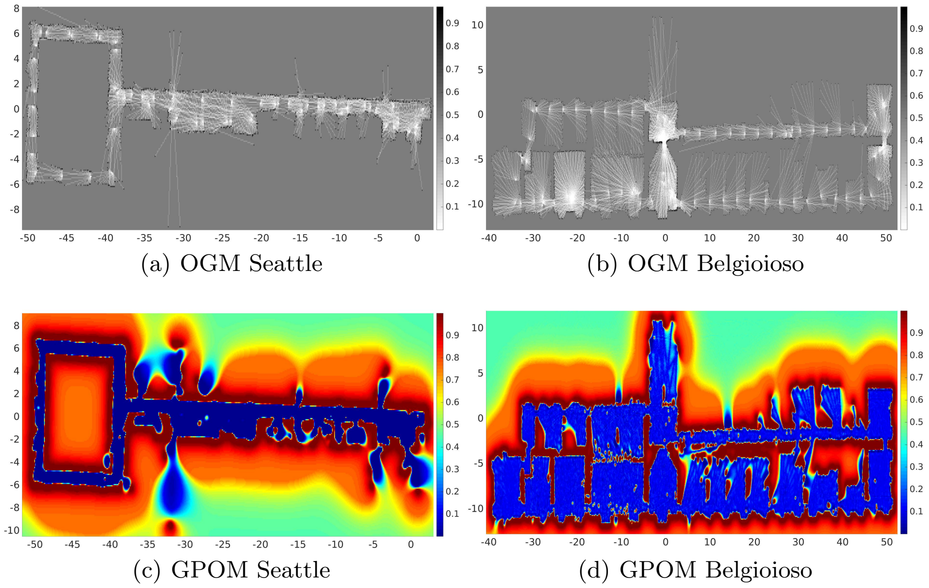
Map	Mean ATE (m)	S. Dev. ATE (m^2)	Ori. Error (rad)
OGM - Simulated	0.1095	0.9987	0.0238
GPOM - Simulated	0.0376	0.3330	0.0073
OGM - Seattle	0.3645	2.6835	0.0404
GPOM - Seattle	0.2612	2.9229	0.0351
OGM - Belgioioso	0.1939	2.1926	0.0304
GPOM - Belgioioso	0.1419	1.5198	0.0232
OGM - Bicocca	0.7410	5.5250	1.9680
GPOM - Bicocca	0.1194	0.2867	0.0252

Table 1. ATE, standard deviation and orientation error of the localization experiments. The GPOM solution outperformed OGM in all scenarios.

with GPOM is named GPOM-PFL. We used the multivariate normal probability density function of Equation 11 to evaluate the likelihod in the GPOM-PFL. In all experiments we performed global localization using 1000 particles. The localization results were evaluated through the Absolute Trajectory Error (ATE) metric [9] that calculates the Euclidean distance between the estimated and ground truth poses.

For the simulated dataset, we first built the maps using the training data, composed by 22 measurements with 17 laser beams each (less than $\frac{1}{10}$ of the standard laser range sensors). The obtained OGM and GPOM are illustrated in Figure 1. Higher occupancy probabilities are associated to values closer to 1.0. Even with sparser measurements, GPOM could reconstruct the scenario with richer details than OGM. For example, it could estimate the occupancy of non-observed areas (the narrow corridor) and reconstruct some objects (boxes on the right side).

Using these maps, the OGM-PFL and GPOM-PFL methods were evaluated using the test data shown in Figure 2(a). This dataset contains 221 scans, each



(a) OGM Seattle

(b) OGM Belgioioso

(c) GPOM Seattle

(d) GPOM Belgioioso

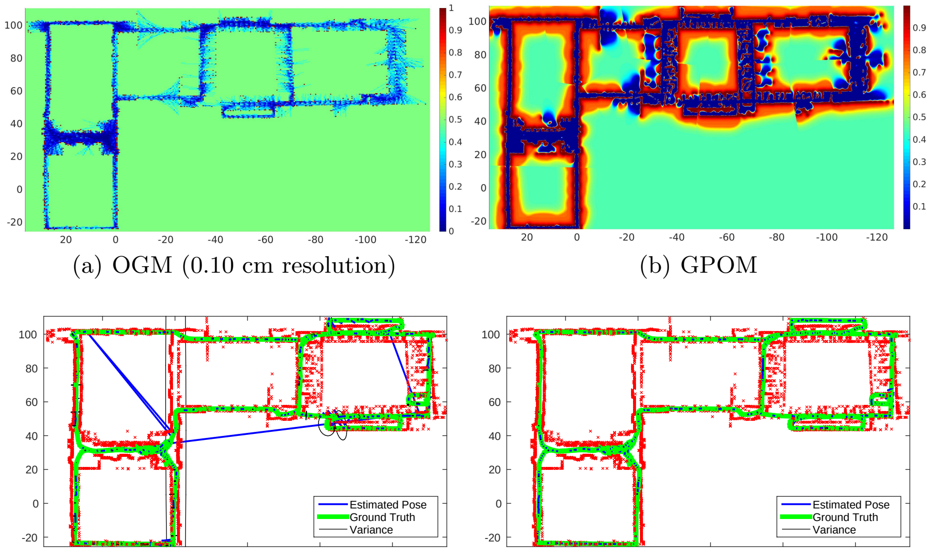
Fig. 3. Mapping results of the Freigburg dataset.(a)-(b) occupancy grid maps and (c)-(d) Gaussian process occupancy maps.

with 80 laser beams. The ATE and absolute orientation error values are in Table 1. GPOM-PFL delivered results almost three times lower than OGM-PFL. Figure 2(b–c) shows the estimated poses (blue line) and particle variance (ellipses) for both approaches. We can notice that OGM-PFL produces higher errors around the narrow corridor, as opposite to GPOM-PFL.

For the Seattle and Belgioioso datasets, we performed the same mapping and localization experiments. These environments are larger compared to the simulated dataset and can test the mapping and localization performance using noisy laser measurements. During mapping, 241 and 132 measurements from Seattle and Belgioioso datasets were used, respectively. Each measurement contains 22 laser beams. Mapping results are presented in Figure 3. The GP based mapping reconstructed fine details of a relatively large scenario using just few laser measurements. Localization was performed using 1 degree resolution laser sensor. Numerical results for localization are presented in Table 1. Note that GPOM-PFL delivered lower ATE and orientation errors.

As the Bicocca dataset is the largest environments, GP calculation becomes very time consuming. To address this, a mixture of 10 GP experts was used to reduce the computational cost. After information theoretic compression, the mapping dataset is formed by 796 poses and a total of 8327 laser beams. The obtained OGM and GPOM are presented in Figure 4.

The localization result for the Bicocca dataset is presented in Table 1. It is possible to notice a much lower ATE and orientation errors for GPOM-PFL. Figure 4 (c–d) illustrate the estimated pose for each approach. Differently than OGM-PFL, GPOM-PFL resulted in poses that more closely matched the ground truth along all trajectory.



(a) OGM (0.10 cm resolution)

(b) GPOM

(c) OGM-PFL

(d) GPOM-PFL

Fig. 4. (a–b) OGM and GPOM generated from Rawseeds' Biccoca dataset. (c–d) Localization results of OGM-PFL and GPOM-PFL using the bicocca dataset.

6 Conclusion and Future Works

We proposed the combination of GPOM with PFL to improve localization in situations where the data is sparse or when there are occlusions. The advantage of GPOM is the possibility to predict the occupancy at any position in space from a set of sparse measurements. To make GPOM work together with PFL, we modeled a likelihood function based on multivariate normal probability density function that uses the occupancy mean and variance, and the range information of each measurement. With this, all the information retrieved from the map can be combined into a single model by considering dependence between all data points.

We run experiments in two datasets: one obtained from a simulated environment; and three obtained from real world environments. GPOM-PFL results were compared with the standard version of PFL. In all experiments the GPOM-PFL obtained ATE and orientation errors at least 65% more accurate than conventional PFL. From these results we can say that GPOM-PFL can handle noisy sensor data, does not need dense sensor measurements or high-frequency odometer data. It also showed the possibility to provide an accurate pose estimation while traversing areas with less sensor information in the map. GPOM-PFL demonstrated to be a promising localization solution compared to approaches based on discrete maps.

Despite the favorable results of GPOM-PFL, experiments in other datasets are still required, such as in dynamic environments and outdoor environments. The GP method is also known to be computationally expensive due to the $O(n^3)$ cost for inverting a matrix. Solutions for reducing the computational time must be explored, and understanding the trade-off between speed and accuracy of localization is an interesting venue for future work.

7 Acknowledgments

The authors acknowledge the grant provided by FAPESP (2012/02354-1; 2014/09096-3), the ACFR and LRM groups for their support.

References

1. Brooks, A., Makarenko, A., Upcroft, B.: Gaussian process models for indoor and outdoor sensor-centric robot localization. Robotics, IEEE Transactions on 24(6), 1341–1351 (Dec 2008)
2. Ferris, B., Fox, D., Lawrence, N.: Wifi-slam using gaussian process latent variable models. In: Proc. of the 20th International Joint Conference on Artifical Intelligence. pp. 2480–2485. IJCAI'07, San Francisco, CA, USA (2007)
3. Kim, S., Kim, J.: Building occupancy maps with a mixture of gaussian processes. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on. pp. 4756–4761 (May 2012)
4. Ko, J., Fox, D.: Gp-bayesfilters: Bayesian filtering using gaussian process prediction and observation models. In: Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on. pp. 3471–3476 (Sept 2008)
5. Kretzschmar, H., Stachniss, C.: Information-theoretic compression of pose graphs for laser-based slam. The International Journal of Robotics Research (2012)
6. O'Callaghan, S.T., Ramos, F.T.: Gaussian process occupancy maps. I. J. Robotic Res. 31(1), 42–62 (2012)
7. Plagemann, C., Kersting, K., Pfaff, P., Burgard, W.: Gaussian beam processes: A nonparametric bayesian measurement model for range finders. In: In Proc. of Robotics: Science and Systems (RSS (2007)
8. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning). The MIT Press (2005)
9. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of rgb-d slam systems. In: Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on. pp. 573–580 (Oct 2012)
10. Yang, S.W., Wang, C.C.: Feasibility grids for localization and mapping in crowded urban scenes. In: Robotics and Automation (ICRA), 2011 IEEE International Conference on. pp. 2322–2328 (May 2011)