Gray-box probabilistic occupancy mapping

Anthony Tompkins[*]
The University of Sydney

Ransalu Senanayake[*]
The University of Sydney

Fabio Ramos
The University of Sydney

Abstract

In order to deploy robots in previously unseen and unstructured environments, the robots should have the capacity to learn on their own and adapt to the changes in the environments. To this end, leveraging the latest developments in automatic machine learning (AutoML) and probabilistic programming, under the Hilbert mapping framework which can represent the occupancy of the environment as a continuous function of locations, we formulate a Bayesian framework to learn all parameters of the map. Crucially, this way, the robot is capable of learning the optimal shapes and placement of the kernels in Hilbert maps by merely embedding high-level human knowledge of the problem by means of prior probability distributions. Experiments conducted on simulated and real-world datasets demonstrate the importance of incorporating prior information.

1 Introduction

Modeling the environment a robot operates in is fundamental to safer decision-making such as path planning. To this end, discerning occupied areas from unoccupied areas of the environment using depth measurements is required. Typically, occupancy states exhibit highly nonlinear and spatially correlated patterns that cannot be captured with a simple linear classification model. Furthermore, because it is required to learn the occupancy level using very few sparse sensor measurements in a reasonable time, kernel methods have been the *de jure* choice in recent occupancy mapping [2, 3].

One of the major challenges in employing kernel methods in occupancy mapping is the requirement of choosing parameters and hyperparameters of the model [4]. In order for mobile robots to maneuver fully autonomously in unknown environments or to interact with humans and other agents, the robots should have the capability to automatically learn their model parameters from data. Only the most simple environments contain spatially homogeneous features, however this is typically not the case in real-world mapping - e.g. walls and furniture may contribute to sharp features while open spaces and large hills may contribute to spatially smooth features. To better understand the significance of representing nonstationarity in terms of kernels, first consider the SE (squared-exponential) kernel which is parameterized with lengthscale and position hyperparameters. As seen in Figure 1, with large lengthscales it is possible to capture smoother changes across the space, while small lengthscales allow one to capture sharp changes in the space. Hyperparameter optimization is critical for almost all machine learning methods and the best values are almost always dependent on the dataset. Often, a single best lengthscale is chosen that performs, on average, the best for the entire dataset. In our contribution, we determine where to place kernels and what lengthscales they should have.

Another important aspect that should be taken into account when designing robot models is the uncertainty inherent to all levels of robots—from sensor and actuator imperfections to model misspecifications. Another incentive to use Bayesian models is that they provide an interface to incorporate high-level human knowledge about the system into the model through prior probability distributions.

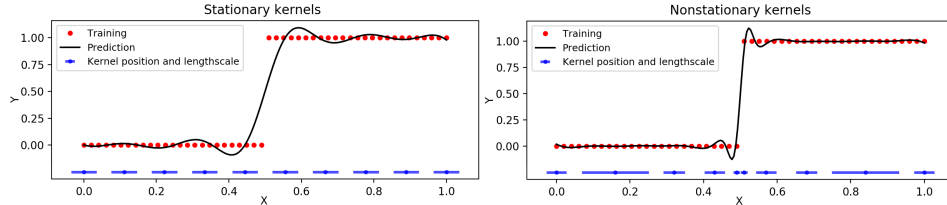[*]Equal contribution. This paper is an based on [1].

Figure 1: Comparison of stationary and nonstationary kernels, $\exp(-\|\mathbf{x} - \tilde{\mathbf{x}}\|_2^2/2l^2)$ with bivariate Gaussian distributions $\tilde{\mathbf{x}}$ hinged on the environment and lengthscales l, and their ability to represent sharp spatial changes. Note that both examples have the same number of kernels, however in the non-stationary case the kernels have different positions and lengthscales to account for abrupt changes in the training data.
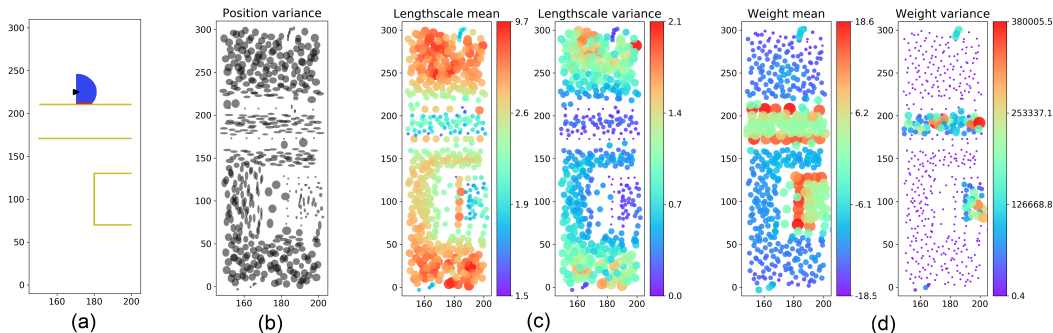

Figure 2: Learned kernel and model parameter distribution means and variances. (a) A portion of the environment (b) Positions of hinge kernels $\tilde{\mathbf{x}}$ (c) Lengthscales (d) Weights

Such an approach where data, alongside prior knowledge or structure, is injected into the model is known as *gray-box modeling* [5, 6].

In this paper, we use stochastic gradient descent with the reparameterization trick [7] to solve a challenging learning problem of automatically determining all parameters for Hilbert maps - which have traditionally used human-designed kernel hyperparameters. We demonstrate the importance of using more involved Bayesian formulations for uncertainty representation and learning thousands of parameters (Figure 2) in both small and big data settings without laborious mathematical derivations.
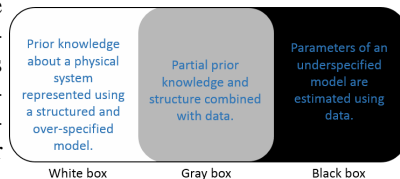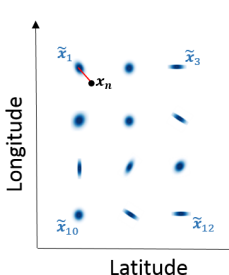

Figure 3: Gray box modeling

We are interested in making use of the physical knowledge about the environment to build a robust map. As illustrated in Figure 3 gray box models combine partial prior knowledge and structure with data [5, 8]. Unlike in black box models where model equations and parameters have no physical meaning, the proposed gray box occupancy mapping model is interpretable. On the other hand, akin to a white box model, we make use of kernels as similarity functions to add structure into the model to specify "nearby points in the space should have similar occupancy values." However, how strong these similarities at a specific location (i.e. nonstationarity) are learned from data. We inject further prior knowledge of what the parameter values should be through prior probability distributions.

2 Bayesian Hilbert maps

With the advancement of depth sensors such as lidar and sonar, occupancy grid maps (OGM) developed in 1980s [9] became a popular choice for representing the environment. To alleviate the disadvantages of OGMs, Gaussian process occupancy maps (GPOMs) [10] were proposed. Eliminating the cubic run-time complexity in GPOMs, Hilbert maps (HMs) [2] and Bayesian Hilbert maps (BHMs) were proposed [11]. In BHMs, the map is learned on a reproducing kernel Hilbert space (RKHS) where kernel functions are used to characterize spatial relationships. A kernel $k(\mathbf{x}, \tilde{\mathbf{x}}) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a function that measures the similarity between two mul-

tidimensional inputs $\mathbf{x}, \tilde{\mathbf{x}} \in \mathcal{X} \subset \mathbb{R}^2$. The pairwise similarities between the elements of the two sets of points $\{\mathbf{x}_n \in \mathbb{R}^2\}_{n=1}^N$ and $\{\tilde{\mathbf{x}}_m \in \mathbb{R}^2\}_{m=1}^M$ are computed. Here, $\mathbf{x}$ are longitude-latitude locations of either free or occupied $y \in \{0, 1\} = \{\text{free}, \text{occupied}\}$ data points sampled from lidar beams and $\tilde{\mathbf{x}}$ are points *hinged* on pre-defined locations of the space. A SE kernel $k(\mathbf{x}_n, \tilde{\mathbf{x}}_m; l) = \exp\left(-\|\mathbf{x}_n - \tilde{\mathbf{x}}_m\|_2^2 / 2l^2\right)$ with a heuristically determined lengthscale l is used to compute the the feature vector $\phi(\mathbf{x}_n; l) = (k(\mathbf{x}_n, \tilde{\mathbf{x}}_1; l), k(\mathbf{x}_n, \tilde{\mathbf{x}}_2; l), ..., k(\mathbf{x}_n, \tilde{\mathbf{x}}_M; l)) \in \mathbb{R}^M$ for all data points $\{\mathbf{x}_n\}_{n=1}^N$. In this sense, $\{(\mathbf{x}_n, y_n)\}_{n=1}^N$ is the dataset and $\{l, \{\tilde{\mathbf{x}}_m\}_{m=1}^M\}$ is the pre-defined parameter set. Once the feature vector is computed, it passes through a sigmoidal function to estimate the occupancy level $\hat{y} = p(y|\mathbf{x}_*, \mathbf{w}) = 1/(1 + \exp(\mathbf{w}^\top \phi(\mathbf{x}_n; l)))$ of a query point in the space $\mathbf{x}_*$, given the weights $\mathbf{w} \sim \mathcal{N}$. As this query point can be any longitude-latitude pair, as opposed to OGMs, BHMs can produce maps with arbitrary resolution at prediction time. In BHMs, the lengthscales of the kernel l and where to place them $\tilde{\mathbf{x}}$ are prefixed values.

3 Nonstationary kernels for Hilbert mapping

Figure 4: Feature vector computation. $\{\tilde{\mathbf{x}}\}_{m=1}^{M=12}$ are hinge distributions and $\mathbf{x}_n$ is the nth data point.

Table 1: Description of model parameters. Assuming independence, individual distributions are associated with all hinge points $m = 1...M$

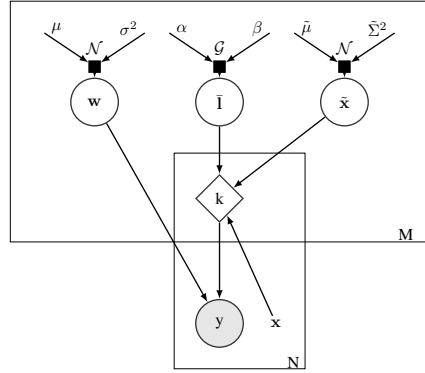| Var.	Distributions
Prior distributions:	
w_m	$\mathcal{N}(\mu_m, \sigma_m^2)$
$\bar{l}_m$	$\mathcal{G}(\alpha_m, \beta_m)$ [2]
$\tilde{\mathbf{x}}_m$	$\mathcal{N}(\tilde{\mu}, \tilde{\Sigma}^2)$
Variational distributions:	
qw_m	$\mathcal{N}(\mu_m^{(q)}, \sigma_m^{2(q)})$
$q\bar{l}_m$	$\mathcal{LN}(\alpha_m^{(q)}, \beta_m^{(q)})$ [3]
$q\tilde{\mathbf{x}}_m$	$\mathcal{N}(\tilde{\mu}^{(q)}, \tilde{\sigma}^{2(q)})$

Figure 5: The graphical model. k represents the kernel which is evaluated $N \times M$ times.

Learning hyperparameters of the map is crucial for driving the robot fully autonomously. In this section, we propose novel techniques for mapping unstructured environments without a human explicitly providing hyper-parameters. As the main contribution of this paper, we propose a principled approach to learn weights, lengthscales, and positions of kernels. Individual lengthscales $\{l_m\}_{m=1}^M$ essentially model the nonstationary behavior and can easily acclimatize to local changes in the environment. To this end, by using the ideas of gray-box modeling, we start with possible locations for kernels as bivariate Gaussians and inverse length-scales as Gamma distributions, and then optimize them using data.

Since observed occupancy values are always binary and are independent of each other, we assume the likelihood follows a Bernoulli distribution $p(\mathbf{y}|\mathbf{x}, \mathbf{w}, \mathbf{l}, \tilde{\mathbf{x}})$ where $\log(\theta/(1 - \theta))) = \mathbf{w}^\top \Phi(\mathbf{x}; \mathbf{l}, \tilde{\mathbf{x}})$. As shown in Figure 4, kernel functions are now implicitly evaluated between datapoints point and hinge distributions, naturally accounting for uncertainty. Our objective is to learn the posterior distribution. However, because of the Bernoulli likelihood, the posterior is intractable and hence is approximated using another distribution q. With the variables defined in Table 1, indicating longitude and latitude with lon and lat, respectively, the basic formulation with mean-field variational approximation is given in Figure 4 and the following equation,

$$\underbrace{\prod_{m=1}^M q(w_m)q(\mathbf{l}_m)q(\tilde{x}_m^{\text{lon}})q(\tilde{x}_m^{\text{lat}})}_{\text{factorized variational distribution}} = \underbrace{q(\mathbf{w}, \mathbf{l}, \tilde{\mathbf{x}})}_{\substack{\text{variational} \\ \text{distribution}}} \approx \underbrace{p(\mathbf{w}, \mathbf{l}, \tilde{\mathbf{x}}|\mathbf{x}, \mathbf{y})}_{\text{posterior}} \propto \underbrace{p(\mathbf{w})p(\mathbf{l})p(\tilde{\mathbf{x}})}_{\text{priors}} \underbrace{p(\mathbf{y}|\mathbf{x}, \mathbf{w}, \mathbf{l}, \tilde{\mathbf{x}})}_{\text{likelihood}}.$$

3

4 Experiments and discussions

Experiments were conducted on a simulated dataset (600×300 m^2 area) and Intel lab dataset implemented using the Edward library. On average it takes around 10 minutes to learn upwards of 57,600 parameters (8 parameters per kernel with more than 7200 kernels) and 300,000 data points using a computer with a GTX1080 Ti 11 GB. More details can be found on [1] and the code is available online: `https://github.com/MushroomHunting/automorphing-kernels`.

Learning lengthscales and kernel locations are assessed in this experiment. A learned environment is shown in Figure 6. To understand the full effect of the proposed model it is not enough to look at the predicted occupancy map—we must consider the underlying distributions. Figure 2 provides a visual map of the means and variances of a learned model's predictive distributions. Accounting for a large part of the upper and lower parts of the map, the position variance in Figure 2b shows that in areas of dense laser scans where no walls exist, a larger but uniform variance for each spatial dimension is learned. For the areas where the laser scanner has detected walls one observes a stark contrast exhibited by the smaller spatial variances. In the walled area spanning the middle of the map the learned variances in the latitudinal direction are stretched out further relative to the longitudinal direction reflecting the narrow corridor-like shape of the wall. Concerning now the lengthscale mean and variance in Figure 2c we can observe the most significant effect in terms of the learned posteriors. At the top and the bottom open areas the largest lengthscales are observed signifying a minimal complexity of occupancy. Paralleling the learned position variances, the learned lengthscale means are clustered around either areas of detail or areas of uncertain occupancy. This effect is repeated in the lengthscale variance. The kernel weights means and variances are depicted in Figure 2d where one can see the highest weights appear around areas associated with the smallest position and lengthscale variances. Contrastingly, the most negative weights appear in regions of highly confident predicted empty occupancy. The weights closest to zero occur in areas of the map the robot has no visual perception and these constitute the insides of walls. The effect of the weight means is reflected in the weight variance where areas of high observability, which include open spaces and walls, have a low uncertainty in their estimates. Areas of low observability, i.e. inner parts of walls, have extremely high variances. This underlying analysis of the learned posterior distributions not only substantiates the motivation for spatially adaptive kernel learning, but also gives an explainable and intuitive understanding of what the model has learned which is often critically important for robotic tasks that interact with real-world environments.

The accuracy metrics are reported in Table 2, variational sparse dynamic Gaussian process occupancy maps (VSDGPOM) [11], HMs, and BHMs. The best lengthscales for previous Hilbert mapping techniques were determined using five-fold cross-validation. The proposed approach, Automorphing BHMs (ABHMs) outperform in both datasets because it models nonstationarity and can adjust the position and shape of kernels to locally adapt to the environment.

Prior distributions in Bayesian inference is one of the principled methods to embed domain knowledge into data-driven models. In this paper, we demonstrated how to enrich an occupancy map through such distributions and further optimize model parameters as a robot collects more data. In a broader sense, in both robot perception and control in dynamic environments, it is possible to define prior probability distributions on parameters of robot models based on domain knowledge. Such prior probabilities can have hierarchies that can be structured by Bayesian hierarchical models [12, 13] and efficiently inferred with the use of probabilistic programming tools [14]. We are interested in using the concepts of gray-box modeling that have historically been used in systems identification for making informed decisions by using both physical knowledge about the problem and data collected by interacting with the environment.
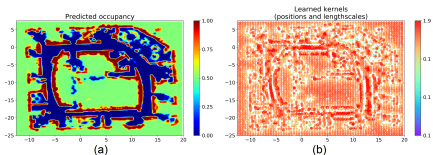

Figure 6: Intel dataset. (a) Predicted occupancy map. (b) learned lengthscales. Note the change in kernel lengthscale with respect to the spatial compactness of the walled areas.

Dataset	Simulation		Intel lab	
	AUC	MNLL	AUC	MNLL
ABHM	**0.999**	**0.015**	**0.994**	**0.093**
BHM	**1.000**	0.176	0.921	0.362
HM	0.992	0.226	0.938	0.666
VSDGPOM	0.801	0.372	0.794	0.530

Table 2: Losses on all real datasets. The higher the area under curve (AUC) or the lower the mean negative log loss (MNLL), the better the model is.

References

[1] R. Senanayake, A. Tompkins, and F. Ramos, "Automorphing kernels for nonstationarity in mapping unstructured environments," 2018.

[2] F. Ramos and L. Ott, "Hilbert maps: scalable continuous occupancy mapping with stochastic gradient descent," in *Proceedings of Robotics: Science and Systems (RSS)*, (Rome, Italy), July 2015.

[3] K. Doherty, J. Wang, and B. Englot, "Probabilistic map fusion for fast, incremental occupancy mapping with 3d hilbert maps," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.

[4] R. Senanayake and F. Ramos, "Bayesian hilbert maps for dynamic continuous occupancy mapping," in *Conference on Robot Learning*, pp. 458–471, 2017.

[5] N. R. Kristensen, H. Madsen, and S. B. Jørgensen, "A method for systematic improvement of stochastic grey-box models," *Computers & chemical engineering*, vol. 28, no. 8, pp. 1431–1449, 2004.

[6] P. Galliani, A. Dezfouli, E. Bonilla, and N. Quadrianto, "Gray-box inference for structured gaussian process models," in *Artificial Intelligence and Statistics*, pp. 353–361, 2017.

[7] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[8] H. Madsen, "Lecture notes: Statistical modelling of physical systems an introduction to grey box modelling," February 2013.

[9] A. Elfes, *Occupancy grids: a probabilistic framework for robot perception and navigation*. PhD thesis, Carnegie Mellon University, 1989.

[10] S. T. O'Callaghan and F. T. Ramos, "Gaussian process occupancy maps," *The International Journal of Robotics Research (IJRR)*, vol. 31, no. 1, pp. 42–62, 2012.

[11] R. Senanayake, S. O'Callaghan, and F. Ramos, "Learning highly dynamic environments with stochastic variational inference," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[12] C. H. Schmid and E. N. Brown, "Bayesian hierarchical models," *Essential Numerical Computer Methods*, p. 199, 2000.

[13] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2, pp. 524–531, IEEE, 2005.

[14] D. Tran, A. Kucukelbir, A. B. Dieng, M. Rudolph, D. Liang, and D. M. Blei, "Edward: A library for probabilistic modeling, inference, and criticism," *arXiv preprint arXiv:1610.09787*, 2016.