# Continuous Occupancy Mapping with Integral Kernels

**Simon T. O'Callaghan** and **Fabio T. Ramos**
Australian Centre for Field Robotics
School of Information Technologies
The University of Sydney, New South Wales, 2006 Australia
{s.ocallaghan, f.ramos}@acfr.usyd.edu.au

**Abstract**

We address the problem of building a continuous occupancy representation of the environment with ranging sensors. Observations from such sensors provide two types of information: a line segment or a beam indicating no returns along them (free-space); a point or return at the end of the segment representing an occupied surface. To model these two types of observations in a principled statistical manner, we propose a novel methodology based on integral kernels. We show that integral kernels can be directly incorporated into a Gaussian process classification (GPC) framework to provide a continuous non-parametric Bayesian estimation of occupancy. Directly handling line segment and point observations avoids the need to discretise segments into points, reducing the computational cost of GPC inference and learning. We present experiments on 2D and 3D datasets demonstrating the benefits of the approach.

## 1 Introduction

One of the most popular representations of the environment for both mobile robotics and robotic manipulation is occupancy maps. Originally proposed as a discretised set of cells, occupancy grid maps (Moravec and Elfes 1985) have been extensively used over the past 25 years. They offer a number of benefits for navigation and planning tasks: first, the environment is divided between occupied and free-space regions which makes path planning easier to be performed; second, they are easy to implement and have constant memory requirements as long as size of the map remains the same; third, they provide a probabilistic measure of occupancy for each cell which is important for robustness in subsequent tasks. Despite these benefits, occupancy grids rely on very strong assumptions of the environment to be efficient. Notably, the assumption that the likelihood of occupancy in one cell is independent of other cells disregard spatial correlations that can be important to infer occupancy in unobserved nearby regions. Additionally, traditional occupancy grids require that the discretisation of the environment be defined *a priori* which makes the spatial resolution constant throughout the map.



To overcome these two major limitations, we propose to represent the environment as a continuous spatial classification problem. For any point in 2D or 3D spaces we develop a classifier that outputs the likelihood of occupancy while modelling spatial dependencies in a principled statistical manner. Most (if not all) of machine learning supervised classification methods assume point inputs in some $\Re^D$ space and the class label as output. However, observations from ranging sensors such as laser scanners contain beams or line segments representing unoccupied space, and point observations generated by laser returns indicating an occupied surface. To model these two types of observations jointly, we propose a new set of classifiers based on the idea of integral kernels. An integral kernel is a positive definite kernel (Mercer kernel) that defines a metric between line segments and points in some high-dimensional feature space. It allows the incorporation of line segments as observations into kernel-based classifiers. For the purpose of building continuous occupancy maps, we use Gaussian process (GP) classifiers as they allow the estimation of kernel parameters by optimisation of a Bayesian quantity, the marginal likelihood.

Specifically, this paper brings the following contributions: 1) A new method for continuous occupancy maps using integral kernels; 2) An efficient adaptive quadrature procedure to approximate integral kernels with no analytic solutions; 3) An efficient methodology to update kernel matrices built from integral kernels that preserves numerical stability. We perform experiments in both 2D and 3D environments demonstrating the applicability and benefits of the approach.

This paper is organised as follows. Sec. 2 summarises the related literature. Sec. 3 describes the proposed algorithm while Sec. 4 presents results from simulated and real datasets. Sec. 5 discusses the principal conclusions.

## 2 Related Work

Many attempts have been made to address the problematic issues inherent in the occupancy grid with varying degrees of success. A number of authors have taken advantage of the intrinsic structure in an environment to develop mapping techniques that remove this independence between cells assumption. A heuristic approach described in (Veeck and Burgard 2004) train polylines to form a continuous representation of the environment's boundaries based on discrete range sam-

ples by employing user-defined optimisation criteria. A technique proposed in (Paskin and Thrun 2005) uses polygonal random fields to probabilistically reason about occupancy, rather than the boundaries, of the environment. The maps generated are continuous and allow for inference to be made in unscanned regions. However, this comes at considerable computational cost.

More recently, GPs have become a popular tool for robotic perception, (Smith, Posner, and Newman 2010; Lang, Plagemann, and Burgard 2007), primarily due to their ability to learn spatial correlation with noisy data in a Bayesian setting. (O'Callaghan, Ramos, and Durrant-Whyte 2009) uses Gaussian processes to infer the hypothesis of occupancy at any point in the environment. The technique provides a continuous representation of the world that uses Bayesian inference to estimate the probability of occupancy in occluded or unscanned areas. Associated predictive variance maps indicate the degree of confidence the model has in its estimates. (Bohg et al. 2010; Gan, Yang, and Sukkarieh 2009) uses these outputs to determine optimised exploration trajectories that maximise information gained about the robot's surrounds.

This paper takes a different approach to continuous occupancy maps by proposing novel kernels capable of providing positive-semi definite matrices computed between line and point observations. This has a number of benefits over previous methods including efficiency and accuracy. We present a detailed description of the entire occupancy map algorithm as follows.

## 3 Algorithm Description

A general overview of the proposed method is illustrated in Fig. 1. Essentially, we treat the occupancy map as a form of classification problem. The robotic platform makes range observations of the real world environment. The resulting occupied points and free-space line segments are actively sampled (Section 3.4) and, if necessary, stored in a cellular data structure (Section 3.3). A Gaussian process is used to perform an initial regression on processed sensor data while a probabilistic least-squares classification algorithm is then trained to identify regions of occupancy and free-space based on the GP's outputs (Section 3.1). The GP framework is extended to handle observations along line segments rather than just deterministic points (Section 3.2).

### 3.1 Gaussian Process Occupancy Maps

Gaussian processes occupancy maps, GPOMs, essentially use GPs to classify the environment into regions of occupancy and free-space using rangefinder sensor observations as training data. GPs provide a powerful framework for learning models of spatially correlated and uncertain data. GP regression provides a robust method of prediction and can handle incomplete sensor data effectively. They can be viewed as a Gaussian probability distribution in function space and are characterized by a mean function $\mu(\mathbf{x})$ and the covariance function $k(\mathbf{x}, \mathbf{x}_*)$.

Hence, the process itself can be thought of as a distribution over an infinite number of possible functions and inference takes place directly in the space of functions. By assuming that the target is data jointly Gaussian, we obtain

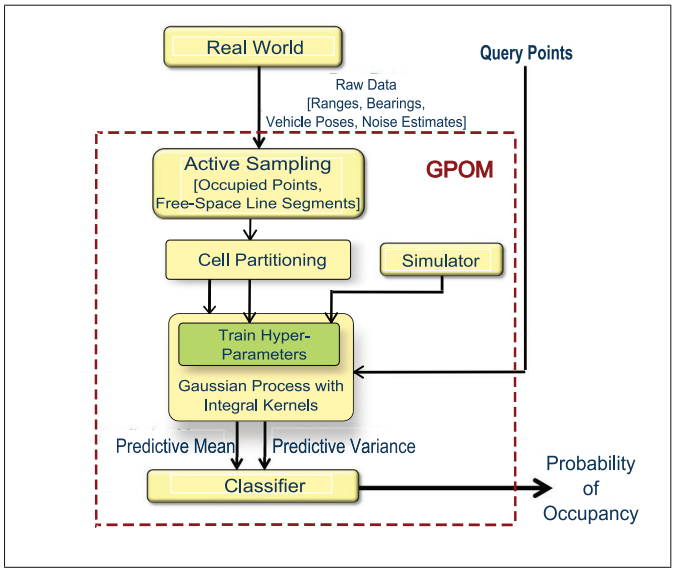


Figure 1: Block diagram of the proposed mapping methodology

$$f(\mathbf{x}_*) = \mathcal{N}(\mu, \sigma), \tag{1}$$

where

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

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

In the case of GP occupancy maps, the training and test data, $\mathbf{x}$ and $\mathbf{x}_*$ respectively, refer to physical locations in space while the target data, $\mathbf{y} \in \{-1, 1\}$, represents the class label (free-space or occupied). $k(\mathbf{x}, \mathbf{x})$ or simply $K$ is the matrix of the covariances evaluated at all pairs of training inputs. Its elements are defined depending on a covariance function $k$ parameterised by hyperparameters $\theta$. $\sigma_n^2$ is the variance of the global noise. A detailed explanation and derivation of the Gaussian process can be found in (Rasmussen and Williams 2006).

Gaussian processes are generally used in regression and so an additional stage is required for classification problems. A probabilistic least-squares classifier, described in (Rasmussen and Williams 2006) and (Platt 2000), uses the GP's predictive mean and variance functions to determine the probability of occupancy for each query point:

$$p(\text{Occupancy}|\mathbf{x}_*) = \Phi\Big(\frac{\alpha\mu(\mathbf{x}_*) + \beta}{1 + \alpha^2\sigma(\mathbf{x}_*)^2}\Big). \tag{4}$$

Here, $\Phi$ is the cumulative Gaussian function while $\alpha$ and $\beta$ are parameters that are optimised by performing leave-one-out cross-validation on the training set. Using the resulting distribution, the environment can be classified into occupied, free-space and unsure regions using user-defined thresholds that depend on the desired level of greediness.

### 3.2 Integral Kernels

The conventional Gaussian process occupancy map described in (O'Callaghan, Ramos, and Durrant-Whyte 2009) has some drawbacks, mainly due to the manner in which the free-space observation is represented in the model. Extracting the closest points on nearby line segments to the test point results in unique training data and consequently unique $K$ matrices for each query. Hence a new matrix must be inverted to evaluate Eq. 2 and 3 each time $P(\text{Occupancy}|x_*)$ is required.

Additionally, it is necessary to discretise the line segment into multiple free-space points during training to approximate a continuous sensor beam. This can inflate the size of the training set to impractical levels and slow the optimization of the hyperparameters considerably due to the $\boldsymbol{O}(n^3)$ nature of the algorithm.

Both of these issues can be solved by extending the Gaussian process framework to model the continuous sensor beam directly as a path integral over a line segment. A method for handling integral observations using linear operators is proposed in (Murray-Smith and Pearlmutter 2005; Osborne 2010) which discretises the line segment in a similar manner as the trapezoidal rule. This approach greatly increases the size of the training set. Consequently we propose a novel approach in which we redefine the covariance function itself to handle a line segment as a single continuous observation.

**Definition 1. (Integral Kernel)** *Let $l_1 : [a, b] \to C_1$ and $l_2 : [c, d] \to C_2$ be two arbitrary bijective parametrizations of curves $C_1$ and $C_2$ respectively such that $l_1(a)$ and $l_1(b)$ give the endpoints of $C_1$ and $l_2(c)$ and $l_2(d)$ give the endpoints of $C_2$. Let $k(\cdot, \cdot)$ be a symmetric positive semi-definite function (Mercer kernel) that maps $k : [e, f] \times [e, f] \to \Re$. We define the integral kernel $k_{II}$ of $k$ with respect to $l_1$ and $l_2$ as*

$$k_{II}(l_1, l_2) = \int_a^b \int_c^d k(l_1(u), l_2(v))dudv, \tag{5}$$

*which exists and is finite.*

Similarly, an integral kernel can be defined between a parameterised curve $l(u) : [a, b] \to C$ and a point $x$ as

$$k_I(l, x) = \int_a^b k(l(u), x)du, \tag{6}$$

where $l(a)$ and $l(b)$ are the endpoints of the curve $C$. It follows from the definition of positive semi-definite kernels that the kernel matrix computed as $\begin{bmatrix} K_{II} & K_I \\ K_I^T & K \end{bmatrix}$ is also positive semi-definite. [1]

For the purpose of continuous occupancy maps, the curves $C$ are line segments. For example, the $2D$ line segment from $(a, b)$ to $(c, d)$ is defined as $\mathbf{l}(u) = (x(u), y(u)) = (a + u(c - a), b + u(d - b)), u \in [0, 1]$.

[1] proof omitted due to space constraints.

**Integral squared exponential.** Consider the integral kernel for the commonly used covariance function, the squared exponential:

$$k(x, x') = \sigma_f^2 \exp\left( - \frac{(x - x')^2}{2l^2} \right) \tag{7}$$

where $\sigma_f$ and $l$ are the hyperparameters representing the signal variance and length-scale, respectively. The closed-form solution for the squared exponential covariance function applied to Eq. 6 can be determined for the 1-D case,

$$k_I(l(u), x) = \sigma_f^2 l \sqrt{\frac{\pi}{2}} \left| \text{erf}\Big(\frac{a - x}{\sqrt{2}l}\Big) - \text{erf}\Big(\frac{b - x}{\sqrt{2}l}\Big) \right|. \tag{8}$$

A closed-form solution for the $n$-dimensional case can be obtained similarly.

**Adaptive Order Quadrature.** While an analytical solution exists for Eq. 6 in the squared exponential case, numerical integration is required to determine the solution for Eq. 5. In the proposed implementation, the Clenshaw-Curtis quadrature, (Gentleman 1972), is employed primarily because of its fast-converging accuracy compared to Gaussian quadrature rules and its applicability to numerous families of covariance functions. We propose an adaptive version of the quadrature whereby the order, $o$, or number of sample points adjusts to suit length of the line segment as well as the covariance function's length-scale. This approach ensures an efficient sampling technique that reduces the risk of producing non-positive definite matrices while also decreasing the overall number of sample points.

This quadrature approximates an integral by a weighted sum as:

$$\int f(u)du \approx \sum_{i=1}^{o} w_i f(u_i), \tag{9}$$

where $w$ and $u_i$ are respectively the weights and locations of the sample points specified in (Gentleman 1972). The order of the quadrature for each line is determined by temporarily warping the lengths of the segments so as to normalise the length-scale hyperparameters, $l$. $o$ is then assigned based on the length of each warped segment. Consequently, long lines or models with short length-scales are automatically sampled more densely compared to shorter lines or smoother functions.

The quadrature can equally be used to evaluate the solution to Eq. 6 for other covariance functions such as the Matèrn family (Stein 1999).

**Target Vector Behaviour** For regression with integral kernels, the target vector $\mathbf{y}$ represents the net area under the function $f(*)$, along each line segment. Extending this to classification, the target for each line observation becomes:

$$\mathbf{y}_i = \text{Class}_i * |\overline{ab}|_i, \tag{10}$$

where $\text{Class}_i$ is the class label (-1 or +1) of the $i^{th}$ observation and $\overline{ab}$ is the line segment. Figure 2 illustrates the application of integral kernels to a simulated regression problem. The ground truth is noisily sampled using both line and point observations.

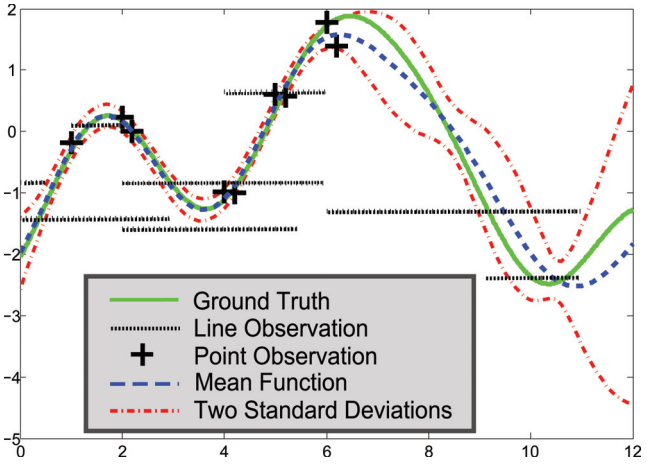


Figure 2: 1-D example. The dataset consists of both line and point observations.

### 3.3 Updating the Covariance Matrix and Limiting the Size of $K$

A key benefit to our proposed approach is the ability to store a covariance matrix, $K$, that is applicable to multiple query points. Consequently, $K^{-1}$ need only be computed once thus eliminating the $\boldsymbol{O}(n^3)$ computational bottleneck of the algorithm proposed in (O'Callaghan, Ramos, and Durrant-Whyte 2009). During online operation, however, it is necessary to perform iterative updates of $K$ to incorporate additional observations into the model. The matrix inversion lemma is usually applied in these cases but it can damage the numerical stability of the inverted matrix. We therefore resort to a more stable procedure based on Cholesky factor updates (Osborne et al. 2008). Consider the positive definite matrix, $[K_{1,1}]$ with Cholesky decomposition $[R_{1,1}]$. Given a new positive definite matrix $\begin{bmatrix} K_{1,1} & K_{1,2} \\ K_{2,1} & K_{2,2} \end{bmatrix}$ Its Cholesky decomposition, $\begin{bmatrix} S_{1,1} & S_{1,2} \\ 0 & S_{2,2} \end{bmatrix}$ can be updated as:

$$S_{1,1} = R_{1,1} \tag{11}$$

$$S_{1,2} = R_{1,1}^T \setminus K_{1,2} \tag{12}$$

$$S_{2,2} = \text{chol}(K_{2,2} - S_{1,2}^T S_{1,2}). \tag{13}$$

This reduces the computational complexity of an update from $\boldsymbol{O}(n^3)$ to $\boldsymbol{O}(n^2)$. Inevitably, even this more efficient implementation becomes impractical as more observations are added and the covariance matrix grows to an intractable size. A number of solutions have been suggested to address this scalability issue such as Bayesian Committee (Tresp 2000). Here, we split the covariance matrix once it reaches an unacceptable size and begin to grow a new model with the incoming observations. It is important to note that we are not discretising the *map* like what is done in occupancy grids but instead we are partitioning the *data* into more manageable cells.

### 3.4 Active Sampling

The frequency of required updates to the model can also be greatly reduced by filtering the incoming observations. Essentially, the information gained from adding each new training point is determined using an active sampling technique. The Kullback-Leibler divergence is used as a metric to evaluate the benefit of including a new observation by comparing the mean and variance at a location before and after the training point is added. Using a threshold specified *a priori*, points below a certain Kullback-Leibler divergence are rejected. Consequently, the sampling rate is naturally adjusted by environment complexity. This threshold can be tuned to balance speed with accuracy depending on the intended application. A similar active sampling technique is described in (Smith, Posner, and Newman 2010).

## 4 Experimental Results

### 4.1 Simulated Data

Initial tests were carried out using synthetic datasets which provided known ground truths. Fig. 3 illustrates the results from a 2-D experiment in which two robots navigate a room taking range observations. Over 26 scans, a total of 249 laser returns were logged. The layout of the environment as well as the occupied points and free-space line segments used as training data are shown in Fig. 3(a). The resulting occupancy map and associated predictive variance produced using GPs with Integral Kernels, GPOMIK, are presented in Fig. 3(b) & (c), respectively.

Using Bayesian inferences the mapping technique produces a continuous underlying function representing the probability of occupancy for the entire region. After coarsely sampling this function at a resolution of 0.5 x 0.5 m, it is can be seen that the map bears a close resemblance to the ground truth. Walls and obstacles are assigned a high probability of occupancy while the free-space regions of the environment are deemed unlikely to be occupied despite significant distances between observations in areas. Predictions made in regions devoid of any measurements (such as the lower left corner) are accompanied by a rise in variance.

The availability of a ground truth makes it possible compare various mapping techniques. Fig. 3(d) shows the Receiver Operator Characteristic or ROC curves for the proposed GPOMIK approach as well as the previous Gaussian Process Occupancy Map approach, GPOM, described in (O'Callaghan, Ramos, and Durrant-Whyte 2009), and the occupancy grid. The independence-of-cells assumption made by the occupancy grid results in large areas of the map remaining unaltered from the prior and explains why it under-performs compared to the Bayesian approaches. By representing the free-space beams as true line segments in the model, our approach benefits from having more representative training data which leads to more suitable hyper-parameters. Additionally, all query points share the same training data so $K$ is only calculated once and contains information from all observations as opposed to the k-nearest neighbour strategy used by the GPOM. The comparisons are quantified in Table 4.1.

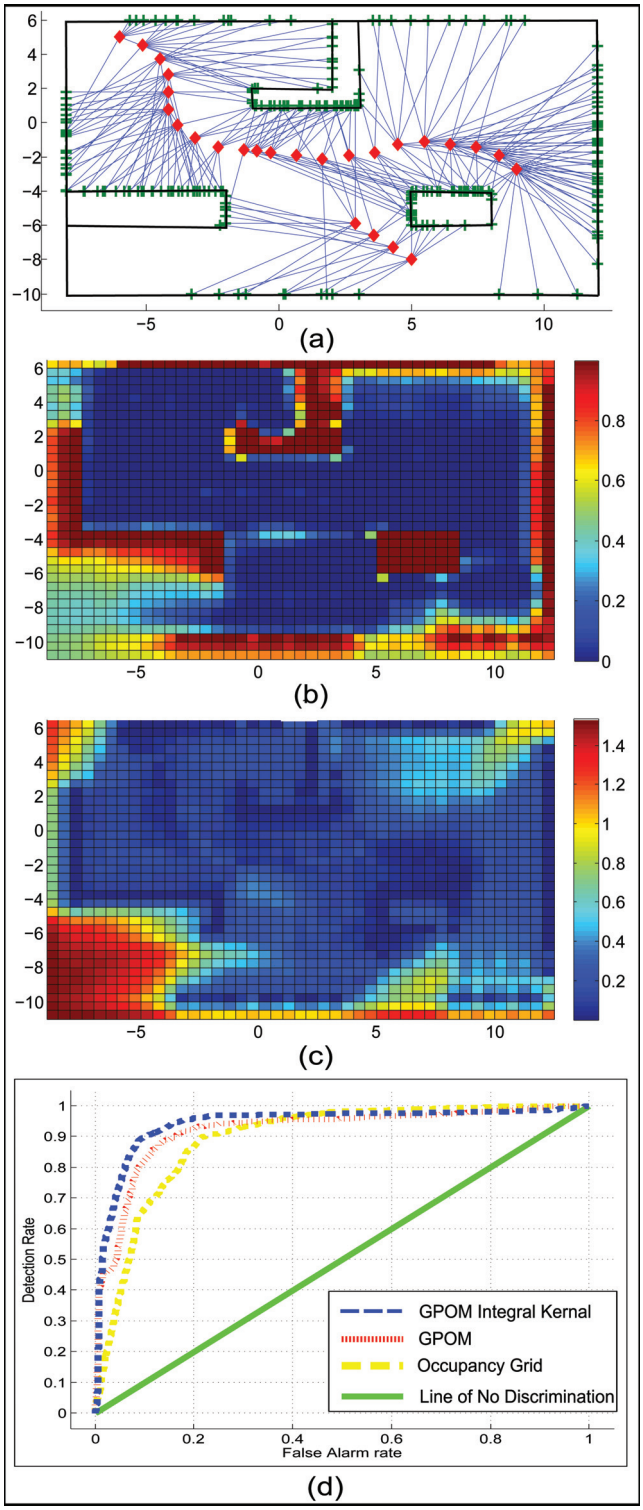


Figure 3: 2-D Simulated Dataset.(a) Plan view of room. Robot poses are shown as red diamonds. Observed occupied points and free-space line segments are represented by green crosses and blue lines, respectively. (b) Probability of occupancy versus location. (c) Predictive variance map.(d) ROC curve comparing the performance of the proposed GPOMIK method against the algorithm described in (O'Callaghan, Ramos, and Durrant-Whyte 2009) and the occupancy grid on a simulated dataset with known ground truth.

Table 1: Quantitative comparison of experimental results

| | Area under the curve | False positive rate when True positive rate = 0.90 |
|---|---|---|
| GPOMIK | 0.9441 | 0.101 |
| Previous GP Method | 0.9162 | 0.7957 |
| Occupancy Grid | 0.8938 | 0.219 |
| No Discrimination | 0.5 | 0.9 |

The theory can also be applied to 3-D datasets as demonstrated in Fig. 4. The top image shows the true state of the environment as well as the locations of laser returns selected as training data by the active sampler. From 1024 observation pairs (laser beam and hit), the active set was reduced to 471 points and 217 line segments. The lower subfigure shows all query points that were classified as occupied and whose variance was below a certain threshold. As GPOMIK models occupancy rather than a surface, representing problematic features such as overhangs is trivial.

Computationally, the GPOMIK iteratively builds and stores a model making the subsequent evaluation faster than the GPOM which must compute a small local model for each query point. In this instance, the model was developed as measurements were received along the trajectory lasting 65 seconds. Fig. 5 compares the times taken for the GPOMIK and the previous method proposed in (O'Callaghan, Ramos, and Durrant-Whyte 2009) to evaluate the model over a range of resolutions. The proposed method requires, on average 6.06 seconds to evaluate 35,000 query points versus the 68.31 seconds needed by the previous approach. In a situation where the global model is not grown incrementally as observations are received but instead has to be calculated from scratch, the GPOMIK becomes computationally more beneficial after approximately 40,000 queries.

## 4.2 Real Data

A Riegl LMS-Z620 lidar was used to gather range measurements from a fixed position on a university green area. 30,000 laser returns were provided as inputs to the algorithm from which the active sampler selected approximately 30% as training data, Fig. 6(a). The maximum size of each covariance matrix, $K$, was limited to increase the overall performance speed, Sec. 3.3, resulting in 291 cells. Once $K$ and its inverse are determined and stored for each cell, querying the probabilistic model is fast. Eq. 2 & 3 now become $\boldsymbol{O}(n^2)$ operations, where $n$ is the number of training points in the relevant cell. Importantly, as the test points are independent of one another, constructing the occupancy map is easily parallelizable.

After querying the model at a resolution of 0.5 x 0.5 x 0.5 m, resulting in 909416 test points and an evaluation time of 63 seconds, the points that were classified as occupied and with a variance below 1 were extracted and plotted in Fig. 6(b & c). RGB data is added to Fig. 6(b) using information from a more detailed scan to aid with the interpretation of the image. Walls, park areas and trees are identifiable in

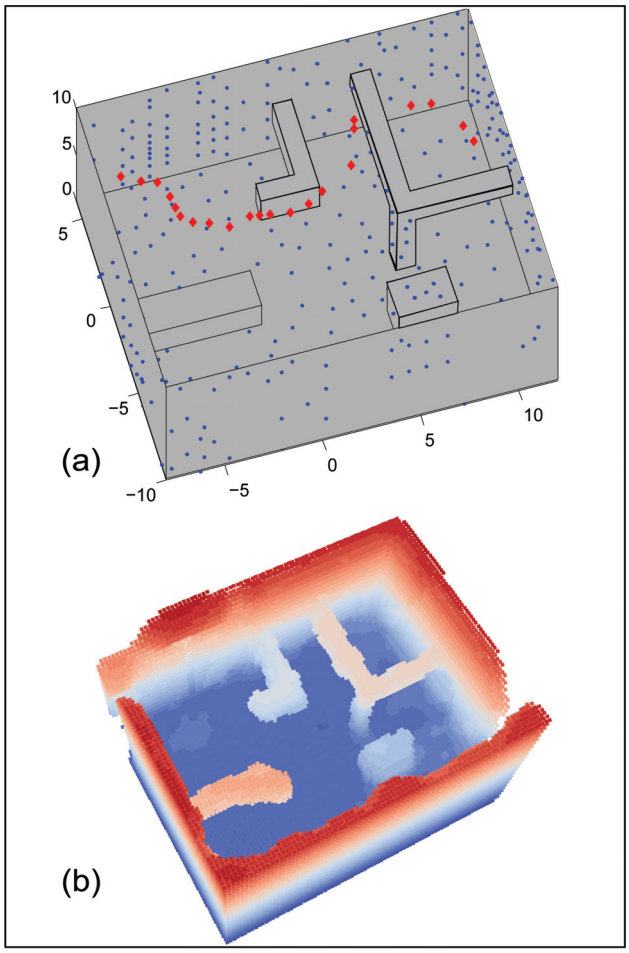


Figure 4: 3D Simulated Room Experiment. (a) Ground Truth with rangefinder sensor poses shown as red diamonds and the laser returns used after active sampling shown in blue dots. (b) Plot of test points labelled as occupied by the classifier.

both viewpoints despite the relative sparseness of the training data and occlusions created in a number of areas. A ROC curve of the outputted map was generated using unused observations as the ground truth. The area under the curve was 0.958 with a false positive detection rate of 0.076 for a true positive rate of 0.9. In some regions, the occlusions are too large to make a reliable classification such as close to the centre of Fig. 6(b). The variance also increases in these areas and could be used as a metric to decide where additional observations should be made.

## 5 Conclusions

In this paper, we presented a new occupancy map technique based on integral kernels. Our method is able to model observations from both line segments representing laser beams and points representing laser returns. Even though the paper explored the use of integral kernels for mapping tasks in robotics, the technique is general and can be used to deal

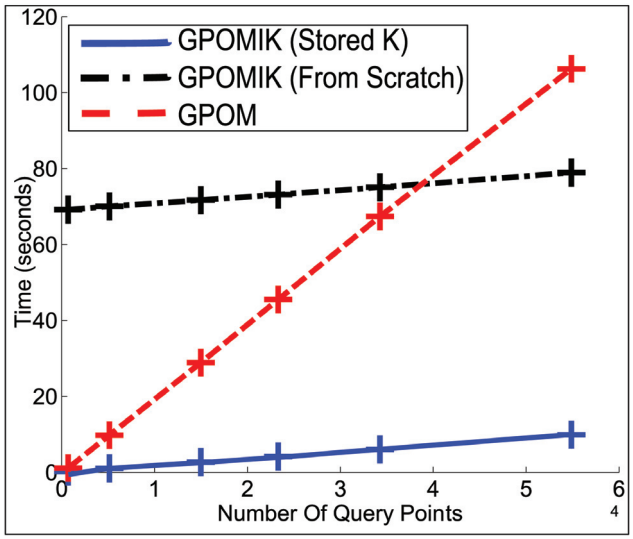


Figure 5: Time comparison between the GPOMIK and the GPOM on the 3D simulated dataset.

with problems were observations arrive not as points but as lines, areas or volumes in $3D$ space. This leads to more accurate algorithms as the physicality of the problem is better represented.

We believe that Gaussian process occupancy maps with integral kernels will have a large range of applications, from navigation tasks in mobile robotics, to object representation in grasping problems. The work can be extended to include noisy observations as in (O'Callaghan, Ramos, and Durrant-Whyte 2010) or to infer object classes for occupied regions. The ability of the technique to handle very sparse data sets and still recover the general shape of the environment is very encouraging as demonstrated in our 3D examples.

## Acknowledgements

This work is partially supported by the Australian Research Council (ARC).

## References

Bohg, J.; Johnson-Roberson, M.; Björkman, M.; and Kragic, D. 2010. Strategies for multi-modal scene exploration. In *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Gan, S.; Yang, K.; and Sukkarieh, S. 2009. 3d path planning for a rotary wing uav using a gaussian process occupancy map. In *Proceedings of the Australasian Conference on Robotics and Automation (ACRA 2009)*.

Gentleman, W. M. 1972. Implementing clenshaw-curtis quadrature, i methodology and experience. *Commun. ACM* 15:337–342.

Lang, T.; Plagemann, C.; and Burgard, W. 2007. Adaptive non-stationary kernel regression for terrain modeling. In *Robotics: Science and Systems (RSS)*.

Moravec, H., and Elfes, A. E. 1985. High resolution maps from

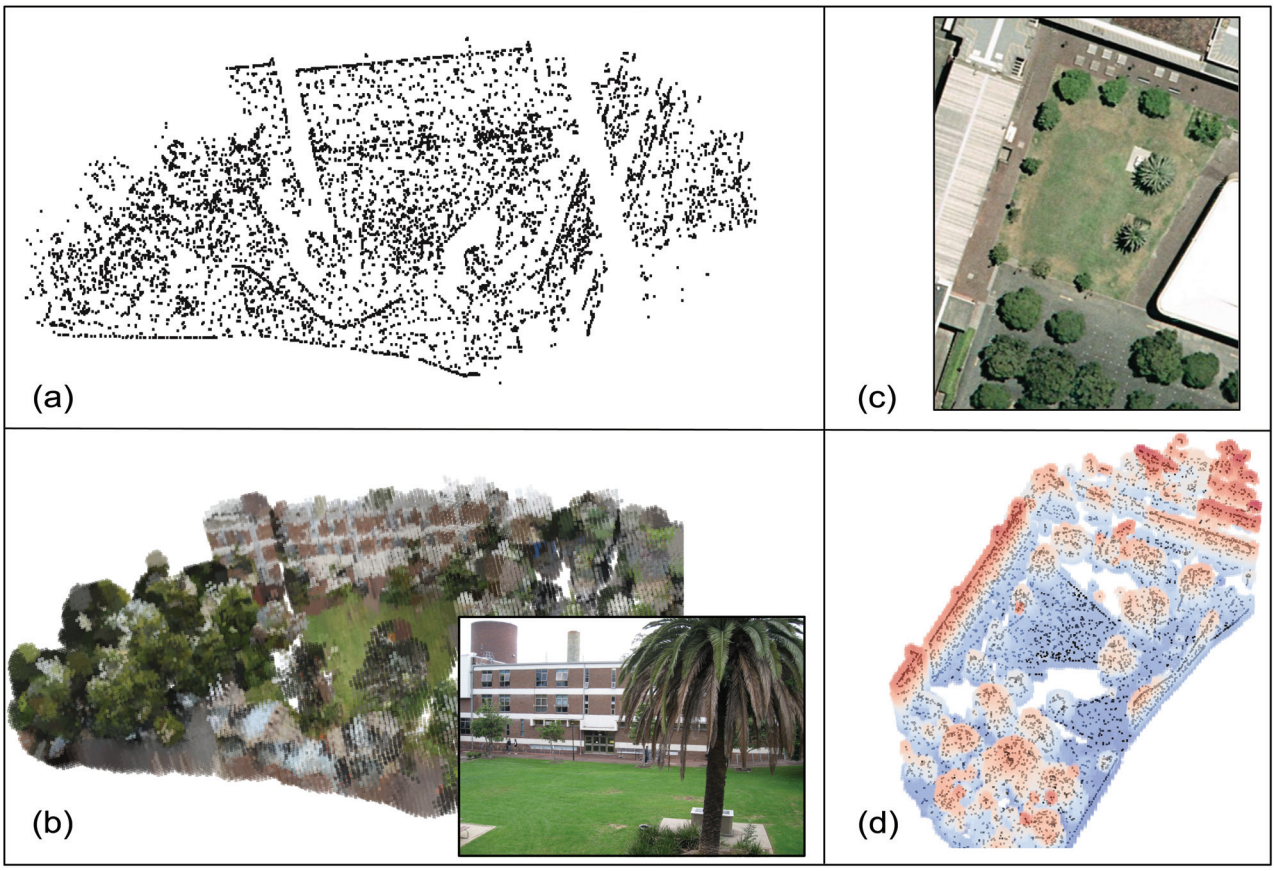


Figure 6: Real outdoor 3D dataset. (a) Training Points. (b & d) Test points that were classified as occupied by the classifier with a variance less than 1. Colour and an inset showing a section of the environment are added to (b) for visualisation purposes. The viewpoints in (a) and (b) are identical. (c) Aerial image of mapped environment from a similar orientation as (d)

wide angle sonar. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 116–121.

Murray-Smith, R., and Pearlmutter, B. 2005. Transformations of gaussian process priors. In Winkler, J.; Niranjan, M.; and Lawrence, N., eds., *Deterministic and Statistical Methods in Machine Learning*, volume 3635 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg. 110–123.

O'Callaghan, S. T.; Ramos, F. T.; and Durrant-Whyte, H. 2009. Contextual occupancy maps using gaussian processes. In *ICRA'09: Proceedings of the 2009 IEEE international conference on Robotics and Automation*, 3630–3636. Piscataway, NJ, USA: IEEE Press.

O'Callaghan, S. T.; Ramos, F. T.; and Durrant-Whyte, H. 2010. Contextual occupancy maps incorporating sensor andlocation uncertainty. In *ICRA'10: Proceedings of the 2010 IEEE international conference on Robotics and Automation*. Piscataway, NJ, USA: IEEE Press.

Osborne, M.; Rogers, A.; Ramchurn, S.; Roberts, S. J.; and Jennings, N. 2008. Towards real-time information processing of sensor network data using computationally efficient multi-output gaussian processes. In *International Conference on Information Processing in Sensor Networks (IPSN 2008)*, 109–120.

Osborne, M. A. 2010. *Bayesian Gaussian Processes for Sequential Prediction, Optimisation and Quadrature*. Ph.D. Dissertation, University of Oxford.

Paskin, M., and Thrun, S. 2005. Robotic mapping with polygonal random fields. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 450–458.

Platt, J. C. 2000. Probabilities for SV machines. In *Advances in Large Margin Classifiers*, 61–74. MIT Press.

Rasmussen, C. E., and Williams, C. K. I. 2006. *Gaussian Processes for Machine Learning*. MIT Press.

Smith, M.; Posner, I.; and Newman, P. 2010. Efficient non-parametric surface representations using active samplingfor push broom laser data. In *Proceedings of Robotics: Science and Systems VI*.

Stein, M. 1999. *Interpolation of Spatial Data: Some Theory for Kriging*. New York: Springer.

Tresp, V. 2000. A bayesian committee machine. *Neural Comput.* 12:2719–2741.

Veeck, M., and Burgard, W. 2004. Learning polyline maps from range scan data acquired with mobile robots. In *In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.