2009 IEEE International Conference on Robotics and Automation
Kobe International Conference Center
Kobe, Japan, May 12-17, 2009

Contextual Occupancy Maps using Gaussian Processes

Simon O'Callaghan, Fabio. T. Ramos and Hugh Durrant-Whyte

Abstract— **In this paper we introduce a new statistical modeling technique for building occupancy maps. The problem of mapping is addressed as a classification task where the robot's environment is classified into regions of occupancy and unoccupancy. Our model provides both a continuous representation of the robot's surroundings and an associated predictive variance. This is obtained by employing a Gaussian process as a non-parametric Bayesian learning technique to exploit the fact that real-world environments inherently possess structure. This structure introduces a correlation between points on the map which is not accounted for by many common mapping techniques such as occupancy grids. Using a trained neural network covariance function to model the highly non-stationary datasets, it is possible to generate accurate representations of large environments at resolutions which suit the desired applications while also providing inferences into occluded regions, between beams, and beyond the range of the sensor, even with relatively few sensor readings. We demonstrate the benefits of our approach in a simulated data set with known ground-truth, and in an outdoor urban environment covering an area of 120,000 m^2.**

I. INTRODUCTION

Constructing accurate maps of an environment remains a fundamental yet challenging task for mobile robots. The generation of meaningful spatial models of the robot's surroundings is central to the goals of navigation [1], [2] and path planning [3]. Since their introduction in the late 1980's by Elfes and Moravec [4], [5] occupancy grids have been widely used throughout the mobile robotics community. Their simplicity and computational efficiency have made occupancy grids popular particularly when mapping indoor environments and they can be easily adapted to process data from a wide range of sensors such as sonar [6], laser [7] and stereo vision [8].

However, despite their widespread success, occupancy grid models have a number of drawbacks. Perhaps the most obvious is the manner in which occupancy grids decompose high dimensional mapping problems into single dimensional calculations by making the 'independence between cells' assumption: The probability of each cell being occupied is solely dependent on the rays which pass through it and is not influenced in any way by the status of neighbouring cells. This simplification ignores the fact that in the real-world, cells of occupancy are not distributed randomly over the environment but rather there exists a spatial correlation between cells due to the physical structure of objects and environment. The independency assumption frequently results in cells of high uncertainty in regions where spatial context could assist in determining the state of a cell. This is perhaps most clearly seen in occluded areas or segments between sensor beams.

A number of other drawbacks to the traditional occupancy grids include the fact that they are constrained to representing structures at a single scale, suffer from discretisation errors, and require large amounts of memory to represent 3D environments at any reasonable level of detail.

Intuitively, the Gaussian process (GP) approach to occupancy maps seeks to exploit the fact that environments contain spatial structure to predict a continuous non-linear, non-parametric function representing the map. The GP uses an optimised covariance function to model context in the robot's surroundings. The Gaussian process is a Bayesian regression technique that trains a covariance function to model a non-parametric underlying distribution. A learning algorithm, which employs the Occam's razor principle [9] to avoid over-fitting, uses prior knowledge of the environment to shape the covariance function into a function that characterizes the correlation between points in the dataset. This knowledge can either be acquired from a previous scan of a similar environment, or using a subset of the new data to train the Gaussian process. The resulting predictive mean and variance distributions can then be used to classify regions of the robot's surroundings into areas of occupancy, non-occupancy or uncertainty using a probabilistic least squares classifier.

The primary contribution of this paper is the development of a mapping method which:

1) Removes independencies between data points.
2) Enables accurate maps to be generated with relatively sparse sensor information.
3) Eliminates the restriction of constructing a map on a single scale.
4) Produces an associated variance plot which could be used to highlight unexplored regions and optimise a robot's search plan.

This paper is organised as follows. Section II discusses related work. Section III describes the fundamental principles behind GPs and how these can be incorporated into the mapping problem. Section IV presents experimental results

Manuscript received February 16, 2009. This work was supported in part by the Australian Government through the EIPRS program as well as funding from the Centre of Autonomous Research.

S. T. O'Callaghan is with the Centre for Autonomous Systems (CAS), University of Sydney, NSW 2008 Australia. s.ocallaghan@cas.edu.au

F. T. Ramos works in the area of machine learning at CAS, University of Sydney, NSW 2006, Australia. f.ramos@cas.edu.au

H. Durrant-Whyte is the Research Director of the ARC Centre of Excellence for Autonomous Systems, University of Sydney, NSW 2006, Australia. h.durrant-whyte@cas.edu.au

for both simulated and real data sets. Section V provides conclusions and a discussion of future work.

II. Related Work

Many papers have attempted to address the problematic issues inherent in the occupancy grid with varying degrees of success [10], [11], [12]. One interesting approach is the use of forward models [13]. Let $\mathbf{m}$ denote the map or the robot's physical surroundings and $\mathbf{z}$ the observation. Forward models consider $p(\mathbf{z}|\mathbf{m})$ rather than the traditional inverse model $p(\mathbf{m}|\mathbf{z})$. This enables the likelihood of the sensor measurements to be calculated and the problem becomes an optimization task in the original high dimensional space. The method works particularly well with sonar where large beam-width would normally result in "regions of conflict" around narrow openings where certain cells appear to be both occupied and non-occupied. Optimising the likelihood in the original high dimensional space using the Expectation Maximisation (EM) algorithm helps to resolve this issue. An unfortunate drawback with this approach is the requirement to optimise the map each time an update is computed which may be impractical for online applications.

Pagac et al. [10] tried to overcome this independence assumption by developing an accurate model of range sensor performance and using the Dempster-Shafer inference rule to fuse the sensor readings into the map. While this evidential approach worked well with sonar because of its wide beam, the method has limited value when using a narrow-beam sensor such as a laser.

Paskin [11] proposed using of polygonal random fields to geometrically represent occupancy based on a consistent probability distribution over the environment. This created a dependency between regions allowing for inference in regions of the map which were not scanned by the sensor. A significant disadvantage of this approach is the computation required to get the random fields to converge, as noted by the authors. Even for reasonably sized indoor datasets, the random fields can take several hours to converge to a final representation.

The Gaussian process is a non-parametric method which is frequently used to solve regression and classification problems [14].Gaussian processes have previously been used with great success in mobile robotics. The Gaussian process' ability to learn behavioral characteristics of non-linear, non-parametric functions has resulted in their growing use in modeling real world phenomena [15].

Gaussian processes have also been used in the robotics community to effectively model less natural and smooth functions such as the work carried out by Plagemann in [16] and [17]. In the former a Gaussian Beam Process was used to interpolate the range to an object in regions between laser beams. The resulting predictive mean and variance outputs were accurate enough to be used to localize a robot within a known environment by comparing new sensor data with the trained Gaussian process representation.

III. Gaussian Process Applied to Mapping Problem

We treat the occupancy map as a form of classification problem. A GP is used to perform an initial regression on processed sensor data and a probabilistic least squares classification algorithm is trained to identify regions of occupancy, non-occupancy and uncertainty based on probability thresholds.

Section III is divided up into six main parts. The primary equations of the GP are first discussed and a brief background to the neural network covariance function is given. Subsections B and C introduce the training methods for the covariance function and the classification process, respectively. Section D details approximations of the covariance matrix. Section E incorporates the sensor model into the GP and Section F provides an overview of how predictions of the robot's environment can be updated online.

A. The Gaussian Process and Covariance Function

The occupancy map is based upon the GP's ability to predict $p(O|\mathbf{x})$, where O is the occupancy hypothesis and $\mathbf{x}$ represents a physical location within map. In this paper, $\mathbf{x}_i$ is assumed to be two dimensional, however it is relatively straightforward to extend the theory to three dimensions. O_i is essentially a class, either occupied or unoccupied, referenced by its corresponding location, $\mathbf{x}_i$.

The Gaussian process is used to fit a likelihood function to the training data $\{\mathbf{x}_i, y_i\}_{i=1:n}$ where n is the number of training points and y_i represents occupancy (+1) or non-occupancy (-1) at a given location. The resulting continuous function can then be used to interpolate between data points to predict the occupancy probability in unscanned and occluded regions.

The Gaussian process itself can be viewed as a distribution over functions and inference takes place directly in the space of functions. By assuming that all occupancy hypotheses, indexed by their corresponding location in the environment, are jointly Gaussian, we obtain

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

where

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

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

Here, $\mathbf{x}_*$ refers to a query or test location, $\mathbf{y}$ represents the noisey target data, σ_n^2 the variance of the global noise, and K is the covariance matrix. Typically, the data is scaled to have an empirical mean, μ, of 0. The elements of the covariance matrix $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ are defined depending on a covariance function k parameterised by hyperparameters θ. The global noise variance is taken to be quite low based on the fact that σ_n^2 relates to the output, O_i or the sensor's ability to detect occupancy/non-occupancy, and is not an input noise originating from uncertainty in the sensor's

bearing and range readings. A detailed explanation and derivation of the Gaussian process can be found in [14].

The trained covariance function is used to represent prior knowledge obtained about the underlying function $f(\cdot)$. Due to the non-stationary behaviour of typical map datasets (sudden changes from non-occupied to occupied regions), the commonly used squared exponential covariance function with its smoothing properties is not suitable for this application. The neural network covariance function is non-stationary and is capable of modeling the sharp shifts in the trend of $f(\cdot)$.

The covariance function is derived from a neural network with a single layer, a bias and N hidden units. It is demonstrated in [18] that as the number of units tends towards infinity, the network becomes a universal approximator for a wide range of transfer functions. By employing the error function as the neutral network's transfer function and allowing the number of hidden units to increase to infinity, [19] and [20] show that the following covariance function can be derived where the hidden weights are chosen to be Gaussian distributions with zero mean and a learnt covariance of Σ,

$$k(\mathbf{x}, \mathbf{x}_*) = \sigma_f^2 \arcsin\left(\frac{2\tilde{\mathbf{x}}^T \Sigma \tilde{\mathbf{x}}_*}{\sqrt{\left(1 + 2\tilde{\mathbf{x}}^T \Sigma \tilde{\mathbf{x}}\right)\left(1 + 2\tilde{\mathbf{x}}_*^T \Sigma \tilde{\mathbf{x}}_*\right)}}\right). \tag{4}$$

Here, $\tilde{\mathbf{x}} = (1, \mathbf{x}_1, ..., \mathbf{x}_d)^T$ is an augmented vector and σ_f^2 is a hyperparameter signal variance used to scale the correlation between points. Translational symmetry does not exist in this function giving it non-stationary properties. [21] demonstrates the ability of non-stationary covariance functions to adapt discontinuities in the underlying function of interest. This can be seen as an infinite layer neural network.

B. Training the Hyperparameters

A crucial aspect of the Gaussian process is the optimisation of the hyperparameters (σ_f^2, Σ and σ_n^2). These are key to developing a realistic model of the dataset and so it is important to ensure that the covariance function they generate accurately captures the extent of the correlation in the environment. This is achieved by maximising a log marginal likelihood function given by

$$p(\mathbf{y}|\mathbf{X}) = \int p(\mathbf{y}|\mathbf{f}, \mathbf{X}) p(\mathbf{f}|\mathbf{X}) d\mathbf{f}, \tag{5}$$

from which it follows that:

$$\ln p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}^T (K + \sigma_n^2 I)^{-1} - \frac{1}{2}\ln\left|K + \sigma_n^2 I\right| - \frac{n}{2}\ln 2\pi. \tag{6}$$

The primary advantage of the marginal likelihood is that it incorporates a trade-off between model fit and model complexity. A function which over-fits the data leads to poor inference and large uncertainties while an over-generalised outcome can result in a likelihood function which chooses to ignore many of the data points in favour of adopting a less responsive behaviour. Equation (6) helps to ensure an even balance between these two extremes.

The training data itself can be generated by randomly sampling a representative subset of the environment that consists of both occupied and non-occupied points obtained by discretising the sensor data in the Cartesian space. The optimal hyperparameters for the neural network covariance function are learnt by maximising the marginal likelihood over those training points using a combination of simulated annealing to identify an approximation of the global maximum and a gradient descent algorithm for further tuning of the parameters.

C. Probabilistic Least-squares Classification

While the predictive mean is useful for establishing the most likely appearance of the occupancy map based on the available sensor data, it can also be misleading if considered in isolation. One of the key advantages of the Gaussian process is its ability to calculate the variance of each prediction. Not only does the variance provide a method of identifying unexplored regions of high uncertainty within the environment but it can also be combined with the predictive mean to generate a distribution representing the probability of occupancy for each point on the map using a probabilistic least-squares classification technique described by Rasmussen in [14].

A post-processing stage is used to "squash" the predictions through a sigmoid function whose hyperparameters (α and β) were determined using a 'Leave-One-Out' (LOO) approach. Platt introduces this sigmoid function in [22] and the implemented version for training the sigmoid's parameters is

$$p(y_i|\mathbf{X}, y_{-i}, \theta) = \Phi\Big(\frac{y_i(\alpha\mu_i + \beta)}{1 + \alpha^2\sigma_i^2}\Big), \tag{7}$$

where $\Phi(\cdot)$ is the cumulative unit Gaussian, y_{-i} refers to the occupancy vector of all of the training data excluding the point $(\mathbf{x}_i, y_i)$. μ_i and σ_i refer to the predictive mean and variance at the point $\mathbf{x}_i$ respectively while θ signifies the trained hyperparameters of the covariance function.

Training α and β can be performed online despite the apparent need to evaluate a new likelihood function and variance distribution for each target point that is excluded. Although this requires determining a unique inverted covariance matrix of size $(n-1)$ x $(n-1)$ for each value of i considered, they can be acquired by simply partitioning the original matrix K^{-1} to eliminate the influence of training points $(\mathbf{x}_i)$. Wahba [23] adopted a similar approach with spline models. Thus the expressions for the LOO predictive mean and variance become

$$\mu_i = y_i - \frac{[K^{-1}y]_i}{[K^{-1}]_{ii}}, \tag{8}$$

$$\sigma_i^2 = \frac{1}{[K^{-1}]_{ii}}. \tag{9}$$

Using this probability distribution, the environment can be classified into occupied, unoccupied and unsure regions using user-defined thresholds which depends on the desired level of greediness.

D. Approximating the Covariance Matrix

One drawback of the Gaussian process is the requirement to invert the covariance matrix in order to calculate the predictive mean and variance. This introduces a computational complexity of $\boldsymbol{O}(n^3)$. For large datasets such as those generated for any outdoor laser scan of an appreciable length, this inversion of K becomes a substantial bottleneck in the algorithm's speed. However, because of the nature of the covariance function, the impact of distant training points have on the value of the test point is considerably less than the influence of nearby data points. Thus the covariance matrix can be accurately approximated by creating a sparsely populated matrix that only considers correlations between influential points near the test point. This can be efficiently implemented using a KD-tree implementation as in [24]. Fast sparse matrix inversion algorithms can then be employed to determine an approximate K^{-1} at a fraction of the computational time with only minor effects on the GP's predictions.

E. The Sensor Model

For each returned laser beam, the robot stores an angle and range (α_i, r_i) to the target at time t_i. This data can then be used to represent a non-occupied line segment originating from the robot's position at time t_i of length r_i at an angle of α_i relative to the platform. An occupied point is located at the far end of the line segment.

Discretising the line segment into numerous non-occupied points is not a viable option as the pre-defined distance between each non-occupied point effectively gives the map a minimum resolution. The number of non-occupied points required to accurately represent the non-occupied line segments increases rapidly with each additional scan and becomes computationally unworkable relatively quickly.

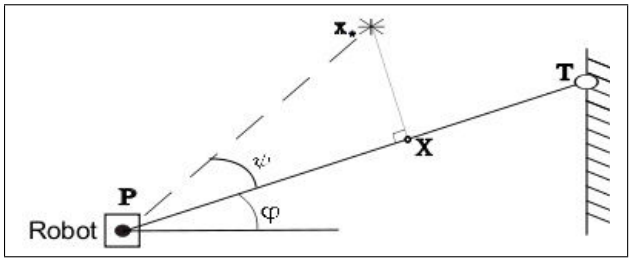


Fig. 1. Notation aid for Equation 10. Extraction of the training points from the sensor data following its mapping to Cartesian space.T and P correspond to the location of the beam's target and the robot's position at the instant that the beam is emitted, respectively. All of the locations are relative to the global coordinate frame. ψ is the angle created between the beam and the line $P\mathbf{x}_*$. ϕ is the angle which represents the direction of the sensor beam relative to the global x-axis.

A solution to this problem is to incorporate the continuous laser beam segment in the covariance function, Equation (4), using Equation (10) below to extract relevant training points from the sensor data.

$$\begin{bmatrix} x_x \\ x_y \end{bmatrix} = \begin{bmatrix} P_x \\ P_y \end{bmatrix} + |P\mathbf{x}_*| \cos(\psi) \begin{bmatrix} \cos(\phi) \\ \sin(\phi) \end{bmatrix}, \mathbf{x} \in \overline{PT} \qquad (10)$$

This has the effect of locating the closest point on each local laser beam to the test point of interest ($\mathbf{x}_*$) and then using these as the training data ($\mathbf{x}$) in the Gaussian process. This approach substantially reduces the size of the training dataset allowing for larger environments to be mapped at a lower computational cost as well as enabling an accurate representation of the continuous non-occupied beams to be passed into the Gaussian process.

Uncertainty in the range estimate is currently modeled by leaving a gap between the termination of the non-occupied line segment and the occupied point. The magnitude of this gap is dependent on the sensor's range variance as specified by the manufacturer. The Gaussian process can then decide on the occupancy hypothesis of this small gap of uncertainty based on the general trend of all other nearby sensor readings.

F. Updating the Probabilistic Classifier's Posterior

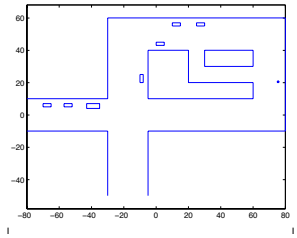
New sensor information which contradicts the Gaussian process' estimation of the environment can greatly alter the occupancy map and the robot's perception of its surroundings. Thus it is important that all new data can be quickly integrated into the system. Currently, for very large data sets this is achieved by temporarily storing the sensor data in locations which approximate their correct positions in the KD-tree. At certain intervals, such as when the robot is stationary or moving through areas of high confidence, the KD-tree can be rebuilt to more accurately position each reading within the data structure.

One of the principal advantages of the GP approach when compared to other methods which seek to evaluate dependencies between cells such as the technique suggested in [13] is that the optimisation is carried out only once and additional sensor data can be immediately incorporated into the process. Coarse, low resolution maps can be generated quickly to determine a possible path for the robot. Finer detailed, high resolution reconstructions can then be evaluated along paths of interest in order to save computational time.

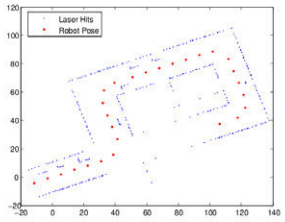
IV. Experimental Results

A. Simulated Dataset

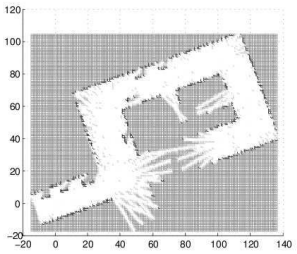
This GP approach to mapping was initially tested and refined using simulated data with a known ground truth. Fig. 2(a). is an example of one of the test environments used. It is scaled to represent a typical street scenario with main streets, side streets and a number of parked cars and vans. The location of the robot for each of the 28 scans, which in this case is assumed to be known, is highlight in red. The ground truth covers an area of approximately 18,000m^2. Each of the scans consists of 17 beams with a maximum range of 34 metres evenly spaced over a 180 degree sweep centered about the robot's direction of travel. This results in a sensor reading database of just 476 entries including non-returns. The locations of the returned laser hits when translated to reflect the robot's position during their associated scan are shown in Figure 2(b).



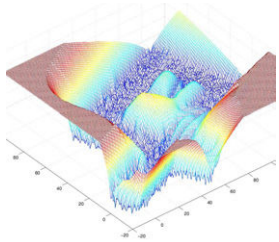
(a) Ground truth



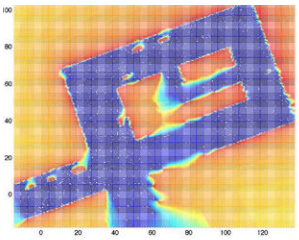
(b) Laser returns and robot pose



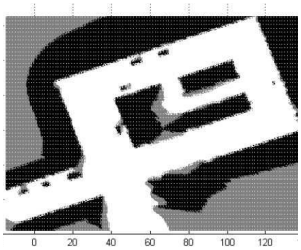
(c) Occupancy grid



(d) Predictive variance of Gaussian process



(e) Probability of occupancy versus location using the GP approach



(f) Bayesian map with conservative classification

Fig. 2. Sequence of images illustrating the simulated results for Bayesian mapping using a Gaussian process.

An occupancy grid using this laser data and some additional noise was generated for comparative purposes (Fig. 2(c)). The posterior of the probalistic classification using the GP occupancy mapping technique is displayed in Fig. 2(e) and its associated predictive variance is shown in Fig. 2(d). Fig. 2(f) shows the resulting map after classification using a thresholding procedure.

If the assumption of low noise in the laser data is applied to the occupancy grid, large portions of the map will remain unaltered from the prior, even at a resolution of 0.25m^2, due to the sparse nature of the sensor readings. These regions of uncertainty would actually increase if an improved laser with lower noise levels was employed as the occupancy grid only updates the occupancy hypothesis of cells within the scope of the sensor beam model. A common trick in occupancy maps is to add surplus noise to the sensor's bearing readings so as to artificially increase with width of the beam. While this method results in less unknown cells, this smearing is not true inference between beams and the additional noise leads to conflicts within cells near boundaries between occupied and unoccupied regions on the map such as doorways.

Even with the addition of noise to the data, large areas of uncertainty still exist in the occupancy grid. This is most evident in the occluded areas behind the vehicles and along roadways where the robot did not travel.

The GP mapping function was implemented in MATLAB and ran on a 2.33GHz Duo Core machine with 1 GB of RAM. Approximately 1000 test points could be evaluated per second. The occupancy map generated using the Gaussian process approach more accurately reflects the ground truth of the environment. From Fig. 2(e) All of the roads and side-streets can be identified despite the fact that the sensor data did not fully map them. The shapes of the buildings are also comparable to those of the actual environment. The cars and vans are also easily identifiable despite the fact that the areas behind them were reasonably occluded from the robot's sensor. Fig. 2(f), classifies the environment into occupied, unoccupied and unsure regions using a straightforward thresholding procedure which allocates class on the basis of probability of occupancy. Regions of uncertainty mainly exist in areas where the GP has relatively little information to accurately estimate the ground truth.

Other aspects worthy of note include the manner in which the probability of occupancy (correctly) tends towards 0.5 in regions with increasing uncertainty due to lack of information such as behind walls, in occluded areas, and further down along streets which the robot has not traveled. The associated variance map (Fig. 2(d)) highlights unexplored regions. This could be used in conjunction with the GP map to identify accessible areas of high variance in order to optimise search algorithms online.

A receiver operating characteristic (ROC) curve was generated for both mapping methods to compare the accuracy of each approach by plotting the rate of true positives (TP) detected versus the rate of false positives (FP) detected as the threshold of discrimination was varied (Fig. 3). The green dashed line which bisects the graph represents a process which randomly guesses the occupancy hypothesis for each cell. The ROC curve for our Bayesian GP implementation is displayed in blue and, as illustrated in Table I, it is possible to achieve a true positive detection rate of 0.95 while the false positive rate is kept at just 0.026. The occupancy grid's ROC curve in comparison indicates a FP rate of almost 0.2 for the same TP rate as a result of the artificial noise that was added in order to produce a reasonable representation of the environment with such sparse data.

TABLE I

NUMERICAL COMPARISON OF ROC CURVES

	Area under the curve	FP detection rate for TP detection rate of 0.95
Random Guess	0.5	0.95
Occupancy Grid	0.8938	0.1913
GP Occupancy Map	0.9948	0.026

B. *Outdoor Dataset*

An outdoor dataset was acquired using a SICK laser rangefinder mounted onboard a vehicle which traveled approximately 650 metres around two blocks of residential

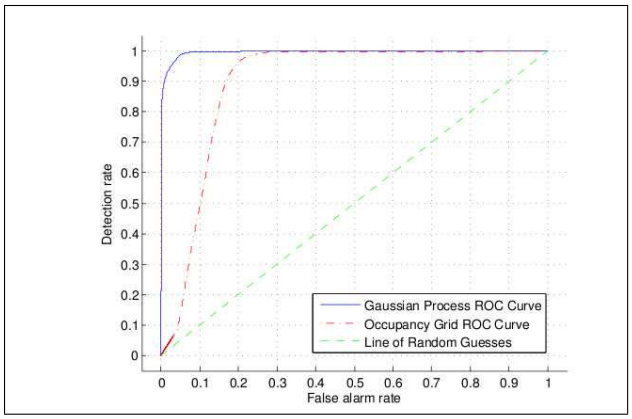
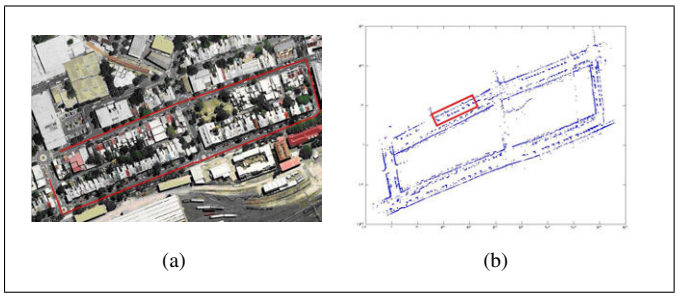


Fig. 3. ROC curves for occupancy grid and the GP mapping technique using simulated environment as a ground truth

and industrial housing (Fig. 6). The data consisted of approximately 600 scans which were aligned using an Iterative Closest Point algorithm [25]. To highlight the Gaussian process' ability to produce relatively accurate maps even in regions where only sparse sensor readings are available, just 4% of the beams from each scan were used as training data.

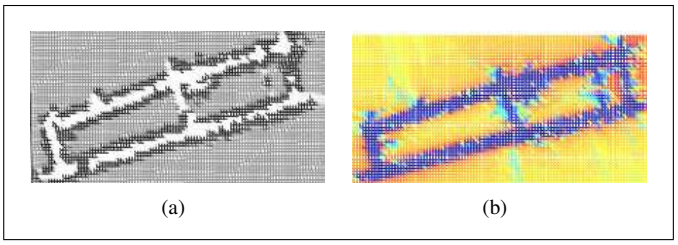
The hyperparameters were learnt using dense laser data acquired during the first 100 metres of the journey. Interestingly, the resulting hyperparameters were quite similar to those learnt during the tests in the simulated street environment; both sets could be interchanged with one another without significantly affecting the performance on the classifier. This could enable the optimisation process to be carried out prior to beginning a mission using data from previous missions in similar environments.



(a) (b)

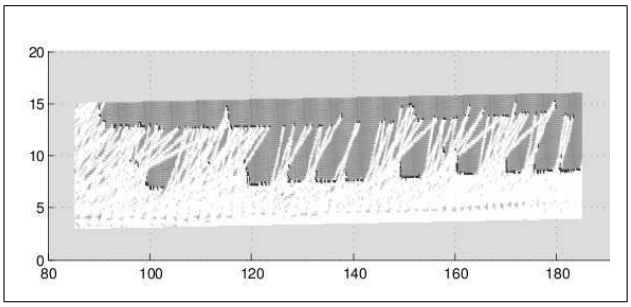
Fig. 4. Plan view of scanned area is shown in (a).The path of the robot is marked in red. Laser returns for the real dataset after ICP matching are displayed in (b). The red rectangle is the area covered by the high resolution maps in Figure 6

Due to the continuous nature of this approach, coarse maps such as that illustrated in Fig. 6 which covers an area of 120,000 m^2 can be generated in less than 5 seconds to identify large objects such as streets and buildings. Despite relatively little data, the beginnings of several side streets were identified in this low resolution reconstruction. It is still possible of distinguish a number of cars parked along the side of the road at this resolution of 5x5 m^2. For comparative purposes, a coarse sensor model was also implemented in order to generate an occupancy grid, Fig. 5(a). Although the occupancy grid also identifies some of the major roadways, the coarse sensor model leads to conflict within several of

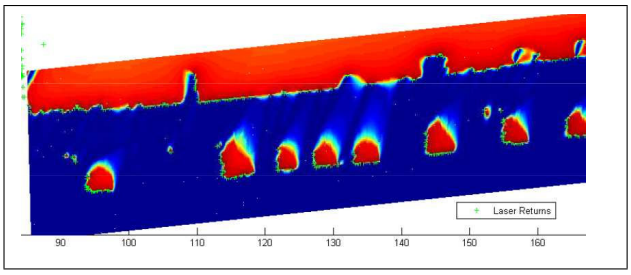


(a) (b)

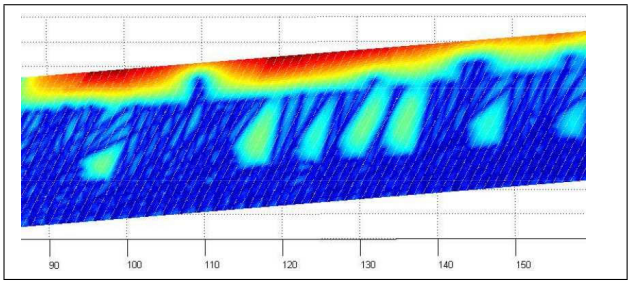
Fig. 5. Low Resolution 5 x 5 m Occupancy Grid (a) and GP map (b) of outdoor dataset.



(a) Occupancy Grid (0.2 x 0.2) m resolution



(b) High resolution Bayesian Map



(c) High resolution predictive variance

Fig. 6. A sequence of figures illustrating the application of the GP mapping technique to real data. Fig.4(a) shows a plan view of the scanned area. Figures 5(b) and 6(b) present the posteriors of the GP Bayesian mapping process. The former covers the entire area with a resolution of 5m x 5m while the latter focuses on a single street (0.1m x 0.1m). Fig. 6(c) shows the predictive variance over this same street.

the cells resulting in incorrect estimations of the occupancy hypothesis.

Focusing in on a particular area of the environment (highlighted in Fig 4(b)), the GP map's ability to estimate the probability of occupancy in occluded areas such as behind parked vehicles using the available sensor data as a context can be clearly seen. Compared to the occupancy grid (Fig 6(a)) which provides no inference into these regions apart from artificial smoothing achieved by adding surplus noise to the sensor, the GP map correctly identifies a number of

cars and the outline of the building despite gaps in excess of a metre between laser returns in a number of areas.

V. CONCLUSIONS AND FUTURE WORK

Gaussian processes with non-stationary neural network covariance functions provide an attractive technique for modeling occupancy in real-world environments. Optimising the covariance function's hyperparameters to model the underlying characteristics of the environment allows for estimation of the probability of occupancy, with associated variance, in regions where no sensory information is available. The continuous nature of the resulting underlying function allows for maps of various resolutions to be generated easily at a scale that suits the intended application. Predictive variance plots of the robot's surroundings also have the potential to enable the development of optimised search algorithms which could identify paths that would maximise information gained regarding environment's layout.

Despite these encouraging results, there is still room for improvement. A current drawback in our method is that spatial dependencies result in sensitivities to spurious sensor readings such as multi-path errors. Although such data is relatively infrequent with lasers, filtering of raw sensor information before it is passed to the GP would reduce the possibility of erroneous data negatively altering the behaviour of the underlying function.

Another potential area for further research is evolving the process so as to model a 3-D environment using data acquired at multiple angles of elevation. The Gaussian process is capable of accommodating additional dimensions without any alterations to the principle theory. The ability of this implementation to accurately represent real-world environments with quite sparse data readings could enable the large datasets typically acquired from 3-D laser range-finders to be reduced in size through sub-sampling without significantly affecting the appearance of the resulting occupancy map. Also, as the Gaussian process produces a continuous non-parametric underlying function which models the environment rather than a dense grid of cells, the computational issues which limit 3-D implementations of occupancy grids should not be as pressing with the GP approach.

Finally, with the current implementation it is assumed that the robot's position is always known however in practice it is usually the case that there is an associated variance with each position estimate. By modifying the covariance function, it may be possible to incorporate this uncertainty of position into the Gaussian process. Essentially, new training data that is acquired when the robot is relatively confident of its position would be strongly correlated to the rest of the data in the covariance matrix, K. However, if the variance in the robot's pose increases, the correlation between new sensor readings and previous data could then be weakened to account for this larger uncertainty in true position.

ACKNOWLEDGMENTS

This work has been supported by the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government.

REFERENCES

[1] B. Schiele and J. L. Crowley, "A comparison of position estimation techniques using occupancy grids," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 1994, pp. 1628–1634.

[2] J. Borenstein, Y. Koren, and S. Member, "The vector field histogram - fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 278–288, 1991.

[3] D. Kortenkamp, M. Huber, C. Cohen, U. Raschke, F. Koss, and C. Congdon, "Integrating high-speed obstacle avoidance, global path planning, and vision sensing on a mobile robot," pp. 53–71, 1998.

[4] A. Elfes, "Occupancy grids: a probabilistic framework for robot perception and navigation," Ph.D. dissertation, Carnegie Mellon University, 1989.

[5] H. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Magazine*, vol. 9, no. 2, pp. 61–74, 1988.

[6] H. Moravec and A. E. Elfes, "High resolution maps from wide angle sonar," in *Proceedings of the IEEE International Conference on Robotics and Automation*, March 1985, pp. 116–121.

[7] S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.

[8] D. Murray and C. Jennings, "Stereo vision based mapping and navigation for mobile robots," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, 1997, pp. 1694–1699.

[9] D. J. C. Mackay, "Information theory, inference & learning algorithms," 2002.

[10] D. Pagac, E. M. Nebot, and H. Durrant-Whyte, "An evidential approach to probabilistic map-building," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, 1996, pp. 745–750.

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

[12] K. Konolige, "Improved occupancy grids for map building," *Autonomous Robots*, vol. 4, no. 4, pp. 351–367, 1997.

[13] S. Thrun, "Learning occupancy grids with forward sensor models," *Autonomous Robots*, vol. 15, pp. 111–127, 2002.

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

[15] B. Ferris, D. Hohnel, and D. Fox, "Gaussian processes for signal strength-based location estimation," in *Proceedings of Robotics Science and Systems*, 2006, pp. 303–310.

[16] C. Plagemann, K. Kersting, P. Pfaff, and W. Burgard, "Gaussian beam processes: A nonparametric bayesian measurement model for range finders," in *Proceedings of Robotics: Science and Systems*, Atlanta, GA, USA, June 2007.

[17] C. Plagemann, F. Endres, J. Hess, C. Stachniss, and W. Burgard, "Monocular range sensing: A non-parametric learning approach," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Pasadena, CA, USA, 2008.

[18] K. Hornik, "Some new results on neural network approximation," *Neural Networks*, vol. 6, no. 9, pp. 1069–1072, 1993.

[19] R. M. Neal, *Bayesian Learning for Neural Networks*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1996.

[20] C. K. I. Williams, "Neural computation with infinite neural networks," *Neural Computation*, vol. 10, pp. 1203–1216.

[21] C. J. Paciorek and M. J. Schervish, "Nonstationary covariance functions for gaussian process regression."

[22] J. C. Platt, "Probabilities for SV machines," in *Advances in Large Margin Classifiers*. MIT Press, 2000, pp. 61–74.

[23] G. Wahba, *Spline Models for Observational Data*. Philadelphia, USA: SIAM, 1990.

[24] E. Snelson and Z. Ghahramani, "Sparse gaussian processes using pseudo-inputs," in *Advances in Neural Information Processing Systems 18*. MIT press, 2006, pp. 1257–1264.

[25] Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces," *International Journal on Computer Vision*, vol. 13, no. 2, pp. 119–152, 1994.