The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Large-scale terrain modeling from multiple sensors with dependent Gaussian processes

Shrihari Vasudevan, Fabio Ramos, Eric Nettleton and Hugh Durrant-Whyte
Australian Centre for Field Robotics, University of Sydney, NSW 2006, Australia
Email: shrihari.vasudevan@ieee.org, {f.ramos,e.nettleton,hugh}@acfr.usyd.edu.au

***Abstract*— Terrain modeling remains a challenging yet key component for the deployment of ground robots to the field. The difficulty arrives from the variability of terrain shapes, sparseness of the data, and high degree uncertainty often encountered in large, unstructured environments. This paper presents significant advances to data fusion for stochastic processes modeling spatial data, demonstrated in large-scale terrain modeling tasks. We explore dependent Gaussian processes to provide a multi-resolution representation of space and associated uncertainties, while integrating sensors from different modalities. Experiments performed on multiple multi-modal datasets (3D laser scans and GPS) demonstrate the approach for terrains of about 5 km^2.**

I. Introduction

Large-scale terrain mapping is an essential problem in a wide range of applications, from space exploration to mining and more. For autonomous robots to function in such high-value applications, an efficient, flexible and high-fidelity representation of space is critical. The key challenges in realizing this are that of dealing with the problems of uncertainty, incompleteness and handling highly unstructured terrain. Uncertainty and incompleteness are virtually ubiquitous in robotics as sensor capabilities are limited. The problem is magnified in a field robotics scenario due to sheer scale of the application (for instance, a mining or space exploration scenario).

State-of-the-art surface mapping methods employ representations based on tesselations. This process, however, does not have a statistically sound way of incorporating and managing uncertainty. The assumption of statistically independent data is a further limitation of many works that have used these approaches. While there are several interpolation techniques known, the independence assumption can lead to simplistic (simple averaging like) techniques that result in inaccurate modeling of the terrain. In [1], a Gaussian process based terrain modeling approach is proposed that provides a multi-resolution representation of the terrain, incorporates uncertainty in a statistically sound way and handles spatially correlated data in an appropriate manner.

Typically, sensory data is incomplete due to the presence of entities that occlude the sensors view. This is compounded by the fact that every sensor has a limited perceptual capability i.e. limited range and limited applicability. Thus, most large-scale modeling experiments would ideally require multiple sensory snapshots and multiple sensors to obtain a more complete model. These sensors may have different characteristics (range, resolution and accuracy). The problem is in fusing these multiple and multi-modal sensory datasets - this is the theme of the paper. Terrain data can be obtained using numerous sensors including 3D laser scanners and GPS. 3D laser scanners provide dense and accurate data whereas a GPS based survey typically comprises of a relatively sparse set of well chosen points of interest. Experiments reported in this work use datasets obtained from both these sensors to develop an integrated picture of the terrain.

The contribution of this work is a novel approach to fusing multiple, multi-modal datasets to obtain a comprehensive model of the terrain under consideration. The fusion technique is generic and applicable as a general Gaussian process fusion methodology. The fusion approach is based on the underlying principles of Gaussian processes and is thus well founded. Experiments conducted using large/real datasets obtained from GPS and laser scanner based surveys in real application scenarios (mining) are reported in support of the proposed approach.

II. Related work

State-of-the-art representations used in applications such as mining, space exploration and other field robotics scenarios as well as in geospatial engineering are typically limited to elevation maps ([2] and [3]), triangulated irregular networks (TIN's) ([4] and [5]), contour models and their variants or combinations ([6] and [7]). Each of these methods have their own strengths and preferred application domains. The former two are more popular in robotics. All of these representations, in their native form, do not handle spatially correlated data effectively and do not have a statistically correct way of incorporating and managing uncertainty.

Gaussian processes [8] (GP's) are powerful non-parametric learning techniques that can handle these issues. They produce a scalable multi-resolution model of the data under consideration. They yield a continuous domain representation of the data and hence can be sampled at any desired resolution. They incorporate and handle uncertainty in a statistically sound way and represent spatially correlated data in an appropriate manner. They model and use the spatial correlation of the given data to estimate the values for other unknown points of interest. In an estimation sense, GP's provide the best linear unbiased estimate [9] based on the underlying stochastic model of the spatial correlation between the data points. They basically perform an interpolation methodology similar to *Kriging* [10] – a standard interpolation technique used in the mining industry. GP's thus handle both uncertainty and incompleteness effectively.

Recently, Gaussian processes have been applied in the context of terrain modeling - see [11] and [1]. The former work is based on using a non-stationary equivalent of a stationary squared exponential covariance function [12] and incorporates kernel adaptation techniques to handle smooth surfaces as well as inherent (and characteristic) surface discontinuities. It introduces the idea of a "hyper-GP", using

a stationary kernel, to predict the most probable length scale parameters to suit the local structure. It also proposes to model space as an ensemble of GP's to reduce computational complexity. The latter work [1], proposes the use of non-stationary kernels (neural network) to model large-scale discontinuous spatial data. It shows that using a suitable non-stationary kernel can directly result in modeling local structure and smoothness. It also proposes a local approximation methodology to address scalability issues relating to the application of this approach to large-scale datasets. This approximation technique is based on an efficient hierarchical representation of the data. It compares performances of GP's based on stationary (squared exponential) and non-stationary (neural network) kernels as well as several other standard interpolation methods applicable to elevation maps and TIN's, in the context of large-scale terrain modeling. It proves that the non-stationary neural-network GP is a very competitive modeling option in comparison to standard interpolation methods (including polynomial interpolation methods [13]) for dense and/or relatively flat data and significantly better in the case of sparse and/or complex data.

Works from the graphics community that relate to this work include [13] and [14]. The former develops an approach to obtain a smooth manifold surface for a point-set through local polynomial approximations using a moving least squares approach. The latter work develops an approach to estimating the uncertainty of a point as the likelihood of a surface fitting the point-set, passing through the point in consideration. This too uses a local least squares approach. Local weighting of points is done using Gaussian influence functions. GP's use the idea that any finite set of random variables is jointly Gaussian distributed towards estimation of the quantity of interest as well as its uncertainty. This is done by conditioning the Gaussian distribution. The estimation results in a weighted combination of the point-set or a local neighborhood of the points. The uncertainty is computed in a similar light to [14]; it looks at the local support for a query point (points in the neighborhood and their correlation to the query point). Additionally, GP's provide a non-parametric (data is neither lost nor modified), multi-resolution (sample a continuous distribution at any desired resolution), flexible (different kernels may be used, not just Gaussian) representation which can be learnt through a Bayesian learning framework that automatically handles the model (parameter) selection problem effectively.

Data fusion in the context of Gaussian processes is required by the presence of multiple, multi-modal, incomplete and uncertain datasets of the entity being modeled. Two recent works that attempt this problem include [15] and [16]. The former bears a "hierarchical learning" flavor to it in that it demonstrates how a GP can be used to model an expensive process by (a) modeling a GP on an approximate or cheap process and (b) using the many input-output data from the approximate process and the few samples available of the expensive one together in order to learn a GP for the expensive process. The latter work attempts to generalize arbitrary transformations on GP priors through linear transformations. It hints at how this framework could be used to introduce heteroscedasticity and how information from different sources could be fused. However, specifics on how the fusion can actually be performed are beyond the scope of the work.

This paper builds on the work presented in [1]. It extends the GP terrain modeling approach to handle multiple multi-modal datasets by developing a data fusion methodology. It treats the data fusion problem as one of (a) modeling each data set using a GP and (b) formulating the data fusion problem as a conditional estimation problem wherein estimation of a GP is improved using information from other GP's - through learning auto-covariances and cross-covariances between them. This idea has been inspired by recent machine learning contributions in GP modeling ([17] and [18]), the latter approach being based on [19]. In kriging terminology, this idea is akin to co-kriging ([20]). This formalism is used to demonstrate data fusion of multiple multi-modal terrain datasets by casting the problem as a conditional estimation problem given multiple dependent GP's. It is also used to demonstrate simultaneous modeling of both elevation and color of terrain data. Experiments are performed on large-scale terrain data obtained from real mining scenarios. The scale of the experiments represents a first of its kind in the context of the topic. Towards ensuring the scalability of the approach, approximation methods have been used in both the learning and inference stages. The contribution of this work is thus a novel method of fusing multiple multi-modal large-scale datasets (terrain data, in this case) into an integrated model using GP's. Note that this work develops only the fusion methodology. The registration of individual datasets to a common reference frame is assumed given for this work.

III. Approach

A. Gaussian processes

Gaussian processes ([8]) (GP's) provide a powerful framework for learning models of spatially correlated and uncertain data. GP regression provides a robust means of estimation and interpolation of elevation information and can handle incomplete sensor data effectively. GP's are non-parametric approaches in that they do not specify an explicit functional model between the input and output. They may be thought of as a Gaussian probability distribution in function space and are characterized by a mean function $m(\mathbf{x})$ and the covariance function $k(\mathbf{x}, \mathbf{x}')$ where

$$m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})], \quad (1)$$

$$k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))], \quad (2)$$

such that the GP is written as

$$f(\mathbf{x}) \sim \mathrm{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \quad (3)$$

The mean and covariance functions together specify a distribution over functions. In the context of the problem at hand, each $\mathbf{x} \equiv (x, y)$ and $f(\mathbf{x}) \equiv z$ of the given data. The covariance function models the relationship between the random variables corresponding to the given data. Although not necessary, the mean function $m(\mathbf{x})$ may be assumed to

be zero by scaling the data appropriately such that it has an empirical mean of zero. There are numerous covariance functions (kernels) that can be used to model the spatial variation between the data points. The most popular kernel is the *squared-exponential* kernel given as

$$k(\mathbf{x}, \mathbf{x}') \;=\; \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{x}')^T\Sigma(\mathbf{x}-\mathbf{x}')\right) \qquad (4)$$

where k is the covariance function or kernel; $\Sigma = \begin{bmatrix} l_x & 0 \\ 0 & l_y \end{bmatrix}^{-2}$ is the length-scale matrix, a measure of how quickly the modeled function changes in the directions x and y. The set of parameters l_x , l_y are referred to as the kernel hyperparameters. Gaussian process regression uses the idea that for a GP, any finite subset of random variables is jointly Gaussian distributed. Thus, any finite set of training (evaluation) data and test data are jointly Gaussian distributed. This idea, shown in Equation 5, yields the standard GP regression Equations 6 and 7 which respectively represent the posterior/expected-value/mean-value and the variance/uncertainty in the prediction.

$$\begin{bmatrix} \mathbf{z} \\ f_* \end{bmatrix} \;\sim\; N\left(0\,,\, \begin{bmatrix} K(X,X) + \sigma_n^2 I & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix}\right) \qquad (5)$$

$$\bar{f_*} \;=\; K(X_*,X)[K(X,X) + \sigma_n^2 I]^{-1}\mathbf{z}. \qquad (6)$$

$$\begin{aligned} \mathrm{cov}(f_*) \;&=\; K(X_*,X_*) - \\ &\quad K(X_*,X)[K(X,X) + \sigma_n^2 I]^{-1}K(X,X_*). \end{aligned} \qquad (7)$$

For n training points and n_* test points, $K(X, X_*)$ denotes the $n \times n_*$ matrix of covariances evaluated at all pairs of training and test points. The terms $K(X, X)$, $K(X_*, X_*)$ and $K(X_*, X)$ can be defined likewise. σ_n^2 represents the noise variance in the observed data, it is learnt along with the other GP hyperparameters. The function values (f_*) corresponding to the test locations (X_*) given the training inputs X, training outputs $\mathbf{z}$ and the covariance function (kernel) are given by Equation 6 and their uncertainties, by Equation 7. A detailed report on Gaussian process modeling of large-scale terrain data (individual datasets which may be from any sensor) is presented in [1].

B. Multi-output / Dependent Gaussian processes

Multi-output Gaussian processes (MOGP's or multi-task GP's) extend the GP approach outlined before to handle multiple dependent outputs simultaneously. The main advantage of this technique is that the model exploits not only the spatial correlation of data corresponding to one output but also those of the other outputs. This improves GP regression/prediction. Two works in this area that have inspired this work include [17] and [18]. In [17], the shared covariance function is learnt as a product of individual covariance functions and an inter-task similarity matrix. The work [18] uses the process convolution approach [19] to derive closed form solutions to auto and cross covariance functions for two dependent GP's. The approach presented in this paper integrates both of these ideas to allow for increased flexibility in learning dependent GP models.

The objective is to model terrain data obtained as (x, y, z) coordinates from multiple and multi-modal datasets. Given the GP models of these datasets (as obtained above), the objective would then be to estimate an elevation map at any chosen resolution and any chosen region of the terrain under consideration. This can be achieved by performing a conditional estimation given the different datasets / their GP models. In the context of GP's, this amounts to conditional GP regression. The problem can be specified as

$$\mathbb{E}[\, f_*(\mathbf{X}_*)\,]\;,\; var(f_*(\mathbf{X}_*)) \mid X_i\;,\; \mathbf{z_i}\;,\; GP_i\;,\; X_*\;, \qquad (8)$$

where $X_i = (x_i, y_i)$ and $\mathbf{z_i} = z_i$ are the given datasets, GP_i is the respective set of hyperparameters and i varies from 1 to the number of datasets available, henceforth denoted by nt. This estimation will need to take into account both the spatial correlation within each dataset as well as the spatial correlation across datasets. Correlations between GP's can be modeled using auto-covariances and cross-covariances between them. By performing GP regression that takes this information into account, conditional estimation can be achieved and this results in a fused elevation estimate given the individual datasets.

The process convolution approach ([19]) is a generic methodology which formulates a GP as a white noise source convolved with a smoothing kernel. Modeling the GP then amounts to modeling the hyperparameters of the smoothing kernel. The advantage of formulating GP's this way is that it readily allows the GP to be extended to model more complex scenarios, one such scenario being the multi-output or dependent GP's (DGP's). The following formulation is based on [19] and [18].

Given that one single terrain is being modeled, a single Gaussian white noise process (denoted by $X(s)$ and representing (x, y) information of the datasets) is chosen as the underlying latent process. This process, when convolved with different smoothing kernel (denoted by k_i) produce different datasets. For the purpose of this paper, the smoothing kernels are assumed to be squared exponential kernel taking the form shown in Equation 4. The result of this convolution is denoted by $U_i(s)$. The observed data is assumed to be noisy and thus an additive white Gaussian noise $N(0, \sigma_i^2)$ (denoted by $W_i(s)$) is added to each process convolution output to yield the final observations (denoted by $Y_i(s)$ and representing the z information of the datasets). Equation 10 shows the mathematical formulation of the process convolution approach,

$$Y_i(s) \;=\; U_i(s) \,+\, W_i(s), \qquad (9)$$

$$U_i(s) \;=\; \int_s k_i(s-\lambda)\, X(\lambda)\; d\lambda. \qquad (10)$$

The fusion GP regression will take into account data from the individual datasets as well as the auto and cross covariances between the respective GP's that model them. The auto-covariances and cross-covariances can be computed through a convolution integral as the kernel correlation, as

demonstrated in [18]. For two GP's $N(0, k_i)$ and $N(0, k_j)$ with length scale matrices Σ_i and Σ_j respectively, the auto and cross-covariances are specified by Equation 11

$$K_{ij}^U(x, x') = K_f * |\Sigma_i + \Sigma_j|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(x - x')^T \Sigma_{ij}(x - x')\right), \quad (11)$$

where $\Sigma_{ij} = \Sigma_i(\Sigma_i + \Sigma_j)^{-1}\Sigma_j = \Sigma_j(\Sigma_i + \Sigma_j)^{-1}\Sigma_i$. K_{ii}^U represents the auto-covariance of the i^{th} data set with itself and K_{ij}^U represents the cross covariance between the i^{th} and j^{th} datasets, without considering the noise components of the datasets. The K_f term in Equation 11 is inspired from [17]. This term models the task similarity between individual tasks. Incorporating it in the auto and cross covariances provides additional flexibility to the dependent GP modeling process. It is a symmetric matrix of size $nt * nt$ and is learnt along with the other GP hyperparameters. The covariance matrix term $K(X, X)$ in Equations 6 and 7 is then specified as

$$K = \begin{bmatrix} K_{11}^Y & K_{12}^Y & \dots & K_{1\,nt}^Y \\ K_{21}^Y & \dots & \dots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ K_{nt\,1}^Y & \dots & \dots & K_{nt\,nt}^Y \end{bmatrix}, \quad (12)$$

where

$$K_{ii}^Y = K_{ii}^U + \sigma_i^2 I \quad (13)$$

$$K_{ij}^Y = K_{ij}^U \quad (14)$$

K_{ii}^Y represents the auto-covariance of the i^{th} data set with itself and K_{ij}^Y represents the cross covariance between the i^{th} and j^{th} datasets. They also take the noise components of the datasets into consideration and are obtained as in Equations 13 and 14 respectively. $K(X_*, X)$ denotes the covariance between the test data and the sets of input data (from the individual datasets) that are used for GP regression. It is given by

$$K(X_*, X) = [K_{i1}^U(X_*, X_1),\ K_{i1}^U(X_*, X_2),\ \dots\ K_{i\,nt}^U(X_*, X_{nt})] \quad (15)$$

where i is the output to be predicted - it can vary from 1 to nt. $K(X_*, X_*)$ represents the a priori covariance of the test points and is specified by

$$K(X_*, X_*) = K_{ii}^U(X_*, X_*) + \sigma_i^2. \quad (16)$$

The noise term is added assuming the test points are as noisy as the data points of the i^{th} GP. Finally, $\mathbf{z}$ represents the sets of z data corresponding to the training data taken from each of the datasets,

$$\mathbf{z} = [\mathbf{z_1}, \mathbf{z_2}, \dots, \mathbf{z_{nt}}]. \quad (17)$$

The hyperparameters of the system that need to be learnt include $nt * (nt + 1)/2$ task similarity values, $nt * 2$ length scale values of the individual kernels and nt noise values correponding to the noise in the observed datasets. In the context of modeling a single terrain using multiple and multi-modal datasets, for each point, the GP that is spatially closest to the test point is chosen for performing GP regression. The regression takes into account spatial correlation with other datasets as described.

C. GP Learning and scalability considerations

The work [1] demonstrated GP learning and inference for a single large-scale terrain data set. GP learning is based on maximizing the marginal likelihood. GP inference is based on the property of GP's that any finite set of training and test points would be jointly Gaussian distributed. Both GP learning and inference are computationally expensive operations in that both require matrix inversion. This operation is of cubic complexity ($O(N^3)$, N being the number of points in the data set) with respect to the number of points in consideration.

This paper deals with the data fusion of multiple large-scale terrain datasets. In [1], an approximate GP inference method was introduced that was based on a moving-window / nearest-neighbor methodology and relied on an efficient hierarchical representation of the data (a KD-tree was used). GP inference was based only on the local neighborhood of points resulting in a reduced complexity ($O(m^3), m << N$, m being the number of points in the neighborhood of a query point). This approximation method is also used here and extended to handle multiple datasets for each GP regression performed.

The work [1] used uniform sampling to select training points from the data to be modeled as using the several hundred-thousand data for learning would be computationally infeasible. In this work, a GP learning approximation is used that is based on the same nearest-neighbor approximation idea that is used for GP inference. A small set of training points are identified through uniform sampling. The KD-tree is then used to select points in each of their neighborhoods as training points. Thus, "patches" of data are selected for training. The KD-tree representation of the available data thus aides in both learning and inference. Once the training data are selected, GP learning proceeds by using the maximum marginal likelihood framework detailed in [1] and using Equation 18.

$$\log\ p(\mathbf{z}|X, \theta) = -\frac{1}{2}\mathbf{z}^T K(X, X)^{-1}\mathbf{z} - \frac{1}{2}\log|K(X, X)| - \frac{N}{2}\log(2\pi), \quad (18)$$

where $\mathbf{z}$ (Equation 17) and X represent the *sets of data* from the multiple datasets available and N is the total number of points across the different datasets that are in consideration. $K(X, X)$ is defined as specified in Equation 12.

The KD-tree based nearest-neighbor GP approximation method enables GP inference using *multiple* large datasets. In order to ensure the scalability of the overall approach, a block-learning procedure is adopted to learn the GP models. Instead of learning with all training points at once, this work uses blocks of points in a sequential marginal likelihood computation process within the optimization step. The block size is pre-defined and depends on the computational resources available. The KD-tree based block learning guarantees that multiple large datasets can be handled using even

limited computing resources. As a result, the GP learning space complexity remains cubic in the number of points, however, points being selected in local neighborhoods results and learning being performed in blocks results in a reduced time complexity. In experiments conducted (see [21]), The KDT based block learning was significantly faster than the uniform sampling based block learning approach to GP learning, for a given number of points and an approximate error margin. This was attributed to two reasons - (1) the KD-tree based point selection is faster than a simple uniform sampling - because it uses an efficient hierarchical representation of the data and (2) learning of hyper-parameters for local neighborhoods is faster than learning them for a widely spread data set - because the same set of hyperparameters would fit well with an entire group of data rather than a single data point.

IV. Experiments

The experiments described here demonstrate data fusion for multiple single and multi-sensor terrain datasets. The technical report version of this paper [21] additionally describes experiments that demonstrate the MOGP/DGP concept, demonstrates data fusion of overlapping and non-overlapping datasets, evaluates the usefulness of the GP learning approximation and finally demonstrates the data fusion of multiple single-sensor terrain data sets. In all cases, the mean squared error (MSE) between the prediction and the ground truth is used as the performance metric. Datasets are split into three parts - training, test and evaluation. The first part is used for learning the GP model, the second part is used for MSE computation only (it provides the ground truth) and finally, the first and third parts together (essentially, all data not in the second part) are used to perform GP regression at the MSE test points as well as any other query points.

A. Simultaneous elevation and color modeling

Fig. 1. Small section of a single RIEGL laser scan from Mt. Tom Price, Australia. The data set has 151,990 points with both elevation and color (RGB) data.

This experiment aims to demonstrate the MOGP idea in the context of modeling both elevation and color of real terrain data. The squared exponential kernel was used. A small section of a RIEGL laser scan taken at Mt. Tom Price mine is used for this experiment. The dataset has 151990 points spread over 27.75 m X 52.75 m X 11.48 m . This dataset has both color (RGB) and elevation information for each point.

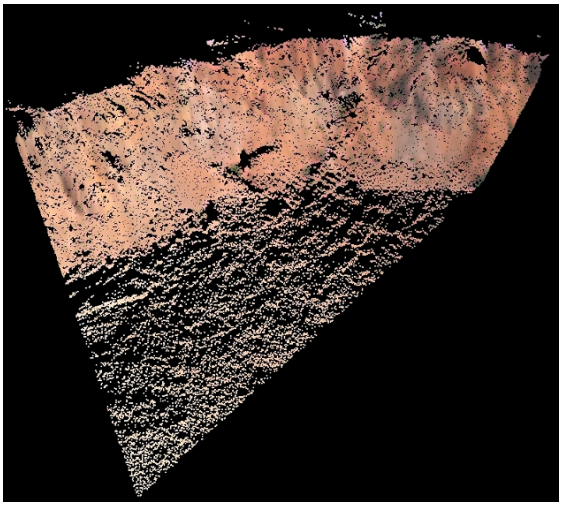

Fig. 2. A squared exponential kernel based MOGP being used to simultaneously model and predict elevation and color (RGB) data at 100,000 test points taken from the Tom price data set (see Figure 1). 2550 points were used for training each task (elevation, red, green and blue).

Figure 2 demonstrates the ability of the presented approach to simultaneously model elevation and color or real terrain data. The RGB and z data of 2550 points were used to train a four-task MOGP as described in Section III. GP learning used the KD-tree block learning procedure described in Section III-C. GP inference used the KD-tree based local approximation method introduced in [1]. This GP was tested on 100000 points uniformly selected from the data set. The test points were different from the training ones and used exclusively for testing. The MSE between the known (ground-truth) elevation and color values and those predicted by the GP are computed. The MSE values obtained were 0.0524 sqm for elevation and 0.0131, 0.0141 and 0.0101 squared units for red, green and blue respectively. Clearly, these values demonstrate the ability of the MOGP/DGP formalism to simultaneously model multiple aspects of the terrain being modeled. Also, it must be noted from Figure 2 that even the shades of grey (see Figure 1) are very effectively reproduced in the GP output. Note also that the scalability of the approach is demonstrated in that learning 4 tasks using 2550 points each is akin to learning a single GP with 10,200 data points. This was learnt in 2.75 hours using a stochastic (simulated annealing) and gradient-based (quasi-Newton) optimization, from random starting points. GP inference for the 100,000 points took just about 12.25 minutes.

B. Fusion of multiple multi-modal datasets

This experiment demonstrates data fusion of multiple multi-sensor data (RIEGL laser scanner and GPS survey) acquired from a large mine pit. Three datasets of the same area and of different characteristics were acquired from Mt.

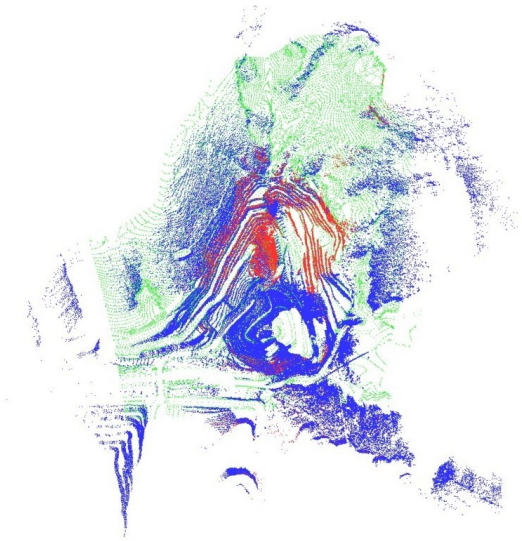

Fig. 3. Three multi-sensor datasets (a GPS survey and two laser scans) overlaid on one another for a clearer picture of the site in consideration. The points in blue represent Laser scan 1 (over 850,000 points spread over 2146.6 m x 2302.1 m x 464.3 m), the points in red represent the second laser scan (about 400,000 points spread over 1416.6 m x 2003.4 m x 497.8 m) and finally, the points in green represent the GPS survey data (a sparse data set consisting of 34,530 points spread over 1437.2 m x 1879.5 m x 380.5 m).

Tom Price mine in Western Australia. The first was a dense wide area (2146.6 m x 2302.1 m x 464.3 m) RIEGL laser scan comprising of over 850,000 points. The second was sparse GPS Survey having only about 34,530 points spread over 1437.2 m x 1879.5 m x 380.5 m. The third data set was a dense (about 400,000 points) RIEGL laser scan spread over a relatively smaller area as compared to the first scan (1416.6 m x 2003.4 m x 497.8 m). Figure 3 depicts the three datasets overlaid on each other to clarify the overall picture of the terrain in consideration.

The objective was to demonstrate the benefits of GP data fusion using these datasets. The sparse GPS data is first modeled alone, then fused with the first laser data set and then the pair are fused with the third data set (laser data). The results of the fusion process are summarized in Table I. The results indicate the root mean squared error (RMSE) and average change of uncertainty for a set of test points from the first data set over successive steps of the fusion process. Figures 4 and 5 depict the surface map and uncertainty estimates obtained after fusing the GPS data with the two laser scanner datasets. As shown in Table I, the uncertainty decreases with each successive fusion step. Thus, the required condition for fusion occurs. Further, it is observed that the RMSE also reduces with each fusion step. This justifies the benefits of data fusion in such a context. The uncertainty/RMSE reduction is more significant when the sparse GPS data is fused with the first dense laser scan. When the second dense laser scan is also fused, the gain in information is less than before. This is intuitive and expected. Note that the experiment here uses test points selected in patches (200 patches of 50 points), rather than a simple uniform point sampling. As demonstrated in [1], this deliberately reduces the influence of nearby points to observe the robustness of the underlying model in predicting elevation. The RMSE increases with the size of patches - this is intuitive and expected. For the same datasets, a simple uniform sampling of 10,000 test points yielded RMSE values in the range of about 3m. These values could be further improved by finding better solutions through the optimization process.

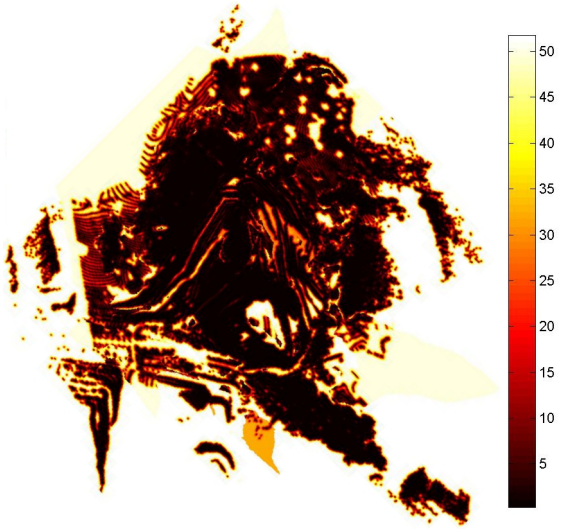

Fig. 5. Uncertainty (in meters) of the predicted elevation map obtained from the GP fusion of the GPS data and the two laser scanner datasets. Fringe areas that are not well supported by the individual datasets observe high prediction uncertainty.

V. Conclusion

This paper demonstrated the use of the multi-output / dependent Gaussian processes (GP's) in the context of fusing multiple multi-modal terrain datasets. This was done by casting the GP data fusion problem as a conditional estimation using several *Dependent GP's*. This formalism can also be used to demonstrate how color and elevation of the terrain can be simultaneously modeled using GP's. Large-scale experiments using real sensor data (3D data using both GPS as well as Laser scanners) taken from a mining scenario were used to demonstrate the approach. The approach presented in the paper has been specifically developed for large-scale applications - GP approximation methods have been developed in both learning and inference stages. The paper demonstrated a generic method of performing GP data fusion and the experiments validated the approach at a scale not attempted before in this field.

Acknowledgments

This work has been supported by the Rio Tinto Centre for Mine Automation and the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government. The authors acknowledge the support of Annette Pal, James Batchelor, Craig Denham, Joel Cockman and Paul Craine of Rio Tinto.

References

[1] S. Vasudevan, F. Ramos, E. Nettleton, and H. Durrant-Whyte, "Gaussian Process Modeling of Large Scale Terrain," *Journal of Field Robotics*, vol. 26(10), 2009, http://www-personal.acfr.usyd.edu.au/shrihari/svjfr09.pdf.

TABLE I

GP FUSION: MT. TOM PRICE DATASETS (GPS - LASER SCANNER FUSION)

Fusion sequence (datasets)	Root Mean Squared Error (m) (10000 test points)	Mean change in Uncertainty (std. dev. in m) (with respect to previous step of the fusion sequence)
GPS data only	9.99	-
GPS data & Laser data set 1	9.66	-2.78 (no cases of increase in uncertainty)
GPS data, Laser data set 1 & Laser data set 2	9.45	-0.59 (no cases of increase in uncertainty)

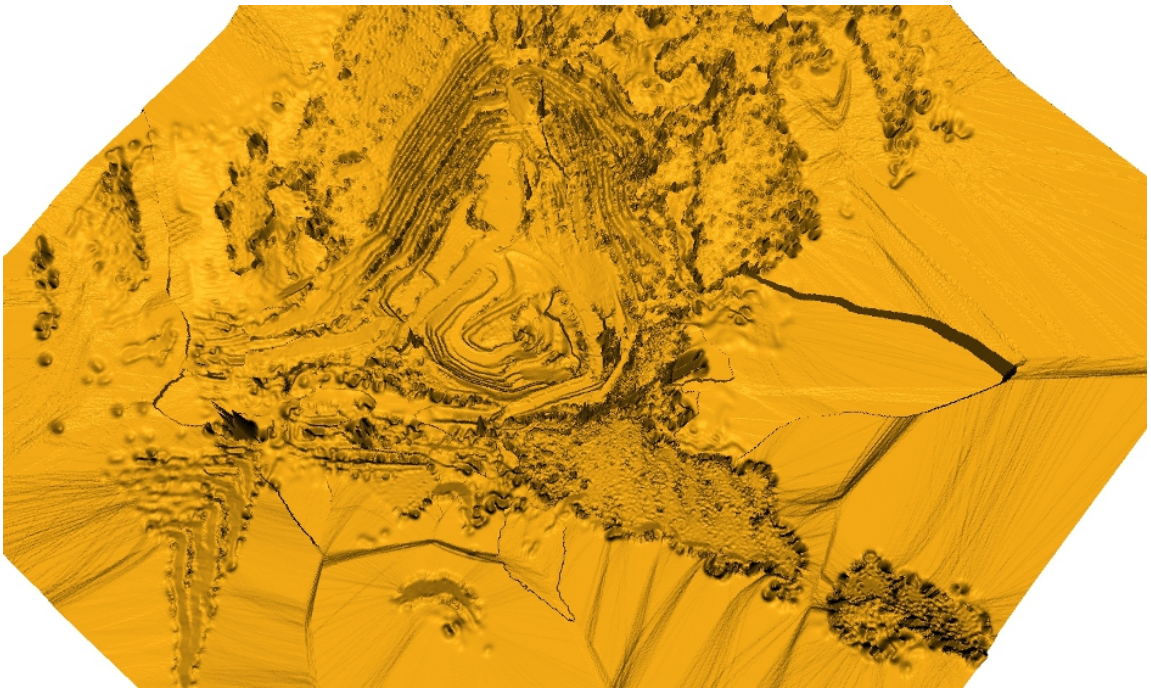

Fig. 4. Output of GP Fusion algorithm applied to the Tom Price datasets (GPS data and the two laser scanner datasets). The test data comprises of 1 million points. The surface map of the output elevation map is depicted in the image.

[2] S. Lacroix, A. Mallet, D. Bonnafous, G. Bauzil, S. Fleury, M. Herrb, and R. Chatila, "Autonomous rover navigation on unknown terrains: Functions and Integration," *International Journal of Robotics Research (IJRR)*, vol. 21(10-11), pp. 917–942, 2002.

[3] R. Triebel, P. Pfaff, and W. Burgard, "Multi-Level Surface Maps for Outdoor Terrain Mapping and Loop Closing," in *International Conference on Intelligent Robots and Systems (IROS)*, Beijing, China, October 2006.

[4] J. Leal, S. Scheding, and G. Dissanayake, "3D Mapping: A Stochastic Approach," in *Australian Conference on Robotics and Automation*, November 2001.

[5] I. Rekleitis, J. Bedwani, D. Gingras, and E. Dupuis, "Experimental Results for Over-the-Horizon Planetary exploration using a LIDAR sensor," in *Eleventh International Symposium on Experimental Robotics*, July 2008.

[6] H. Durrant-Whyte, "A Critical Review of the State-of-the-Art in Autonomous Land Vehicle Systems and Technology," Sandia National Laboratories, USA, Tech. Rep. SAND2001-3685, November 2001.

[7] I. D. Moore, R. B. Grayson, and A. R. Ladson, "Digital terrain modelling: A review of hydrological, geomorphological, and biological applications," *Hydrological Processes*, vol. 5-1, pp. 3–30, 1991.

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

[9] P. K. Kitanidis, *Introdcution to Geostatistics: Applications in Hydrogeology*. Cambridge University Press, 1997.

[10] G. Matheron, "Principles of Geostatistics," *Economic Geology*, vol. 58, pp. 1246–1266, 1963.

[11] C. Plagemann, S. Mischke, S. Prentice, K. Kersting, N. Roy, and W. Burgard, "A Bayesian regression approach to terrain mapping and an application to legged robot locomotion," *Journal of Field Robotics*, vol. 26(10), 2009.

[12] C. J. Paciorek and M. J. Schervish, "Nonstationary Covariance Functions for Gaussian Process Regression," in *Advances in Neural Information Processing Systems (NIPS) 16*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004.

[13] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva, "Point set surfaces," *IEEE Visualization*, pp. 21–28, October 2001.

[14] M. Pauly, N. J. Mitra, and L. Guibas, "Uncertainty and variability in point cloud surface data," in *Symposium on Point-Based Graphics*, 2004, pp. 77–84.

[15] M. El-Beltagy and W. Wright, "Gaussian processes for model fusion," in *International Conference on Artificial Neural Networks (ICANN)*, 2001.

[16] R. Murray-Smith and B. Pearlmutter, *Deterministic and Statistical Methods in Machine Learning, LNAI 3635*. Springer-Verlag, 2005, ch. Transformations of Gaussian Process priors, pp. 110–123.

[17] E. Bonilla, K. M. Chai, and C. Williams, "Multi-task gaussian process prediction," in *Advances in Neural Information Processing Systems 20*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. Cambridge, MA: MIT Press, 2007, pp. 153–160.

[18] P. Boyle and M. Frean, "Dependent gaussian processes," in *Advances in Neural Information Processing Systems 17*, L. K. Saul, Y. Weiss, and L. Bottou, Eds. Cambridge, MA: MIT Press, 2004, pp. 217–224.

[19] D. Higdon, *Quantitative Methods for Current Environmental Issues*. Springer, 2002, ch. Space and Space-Time Modeling Using Process Convolutions, pp. 37–54.

[20] H. Wackernagel, *Multivariate geostatistics: an introduction with applications*. Springer, 2003.

[21] S. Vasudevan, F. Ramos, E. Nettleton, and H. Durrant-Whyte, "Dependent gaussian processes for data fusion in large scale terrain modeling," Australian Centre for Field Robotics, The University of Sydney, Tech. Rep. CMA003.109, 2010.