

# Information Agents for Pervasive Sensor Networks

A. Rogers, S. D. Ramchurn and N. R. Jennings  
School of Electronics and Computer Science  
University of Southampton  
Southampton, SO17 1BJ, UK.  
{acr, sdr, nrj}@ecs.soton.ac.uk

M. A. Osborne and S. J. Roberts  
Department of Engineering Science  
University of Oxford  
Oxford, OX1 3PJ, UK.  
{mosb, sjrob}@robots.ox.ac.uk

## Abstract

*In this paper, we describe an information agent, that resides on a mobile computer or personal digital assistant (PDA), that can autonomously acquire sensor readings from pervasive sensor networks (deciding when and which sensor to acquire readings from at any time). Moreover, it can perform a range of information processing tasks including modelling the accuracy of the sensor readings, predicting the value of missing sensor readings, and predicting how the monitored environmental parameters will evolve into the future. Our motivating scenario is the need to provide situational awareness support to first responders at the scene of a large scale incident, and we describe how we use an iterative formulation of a multi-output Gaussian process to build a probabilistic model of the environmental parameters being measured by local sensors, and the correlations and delays that exist between them. We validate our approach using data collected from a network of weather sensors located on the south coast of England.*

## 1 Introduction

Sensor networks have recently generated a great deal of research interest within the computer and physical sciences, and their use for the scientific monitoring of remote and hostile environments is increasingly common-place (see [5] for a review of such environmental sensor networks). More recently, the notion of pervasive sensor systems has gained ground, and trial deployments of such networks are now taking place (the CitySense project of Harvard University is a good example [7]). In these systems, sensors owned by multiple stakeholders are ubiquitously deployed within urban environments and make their information available to multiple users directly through standard web interfaces.

Such open systems have many applications including traffic and pollution monitoring and local weather forecasting. Within the ALADDIN project

([www.aladdinproject.org](http://www.aladdinproject.org)) we are seeking to use such networks to provide situational awareness support to first responders at the scene of a large scale incident. We envisage providing these first responders with a mobile computer or personal digital assistant (PDA) that is capable of collecting information from local sensors, compiling a coherent world view, and then assisting in decision making. An example application would be to provide fire fighters with local weather information, and to predict wind changes using observations from nearby sensors.

Using pervasive sensor networks within such applications presents many novel challenges; not least the need for low-power wireless communication standards, self-describing data formats, and standard protocols such that sensors can advertise their existence and capabilities to potential users of the network. However, more significantly for us, many of the information processing tasks that would previously have been performed by the owner or single user of an environmental sensor network (such as detecting faulty sensors, fusing noisy measurements from several sensors, and deciding how frequently readings should be taken) are now delegated to the multiple different users of the system, all of whom may have different goals and may be using sensor readings for very different tasks. Furthermore, the open nature of the network (in which additional sensors may be deployed at any time, and existing sensors may be removed, repositioned or updated) means that these users may have only limited knowledge of the precise location, capabilities, reliability, and accuracy of each sensor.

Thus, there is a clear need for the mobile computers and PDAs carried by our first responders to incorporate an *information agent* that is capable of autonomously performing the acquisition and processing of information from such pervasive sensor networks. Given this, in this paper, we describe our work developing just such an agent. This agent uses a novel iterative formulation of a multi-output Gaussian process (described in more detail in [8]) to build a probabilistic model of the environmental parameters being measured by local sensors, and then uses this model to perform

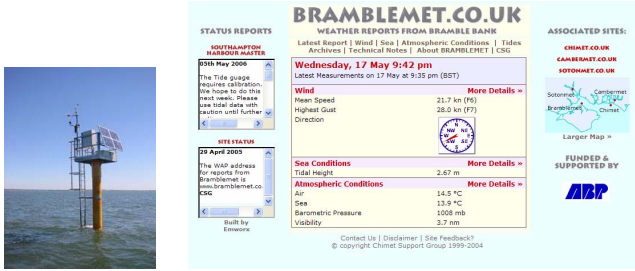


Figure 1: The Bramble Bank weather station and associated web site (see [www.bramblemet.co.uk](http://www.bramblemet.co.uk)).

a number of information processing tasks including: modelling the accuracy of the sensor readings, predicting the value of missing sensor readings, predicting how the monitored environmental parameters will evolve in the near future, and performing active sampling by deciding when and which sensor to acquire readings from.

In more detail, we describe how we have used a network of weather sensors on the south coast of England to validate this approach, and we illustrate its effectiveness by benchmarking against the more conventional single-output Gaussian processes that models each sensor independently. Our results on this data set are promising, and indicate that this approach will ultimately allow us to deploy information agents with minimal domain knowledge, and use principled machine learning techniques to autonomously acquire sensor readings, and perform information processing tasks.

## 2 Information Processing Problem

As discussed above, we require that our information agent be able to autonomously perform data acquisition and information processing despite having only limited specific knowledge of each of the sensors in its local neighborhood (e.g. their precise location, reliability, and accuracy). To this end, we require that it explicitly represent:

1. The noise in the sensor readings, and hence, the uncertainty in the environmental parameter being measured; sensor readings will always incorporate some degree of measurement noise, and thus there will always be some uncertainty in the agent's world picture.
2. The correlations or delays that exist between sensor readings; sensors that are close to one another, or in similar environments, will tend to make similar readings, while many physical processes involving moving fields (such as the movement of weather fronts) will induce delays and correlations between sensors.

We then require that the information agent use this explicit representation in order to:

1. Perform efficient active sampling by selecting when to take a reading, and which sensor to read from, such

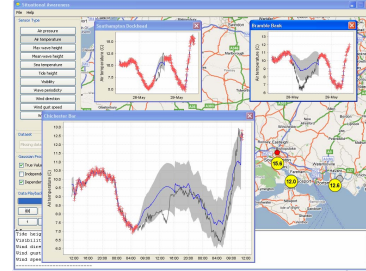


Figure 2: Java implementation of the information agent.

that the minimum number of sensor readings are used to maintain an agent's world uncertainty below a specified threshold (or similarly, minimising uncertainty given a constrained number of sensor readings).

2. Perform regression and prediction of sensor readings; that is, interpolate between sensor readings to predict the value of missing sensors (i.e. sensors that have failed or are unavailable through network outages), and perform short term prediction of sensor readings into the future in order to support decision making.

More precisely, we consider a multivariate regression problem in which we have  $m = 1 \dots M$  environmental parameters of interest (such as air temperature, wind speed or direction specified at different sensor locations) represented by the space  $\mathcal{Y} = \mathbb{R}^M$ . Given a set of  $N$  sensor readings,  $D = \{(t_1, \mathbf{y}_1), \dots, (t_N, \mathbf{y}_N)\}$ , where  $\mathbf{y}_i$  may be fully or partially specified (corresponding to observation of all, or some subset of the environmental parameters), we attempt to infer the value of  $\mathbf{y} = \{y^1, \dots, y^M\} \in \mathcal{Y}$  at any time  $t$ .

## 3 Gaussian Processes

Multivariate regression problems of the form described above have often been addressed using multi-layer neural networks. However, Gaussian processes (GPs) are increasingly being applied in this area, since they represent a powerful way to perform Bayesian inference about functions [11]. When using a GP, we assign a multivariate Gaussian prior distribution over the outputs of the regression problem and then produce analytic posterior distributions for outputs of interest, conditional on whatever sensor readings have been collected. Crucially, the posterior distributions are also Gaussian, and thus we have a predictive mean, and also a variance that explicitly represents uncertainty.

Gaussian process regression has a long history of use within geophysics and geospatial statistics (where the process is known as kriging [3]), but has only recently been applied within sensor networks. Examples here include the use of GPs to represent spatial correlations between sensors in order that they may be positioned to maximise mutual information [6], and the use of multi-variate Gaussians to

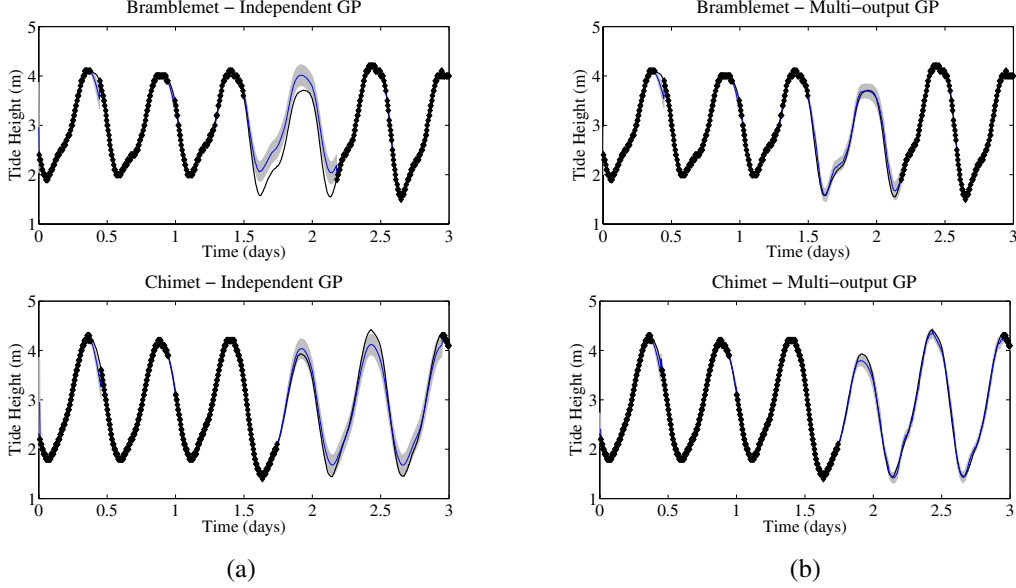


Figure 3: Prediction and regression of tide height data for (a) independent and (b) multi-output Gaussian processes.

represent correlations between different sensors and sensor types for energy efficient querying of a sensor network [4].

Our work differs from this earlier work in that we use a novel iterative formalism of a multi-output GP to represent both temporal correlations, and correlations and delays between sensors. Space precludes a full description of this algorithm (see [8] for the full details), however we describe the intuition behind this algorithm here.

### 3.1 Covariance Functions

The covariance matrix of the GP informs it of how different outputs are related to one another. To generate this matrix, we use *covariance functions*. Fortunately, there exist a wide variety of functions that can serve in this purpose [1], all of which can then be combined and modified in a multitude of ways. This gives us a great deal of flexibility in our modelling of functions, and covariance functions can be found to model periodicity, delay, noise and long-term drifts.

More specifically, we represent the covariance matrix by the Hadamard product of a covariance function over time alone, and a covariance function over environmental parameter labels alone, such that:

$$K([m, t], [m', t']) = C(m, m')K(t - d_m, t' - d_{m'}) \quad (1)$$

where  $\mathbf{d}$  represent delays between environmental parameters. Assuming no prior knowledge of what the correlations over environmental parameters are, we use the completely general *spherical parameterisation*,  $\mathbf{s}$ , such that:

$$C(m, m') = \text{diag}(\mathbf{1})\mathbf{s}^T \text{diag}(\mathbf{1}) \quad (2)$$

where  $\mathbf{1}$  gives represents an intuitive length scale for each environmental parameter, and  $\mathbf{s}^T \mathbf{s}$  is the correlation matrix [9]. Similarly, we can represent correlations over time

with a wide variety of covariance functions, incorporating as much domain knowledge as we have. However, in general, we find that the additive combination of a periodic term and a disturbance term performs well on a wide range of data sets, and we represent both using the standard Matérn class (with  $\nu = 5/2$ ), given by:

$$K(t, t') = h^2 \left( 1 + \sqrt{5}r + \frac{5r^2}{3} \right) \exp(-\sqrt{5}r) \quad (3)$$

where  $r = \left| \frac{t-t'}{w} \right|$  for non-periodic terms, and  $r = \sin \pi \left| \frac{t-t'}{w} \right|$  for periodic ones.

### 3.2 Marginalisation

In order to use the GP for regression or prediction, the correlation hyperparameters (i.e.  $\mathbf{l}$ ,  $\mathbf{s}$  and  $\mathbf{d}$ ), along with others such as the periods and amplitudes of each covariance term (i.e.  $h$  and  $w$ ), must be marginalised from our model. To each we assign an independent Gaussian or log Gaussian prior distribution (if the hyperparameter is strictly positive). We then use *Bayesian Monte Carlo* [10] in order to numerically resolve the non-analytic marginalisation integrals. This essentially involves the assignation of another GP to the likelihood of the data as a function of the covariance hyperparameters. We evaluate predictions for a set of sample hyperparameters, and use this second GP to infer what predictions for other possible hyperparameters, producing a posterior for our marginalised predictions.

### 3.3 Iterative Formulation

Gaussian processes have traditionally been used largely for regression, producing predictions for a fixed set of data.

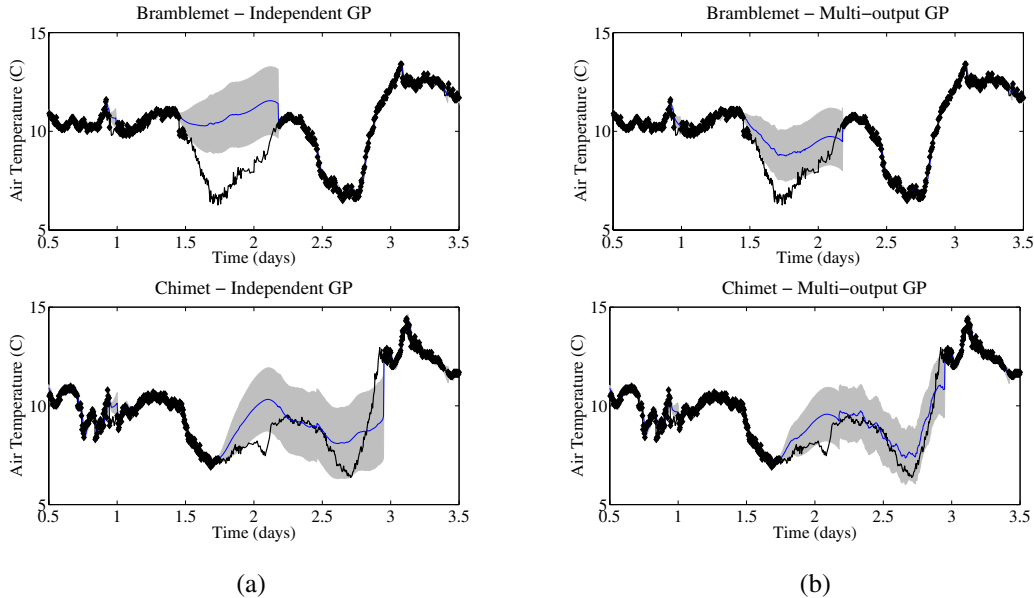


Figure 4: Prediction and regression of air temperature data for (a) independent and (b) multi-output Gaussian processes.

However, in our setting both the environmental parameters of interest and the data available are constantly updated. In order to manage this situation, we employ a novel iterative formulation of a GP, which allows us to efficiently update our predictions upon the receipt of new data.

Similarly, we allow the GP to discard old data once it judges it sufficiently uninformative, hence reducing memory usage and computational requirements. In this, it is guided by the uncertainty in its predictions; the GP will retain only as much data as necessary to achieve a pre-specified degree of accuracy (a principled form of ‘windowing’). These features give us an efficient on-line algorithm.

### 3.4 Active Data Selection

Our algorithm is also able to perform active data selection, whereby the GP decides for itself which observations it should take. In this, we use once again the uncertainty in our predictions as a measure of utility. For a GP, this uncertainty increases monotonically in the absence of new data – once it grows to our pre-specified threshold, our algorithm takes a sample in order to reduce it once again. The algorithm can also decide which observation to make at this time, by determining which sensor will allow it the longest period of grace until it would be forced to observe again. Hence we maintain our uncertainty below a specified threshold, while taking as few observations as possible.

## 4 Empirical Evaluation

In order to empirically evaluate the GP formalism described in the previous section, we have used a network of weather sensors (see [www.bramblemet.co.uk](http://www.bramblemet.co.uk)) located on the

south coast of England. This network consists of four sensors (each measuring a range of environmental parameters such as wind speed and direction, air temperature, sea temperature, tide height, etc.) which make up-to-date sensor measurements available through separate web pages (see figure 1). This data is used by recreational sailors to monitor local weather conditions, and by port authorities for deciding on shipping movements within the Port of Southampton. The use of such weather sensors is attractive since they have immediate application within our motivating disaster response scenario, they exhibit challenging correlations and delays, and they are subject to network outages that generate real instances of missing sensor readings on which we can evaluate our information processing algorithms.

In order to simulate the pervasive sensor network that we described in the introduction, we have supplemented each sensor web site with machine readable RDF data, and have implemented an information agent in Java (see figure 2) that is able to acquire readings from the weather sensors, parse and store the RDF data, and perform the information processing tasks described here.

## 5 Results

In order to validate our multi-output GP formalism we have applied it to real weather data (collected from the network described above), and compare it against conventional independent GPs in which each environmental parameter is modeled separately (i.e. correlations between these parameters are ignored). In this comparison, we present results for two different sensor types: tide height and air temperature. Tide height was chosen since it demonstrates the abil-

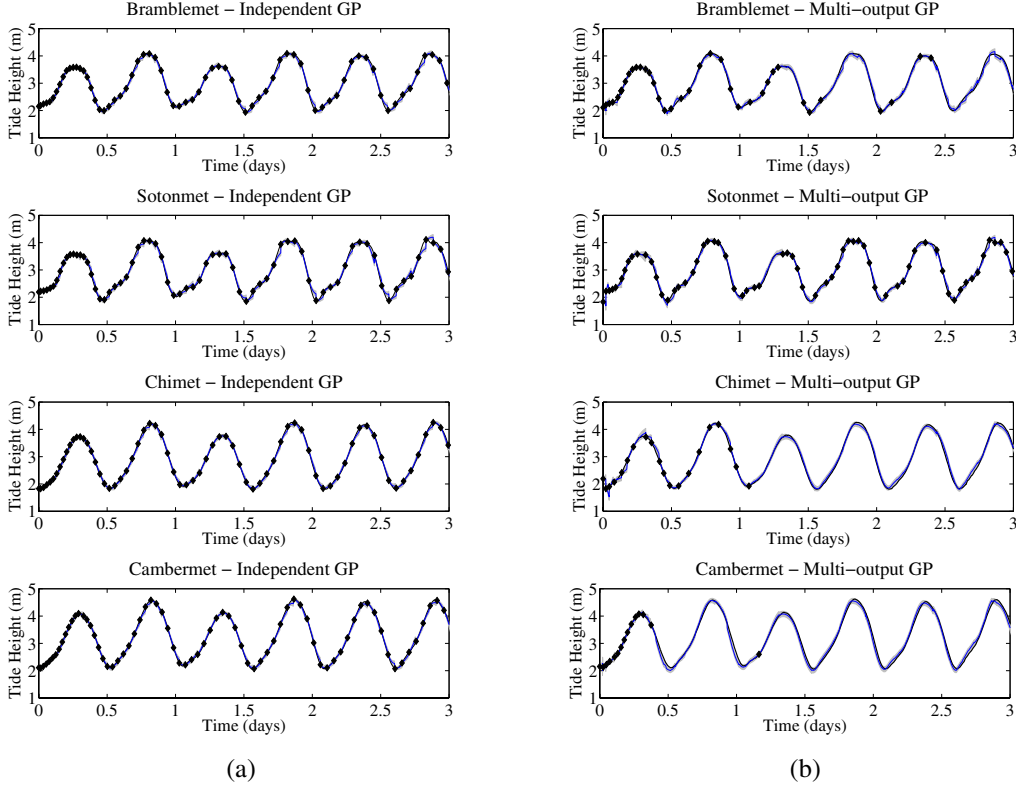


Figure 5: Comparison of active sampling of tide data using (a) independent and (b) multi-output Gaussian processes.

ity of the GP to learn and predict periodic behaviour, and more importantly, because this particular data set contains an interesting period in which extreme weather conditions (a Northerly gale) cause both an unexpectedly low tide and a failure of the wireless connection between the sensor and the shore that prevents our information agent acquiring sensor readings. Air temperature was chosen since it exhibits very different noise and correlation to the tide height measurements, and yet, our multi-output GP formalism is able to provide reliable regression and prediction on both.

### 5.1 Regression and Prediction

Figures 3 and 4 illustrate the efficacy of our GP formalism in this scenario. We plot the sensor readings acquired by the information agent (shown as markers), the mean and standard deviation of the GP prediction (shown as a solid line with the standard deviation shown as shading), and the true fine-grained sensor readings (shown as bold) that were downloaded directly from the sensor (rather than through the web site) after the event. Note that we present just two sensors for reasons of space, but we use readings from all four sensors in order to perform regression.

We consider the performance of our multi-output GP formalism when the Bramblemet sensor drops out at  $t = 1.45$  days. In this case, note that the independent GP predictions quite reasonably predicts that the tide will continue to do

more or less what it has seen before, and predicts the same periodicity it has observed in the past. However, the GP can achieve better results if it is allowed to benefit from the knowledge of the other sensor’s readings during this interval of missing data. Thus, in the case of the multi-output GP, by  $t = 1.45$  days, the GP has successfully determined that the sensors are all very strongly correlated. Hence, when it sees an unexpected low tide in the Chimet sensor data (caused by the strong Northerly wind), these correlations lead it to infer a similarly low tide in the Bramblemet reading, and produces significantly more accurate predictions.

Exactly the same effect is seen in the later predictions of the Chimet tide height, where the multi-output GP predictions use observations from the other sensors to better predict the high tide height at  $t = 2.45$  days. Furthermore, figure 4 shows the air temperature sensor readings where a similar effect is observed. Again, the multi-output GP is able to better predict the missing air temperature readings from the Chimet sensor having learnt the correlation with other sensors, despite the fact that the data set is much noisier and the correlations between sensors are much weaker.

### 5.2 Active Data Selection

We now demonstrate our active data selection algorithm. Using the fine-grained data (downloaded directly from the sensors), we can simulate how our GP would have chosen



Figure 6: Prototype information agent deployed on a PDA and weather sensor incorporating a Wi-Fi access point.

its observations had it been in control. Results from the active selection of observations from all the four tide sensors, are displayed in figure 5. Again, these plots depict dynamic choices; at time  $t$ , the GP must decide when next to observe, and from which sensor, given knowledge only of the observations recorded prior to  $t$ , in an attempt to maintain the uncertainty in tide height below 10cm.

Consider first the independent case shown in figure 5(a), in which separate GPs are used to represent each sensor. Note that a large number of observations are taken initially as the dynamics of the sensor readings are learnt, and then later, a low but constant rate of observation is chosen.

In contrast, for the independent case shown in figure 5(b), the GP is allowed to explicitly represent correlations and delays between the sensors. This data set is notable for the tide heights at the Chimet and Cambermet sensors, which due to tidal flows in the area are slightly delayed relative to the Sotonmet and Bramblemet sensors. Note that after an initial learning phase as the dynamics, correlations, and delays are inferred, the GP chooses to sample predominantly from the undelayed Sotonmet and Bramblemet sensors<sup>1</sup>. Despite no observations at all subsequently being made of the Chimet sensor, the resulting predictions remain remarkably accurate. Consequently only 66 observations are required to keep the uncertainty below the specified tolerance, whereas 127 observations were required in the independent case.

## 6 Conclusions and Future Work

In this paper we have demonstrated the use of a novel iterative formalism of a multi-output Gaussian process to perform information processing on sensor readings acquired from a pervasive sensor network, and shown that with minimal domain knowledge we can perform effective prediction, regression, and active sampling.

Our future work in this area consists of two main areas. First, we intend to investigate the use of correlations between different sensor types (rather than between different

<sup>1</sup>The dynamics of the tide height at the Sotonmet sensor are more complex than the other sensors due to the existence of a ‘young flood stand’ and a ‘double high tide’ in Southampton. For this reason the GP selects Sotonmet as the most informative sensor and samples it most often.

sensors of the same type as we have presented here) to perform regression and prediction, and also to use the probabilistic model represented by the GP to automatically detect sensor failures. Second, to investigate the practical issues of using such information agents within pervasive sensor networks, we are developing prototype stand-alone weather sensors (to be deployed at the University of Southampton) that make their sensor readings available in RDF format and form ad-hoc Wi-Fi connections with information agents deployed on PDA (see figure 6).

## Acknowledgments

This research was undertaken as part of the ALADDIN (Autonomous Learning Agents for Decentralised Data and Information Networks) project and is jointly funded by a BAE Systems and EPSRC strategic partnership (EP/C548051/1). We would like to thank B. Blaydes of the Bramblemet/Chimet Support Group, and W. Heaps of Associated British Ports (ABP) for allowing us access to the weather sensor network, hosting our RDF data, and for providing raw sensor data as required.

## References

- [1] P. Abrahamsen. A review of Gaussian random fields and correlation functions. Technical Report 917, Norwegian Computing Center, Box 114, Blindern, N-0314 Oslo, Norway, 1997. 2nd edition.
- [2] P. Boyle and M. Frean. Dependent Gaussian processes. In *Advances in Neural Information Processing Systems 17*, pages 217–224. The MIT Press, 2005.
- [3] N. A. C. Cressie. *Statistics for spatial data*. John Wiley & Sons, 1991.
- [4] A. Deshpande, C. Guestrin, S. Madden, J. Hellerstein, and W. Hong. Model-driven data acquisition in sensor networks. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases (VLDB 2004)*, pages 588–599, 2004.
- [5] J. K. Hart and K. Martinez. Environmental Sensor Networks: A revolution in the earth system science? *Earth-Science Reviews*, 78:177–191, 2006.
- [6] A. Krause, C. Guestrin, A. Gupta, and J. Kleinberg. Near-optimal sensor placements: Maximizing information while minimizing communication cost. In *Proceedings of the Fifth International Conference on Information Processing in Sensor Networks (IPSN '06)*, pages 2–10, Nashville, Tennessee, USA, 2006.
- [7] R. Murty, A. Gosain, M. Tierney, A. Brody, A. Fahad, J. Bers, and M. Welsh. Citysense: A vision for an urban-scale wireless networking testbed. Technical Report TR-13-07, Harvard University, September 2007.
- [8] M. Osborne and S. J. Roberts. Gaussian processes for prediction. Technical Report PARG-07-01. Available at [www.robots.ox.ac.uk/~parg/publications.html](http://www.robots.ox.ac.uk/~parg/publications.html), University of Oxford, September 2007.
- [9] J. Pinheiro and D. Bates. Unconstrained parameterizations for variance-covariance matrices. *Statistics and Computing*, 6:289–296, 1996.
- [10] C. E. Rasmussen and Z. Ghahramani. Bayesian Monte Carlo. In *Advances in Neural Information Processing Systems 15*, pages 489–496. The MIT Press, 2003.
- [11] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.
- [12] Y. W. Teh, M. Seeger, and M. I. Jordan. Semiparametric latent factor models. In *Proceedings of the Conference on Artificial Intelligence and Statistics*, pages 333–340, 2005.