THE UNIVERSITY OF
SYDNEY

Autonomous Exploration of Large-Scale Natural Environments

Asher Bender, BE (Hons 1) BComm

A thesis submitted in fulfillment
of the requirements of the degree of
Doctor of Philosophy

THE UNIVERSITY OF
SYDNEY

Australian Centre for Field Robotics
School of Aerospace, Mechanical and Mechatronic Engineering
The University of Sydney

March 2013

Declaration

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the University or other institute of higher learning, except where due acknowledgement has been made in the text.

Asher Bender

28 March 2013

Abstract

Asher Bender
The University of Sydney

Doctor of Philosophy
March 2013

Autonomous Exploration of Large-Scale Natural Environments

This thesis addresses issues which arise when using robotic platforms to explore large-scale, natural environments. Two main problems are identified: the volume of data collected by autonomous platforms and the complexity of planning surveys in large environments.

Autonomous platforms are able to rapidly accumulate large data sets. The volume of data that must be processed is often too large for human experts to analyse exhaustively in a practical amount of time or in a cost-effective manner. This burden can create a bottleneck in the process of converting observations into scientifically relevant data.

Although autonomous platforms can collect precisely navigated, high-resolution data, while operating beyond the limits of human endurance and safety, they are typically limited by finite battery capacities, data storage and computational resources. Additionally, the projects which support these vehicles have limited budgets and time frames to complete their objectives. These constraints make it impractical to sample the environment exhaustively. To use the limited resources effectively, trajectories which maximise the amount of information gathered from the environment must be designed.

This thesis address these problems. Three primary contributions are presented: a new classifier designed to accept probabilistic training targets rather than discrete training targets; a semi-autonomous pipeline for creating models of the environment; and an offline method for autonomously planning surveys.

While the thesis specifically focuses on mapping and exploring marine habitats with an autonomous underwater vehicle (AUV), the research applies equally to other applications such as aerial and terrestrial environmental monitoring and planetary exploration. Environmental models are established by learning the correlation between data extracted from a digital elevation model (DEM) of the seafloor and habitat categories derived from in-situ images. The DEM of the seafloor is collected using ship-borne multibeam sonar and the in-situ images are collected using an AUV.

In principle, the AUV imagery could be classified into habitats by a human expert. However, like other robotic platforms, the volume of data collected by an AUV is often too large for human experts to analyse exhaustively in a practical amount of time or in a cost-effective manner. To reduce labour requirements, human classification of the imagery is replaced by unsupervised clustering with a variational Dirichlet process (VDP) model. Rather than manually classifying a large volume of individual images, human experts are required to ensure a relatively small set of clusters, produced by the VDP model, represent valid habitat 'proxies'.

To fully utilise all of the information contained in the probabilistic data generated by the VDP, a novel Gaussian process (GP) classifier capable of accepting probabilistic training targets is proposed. The combination of the VDP clustering algorithm and the new classifier form the core of a novel, semi-autonomous habitat mapping pipeline. The proposed pipeline is demonstrated on real benthic data where a large habitat map is created with minimal human input.

A solution for planning informative surveys offline is also proposed. The planning framework is designed to select the optimal location to place a prespecified survey template. Assessing every possible survey placement exhaustively is a large problem. Instead, a functional representation of the survey utility is built using a small set of training placements. This enables the utility of a candidate survey placement to be queried in a continuous space and in arbitrary locations. The prohibitive cost of evaluating every survey placement has been replaced with the tractable requirement of calculating data to support training and inference in the utility model. This novel approach is shown to scale up to very large, realistic planning problems.

Acknowledgements

I am deeply grateful for the support provided by my supervisors Stefan Williams and Oscar Pizarro. To Stef, your enthusiasm for teaching and learning inspired me to embark on my adventure down the rabbit hole. Thank you for providing me the opportunity to do unique research in an exciting field. Your patience, support, encouragement and advice (despite a sanity defying schedule) have always helped me improve. To Oscar, your keen eye for detail and clear approach to problem solving saved me countless hours. Thank you for keeping research light-hearted and the stimulating discussions outside of work.

In the course of my research I have been privileged to work with many others who have contributed towards the success of the marine robotics program at the Australian Centre for Field Robotics (ACFR). To Mike Jakuba, your support early in my research was invaluable. Thank you for also encouraging me to get my hands dirty (and my toes muddy). Nothing would have happened without our talented technical staff. During the years, Ritesh Lal, Jeremy Randle, Duncan Mercer, George Powell, Andrew Durrant and Christian Lees have been great company and provided me with invaluable technical advice. Their skill in keeping robots functional makes our research possible.

Thanks to the eccentric 'upstairs crew' for generating hilarious and stimulating discussions over lunch and coffee. In particular, thanks to my cubicle mates Daniel Steinberg, Ariell Friedman and John Vial. Dan, you made doing a PhD look so easy and I would have been lost without your advice. Ari, thanks for keeping me level headed and my feet on the ground. John, your cheery disposition made it difficult to have a bad day. To all, thanks for putting up with my distractions and making me look forward to coming to work.

Lastly, I owe my greatest thanks to my family. To Mum and Dad, thank you for a lifetime of unconditional love and support. From my earliest years you always facilitated my interests and education. Without your support this would not have been possible. Dad, thank you for putting aside more entertaining literature to proof-read my thesis. I know it is a hard job! To Hannah, you have always been an inspirational role model. Your wise PhD advice has kept me on track and realistic. Thank you for seeing it through with me. Finally, to Elina, thank you for your love and support throughout the years, especially during my thesis induced mania.

For my family.

Contents

List of Figures

List of Tables

List of Symbols

$\mathbf{0}$	A matrix of zeros.
$\boldsymbol{\Sigma}$	Covariance of a multivariate normal distribution.
D	Dimensionality of a data space.
$\mathcal{E}$	An experiment.
$\mathbb{E}[\cdot]$	Expected value of a random variable.
$\mathbb{V}[\cdot]$	Expected variance of a random variable.
$\mathcal{GP}$	Gaussian process.
$\mathbf{f}$	Latent function values of a Gaussian process.
$m(\cdot)$	Posterior mean function of a Gaussian process.
$\mathrm{Cov}(\cdot)$	Posterior covariance function of a Gaussian process.
$\mathrm{H}[\cdot]$	Entropy of a distribution.
$\mathbf{I}$	An identity matrix (a square matrix with ones on the main diagonal and zeros elsewhere).

$\mathbf{K}$	Covariance function of a Gaussian process matrix. $\mathbf{K}$ is short hand for $\mathbf{K}(\mathbf{X}, \mathbf{X})$ - the covariance function evaluated at the training inputs.
$KL(P\|\|Q)$	KullbackLeibler divergence between the random variables P and Q.
ϕ	linear-logistic sigmoid function.
$\mathcal{N}$	Normal distribution.
Φ	normal cumulative distribution function.
$\mathbb{R}$	the set of real numbers.
θ	Hyperparameters (free parameters) of the covariance function.
U	Utility.
$\mathbf{X}$	A matrix of multiple D-dimensional data or training inputs.
$\mathbf{x}_*$	A D-dimensional query input.
$\mathbf{x}$	A vector of D-dimensional data or a training input.
ξ	Space of possible experiments.
$\mathbf{X}_*$	A matrix of multiple D-dimensional query inputs.
$\mathbf{y}$	A vector of training targets which may be either continuous (regression) or discrete (classification).
y	A training target which may be either continuous (regression) or discrete (classification).

Acronyms

ACFR	Australian Centre for Field Robotics.
AUV	autonomous underwater vehicle.
BPI	bathymetric position index.
DEM	digital elevation model.
DPV	diver propulsion vehicle.
DVL	Doppler velocity log.
GA	Geoscience Australia.
GMM	Gaussian mixture model.
GP	Gaussian process.
KLD	Kullback-Leibler divergence.
LBP	Local Binary Pattern.
LIDAR	light detection and ranging.
LOO	leave one out.
MBES	multibeam echo-sounder.

MCMC	Markov chain Monte Carlo.
MDP	Markov decision process.
MSE	mean squared error.
OEH	Office of Environment and Heritage.
OVA	one-vs-all.
PCA	principal component analysis.
PDF	probability density function.
PLSC	probabilistic least squares classifier.
POMDP	partially observable Markov decision process.
PTLSC	probabilistic targets least squares classifier.
ROV	remotely operated vehicle.
SLAM	simultaneous localisation and mapping.
SONAR	sound navigation and ranging.
SVM	support vector machine.
TAFI	Tasmanian Aquaculture and Fisheries Institute.
TRI	terrain ruggedness index.
USBL	ultrashort baseline.
USyd	University of Sydney.
VDP	variational Dirichlet process.

Chapter 1

Introduction

1.1 Background and Motivation

Images collected from the benthos are a rich source of information. Ecologists can use the data to classify habitats, characterise species abundance and for monitoring change. Commercial applications such as prospecting for resources and pipe-laying can use this data to reduce their environmental impact and avoid difficult terrain.

Autonomous underwater vehicles (AUVs) equipped with an optical imaging package are flexible tools for surveying the benthos. By maintaining a constant altitude close to the seafloor, an AUV can provide consistent illumination and collect high-quality images. The position and orientation of the data set can be resolved accurately by reconstructing the survey path of the vehicle using onboard sensors and terrain-based navigation. In addition to providing high-quality, geo-referenced data, AUVs also have less operational limitations. They are decoupled from surface motions, their ability to operate is depth-independent and they can follow rugged terrain.

Although AUVs are a flexible tool in oceanography, they have limitations. Bottom-following AUVs use instruments with a narrow field of view to collect high-resolution data directly under the vehicle at the cost of coverage. The extent of these dives is also limited by the energy consumption and capacity of the vehicle. Multiple dives can be used to increase coverage but project budgets will restrict the number of deployments.

In the past decade, remote sensing techniques have improved the extent and resolution of broad-scale digital elevation models (DEMs). For instance ship-borne multibeam echo-sounder (MBES) systems allow oceanographers to develop broad-scale, high-resolution DEMs of the seafloor at a relatively low cost. By intelligently combining this data with in-situ observations of the seafloor it is possible to create a useful habitat model of the distribution of benthic flora and substrate. Performing inference in these models allows the distribution of habitats to be inferred over the extent of the broad-scale DEM. These predictive models fill in the gap created by the narrow coverage of in-situ sampling techniques.

The first half of this thesis investigates how ship-borne MBES data and optical data collected from an AUV can be combined to generate habitat maps over large regions. The primary aim of the research is to process large benthic data sets with minimal human intervention. The quality of a habitat map ultimately depends on the information contained within the observed data. Once the ship-borne, broad-scale data has been collected it is considered a fixed resource.

Establishing and improving the habitat model is only made possible by collecting in-situ observations of the seafloor. There are limitations to this process. In the case of an AUV, limited battery capacity and the cost of the support ship constrains the extent of the data collected. Once a dive has commenced, it is also difficult to change the objectives of the dive. Water has a high attenuation coefficient making electromagnetic communication ineffective. Acoustic modems can propagate signals but are limited to low data rates. The data rates acoustic modems provide are often insufficient to relay the sort of information human experts require to evaluate whether the dive is fulfilling its scientific objectives.

Given the practical constraints of operating an AUV, complete coverage is impractical and operators must commit to a limited number of pre-programmed dives. To use the limited resources most effectively, scientists and AUV operators have to optimise their survey plans to maximise the quality of a habitat map. This is the focus of the second half of this thesis.

1.2 Thesis Contributions

This thesis is concerned with producing benthic habitat maps efficiently. Specific contributions made in this thesis are aimed at reducing the amount of human supervision required to produce a habitat map and plan additional deployments in large-scale environments. The principal contributions of this thesis are as follows:

- A Gaussian process (GP) based classifier capable of accepting probabilistic training data rather than discrete class assignments. This model is better equipped to preserve the probabilistic information generated by supervisory probabilistic models. As a result the new model is more amenable to being used in conjunction with other unsupervised methods capable of processing the training data.

- A semi-autonomous data processing pipeline for creating benthic habitat maps. The proposed pipeline is designed to reduce the labour requirements of labelling the large volumes of in-situ data collected by autonomous platforms. The pipeline is demonstrated on real marine data where it is able to generate a habitat map for a large environment with minimal human input.

- An offline method for autonomously planning benthic surveys. The proposed framework is designed to select the optimal location to place a prespecified survey template. By building a functional representation of the survey utility, the utility of a candidate survey placement can be queried in a continuous space and in arbitrary locations. This novel approach is shown to scale up to very large, realistic planning problems.

- Utility functions for evaluating the value of a candidate survey are proposed. The Kullback-Leibler divergence (KLD) is adopted as a novel planning heuristic for scenarios where no prior observations of the environment exist. In cases where prior observations of the environment are available, the A-optimal criteria is recommended. Justification for using this utility function in a GP classification context is provided.

1.3 Thesis Overview

Chapter 2 establishes GP theory and nomenclature. This review provides a context for the research contained in subsequent chapters of the thesis.

Chapter 3 addresses the labour requirements of classifying large data sets. A reduction in the amount of human effort required to label a data set is achieved by clustering the data with an unsupervised variational Dirichlet process (VDP) model [46]. To make full use of the subtle information conveyed by the cluster probabilities, a novel GP-based classifier is proposed. Rather than using discrete class assignments during training, probabilistic class assignments are used. The model is demonstrated on controlled experiments using simulated data.

Chapter 4 discusses current approaches used to create benthic habitat maps. The advantages and new problems which arise from using an AUV to collect data for benthic habitat mapping are discussed. A semi-autonomous method for producing benthic habitat maps is proposed. A reduction in the human labour required to label the AUV imagery is achieved by applying the method proposed in [91] to cluster the AUV imagery. The model proposed in Chapter 3 is used to correlate the AUV image clusters with features extracted from ship-borne MBES data. This semi-autonomous data processing pipeline is demonstrated on data collected from the southeastern Tasman peninsula in Tasmania, Australia. The data set includes almost $100km^2$ of ship-borne MBES bathymetry and 80,000 images of the seafloor collected by an ocean-going AUV.

Chapter 5 investigates benthic habitat mapping in an offline context. Whilst real-time exploration is theoretically possible, AUVs do not often possess the necessary software architecture or computing resources to analyse data in real-time. This chapter proposes a method for recommending informative survey designs in large environments. To support exploration when no prior observations exist, the KLD is used as a novel planning heuristic. The planning method is designed to select the most informative location to place a prespecified survey template. The method is demonstrated in a large-scale marine environment using real data.

Chapter 6 extends the proposed planning framework to scenarios where prior in-situ observations of the environment exist. By exploiting a pre-existing habitat model, the information contained in prior observations of the environment can be used to guide exploration efforts. Closing the loop between decision making, observations and modelling allows a planning algorithm to directly optimise a statistical property of the habitat model. This chapter recommends an A-optimal utility function. This utility function can be derived from information theoretic principles for GPs and has found practical use in the robotics literature. The planning framework is demonstrated in a marine environment where it was used to plan deployments.

Chapter 7 presents conclusions and discusses future work for building on this thesis.

Chapter 2

Gaussian Processes

Scientists are often interested in characterising naturally occuring phenomena in a region of interest. This information can be used to provide policy makers with the necessary data to make informed decisions about managing the region. Collecting accurate and well distributed observations of naturally occuring phenomena is costly and often labour-intensive. As a result, data sets typically include sparse and unevenly sampled observations. To draw principled conclusions about the gathered data, techniques for predicting values at unobserved locations are often required. The generation of environmental models for the purpose of making predictions in unobserved locations is a founding concept in this thesis.

Gaussian processes (GPs) are a flexible, non-parametric model which offer an elegant framework for solving regression and classification problems. Freed from strict modelling assumptions, GPs create an effective Bayesian black-box model for inference which allows the data to 'speak for itself'. This flexibility makes GPs ideal for modelling correlated environmental data.

The remainder of this chapter is arranged as follows. Section 2.1 describes how continuous data is modelled in regression problems using a GP. Section 2.2 describes how discrete data is modelled in classification problems using a GP. Section 2.3 summarises and concludes the chapter.

2.1 Gaussian Process Regression

A GP is a specific example of a stochastic process. A stochastic process defines a probability distribution over random functions. A GP is a collection of random variables, any finite number of which have a joint Gaussian distribution [75]. If one loosely considers a function as an infinitely long vector, then a GP can be thought of as a multivariate Gaussian distribution embedded in a space of infinite dimensions.

For an underlying process $f(\mathbf{x})$, if it is assumed that the value of the function at the location $\mathbf{x}$ is randomly distributed according to a GP then

$$f(\mathbf{x}) \sim \mathcal{GP}\left(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\right) \tag{2.1}$$

where the mean function is given as

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

and the covariance function is given by

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

Although a GP defines a probability distribution over an infinite space of functions Equation (2.1), it is not necessary to work directly in this infinite space.

Assuming the underlying function can be modelled by a GP then, by definition, any finite collection of observations of the function can be modelled using a multivariate Gaussian. Since the process is assumed to be stochastic, it is not possible to sample the function directly. Instead, it is only possible to observe noisy realisations of the underlying function according to

$$\mathbf{y} = f(\mathbf{X}) + \epsilon$$

where $\mathbf{y}$ are noisy observations $(y_1, y_2, ..., y_n)^T \in \mathbb{R}$ of the underlying function, $\mathbf{f} = (f(\mathbf{x}_1), f(\mathbf{x}_2), ..., f(\mathbf{x}_n))^T$, at the input locations, $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n)^T \in \mathbb{R}^D$. The

observations of the underlying function are assumed to be perturbed by independent identically distributed Gaussian noise, ϵ, with a variance of σ_n^2.

Assuming a zero-mean, the prior on the latent (hidden) function values at the observed input locations is given by

$$p(\mathbf{f} \mid \mathbf{X}, \theta) = \mathcal{N}(\mathbf{f} \mid \mathbf{0}, \mathrm{Cov}). \tag{2.2}$$

where Cov is the covariance function and θ are parameters of the covariance function. Since the observations are perturbed by Gaussian white noise, the likelihood is given by

$$p(\mathbf{y} \mid \mathbf{f}) = \mathcal{N}\left(\mathbf{y} \mid \mathbf{f}, \sigma_n^2 \mathbf{I}\right). \tag{2.3}$$

The zero-mean assumption is common within the GP literature and implies that the mean value of random functions sampled from the GP prior and evaluated at a particular location $\mathbf{X}$ would be zero. It is important to note that although the prior process is restricted to a zero mean, this is not true of the posterior process (see Equation (2.7)). This assumption is not required and can easily be relaxed by specifying a mean function. The zero-mean assumption is often adopted for simplicity and convenience.

The covariance function produces a covariance matrix which models the expected correlations between pairs of points drawn from the GP. Any function which takes pairs of points from the input space and produces a symmetric and positive definite covariance matrix is a valid covariance function. Choosing this function is an important practical consideration when using GPs. An appropriately chosen covariance function will model the properties of the functions embedded in the GP distribution.

If the signal noise, σ_n^2, and hyperparameters θ are known, the prior Equation (2.2) and likelihood functions Equation (2.4) are correct and it is possible to make predictions. These values are seldom known in advance and are more commonly inferred from the data.

2.1.1 Training

Before it is possible to make predictions or even consider Equation (2.2) to be a sensible prior and Equation (2.4) to be a sensible likelihood, the free parameters of the model must be properly specified. The parameters of the covariance function, θ and the signal noise σ_n^2 must be selected to reflect the space which the underlying function resides in. Since these parameters control the distribution over the latent function, they are known as hyperparameters. Determining the optimal value for these hyperparameters is known as model selection.

Since GPs are fully probabilistic models, Bayesian model selection can be employed to determine values for the hyperparameters which represent a trade-off between model complexity and data-fit. In a fully Bayesian treatment, a prior is placed over the hyperparameters and Bayes' rule is used to calculate a posterior. This process requires integration over the hyperparameters and is usually analytically intractable. Markov Chain Monte Carlo methods can be used to approximate the required integrals [63, 74] but are computationally expensive methods. An alternative, computationally convenient method is maximum likelihood approximation.

The marginal likelihood (or model evidence) is a measure of how well the model explains the observed data. It is possible to perform model selection by maximising the marginal likelihood with respect to the hyperparameters. The hyperparameters which maximise the marginal likelihood are the hyperparameters which produce the model capable of explaining the observed data the best. The marginal likelihood is given by marginalising over the latent function variables, $\mathbf{f}$, of the likelihood times the prior

$$\begin{aligned} p(\mathbf{y} \mid \mathbf{X}, \theta) &= \int p(\mathbf{y} \mid \mathbf{f}, \mathbf{X})\, p(\mathbf{f} \mid \mathbf{X}, \theta)\; d\mathbf{f} \\ &= \int \mathcal{N}\left(\mathbf{y} \;\middle|\; \mathbf{f}, \sigma_n^2 \mathbf{I}\right) \mathcal{N}\left(\mathbf{f} \mid \mathbf{0}, \mathrm{Cov}\right)\; d\mathbf{f}. \end{aligned}$$

Since the likelihood and prior are both multivariate normal distributions, the marginal

likelihood can be calculated analytically

$$p(\mathbf{y} \mid \mathbf{X}, \theta) = \mathcal{N}\left(\mathbf{y} \mid \mathbf{0}, \mathrm{Cov} + \sigma_n^2 \mathbf{I}\right). \tag{2.4}$$

The marginal likelihood, Equation (2.4), can be maximised by minimising the negative log marginal likelihood with respect to the hyperparameters:

$$\log p(\mathbf{y} \mid \mathbf{X}, \theta) = -\frac{1}{2}\mathbf{y}^T \left(\mathrm{Cov} + \sigma_n^2 \mathbf{I}\right)^{-1} \mathbf{y} - \frac{1}{2}\log\left|\mathrm{Cov} + \sigma_n^2 \mathbf{I}\right| - \frac{n}{2}\log 2\pi \tag{2.5}$$

where the covariance function Cov is dependent on the parameters θ. The first term of Equation (2.5) models data fit. The second term is a complexity penalty and the final term is a normalisation constant. Optimising the marginal likelihood is a useful objective function that is resilient to over-fitting as there is a trade-off between data fit and model complexity. Whilst the complexity penalty in Equation (2.5) reduces the chances of over-fitting, as more flexibility is permitted in the model, maximising the marginal likelihood becomes increasingly prone to over-fitting. For complex models with a large number of hyperparameters, fully Bayesian model selection mechanisms such as MCMC may be more robust [63].

Optimising the negative log marginal likelihood can be done using a multivariate optimisation algorithm such as conjugate gradient descent. In practice these algorithms are effective but minimising the negative log marginal likelihood is a non-convex optimisation problem. Care must be taken to ensure the hyperparameters do not represent a local maximum in the marginal likelihood, particularly when data is scarce. For instance, two plausible models for a small data set might be low noise and a small correlation length scale or high noise and a large correlation length scale. This problem can be avoided by performing several optimisations, starting from different locations in the parameter space, and selecting the hyperparameters which produce the highest marginal likelihood. Once the hyperparameters have been properly specified, it is possible to perform inference.

2.1.2 Inference

It is possible to perform inference at arbitrary query locations, $\mathbf{X}_* = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m)^T \in \mathbb{R}^D$, by exploiting the properties of the GP. Recall that any number of observations sampled from a GP will be jointly Gaussian. Since the unobserved query targets belong to the same GP as the observed training data, they will be jointly Gaussian. This joint distribution, under the prior, can be written as a partitioned multivariate Gaussian

$$\begin{aligned} p(\mathbf{y}, \mathbf{f}_* \mid \mathbf{X}, \mathbf{X}_*, \theta) &= \mathcal{N}\left(\mathbf{y}, \mathbf{f}_* \,\middle|\, \mathbf{0}, \mathrm{Cov}^+\right) \\ &= \mathcal{N}\left(\begin{bmatrix}\mathbf{y} \\ \mathbf{f}_*\end{bmatrix} \,\middle|\, \mathbf{0}, \begin{bmatrix}\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I & \mathbf{K}(\mathbf{y}, \mathbf{X}_*) \\ \mathbf{K}(\mathbf{X}_*, \mathbf{X}) & \mathbf{K}(\mathbf{X}_*, \mathbf{X}_*)\end{bmatrix}\right) \end{aligned} \tag{2.6}$$

where the joint covariance matrix, Cov^+, is a block matrix, partitioned such that $\mathbf{K}(\mathbf{X}, \mathbf{X})$ is the covariance for the training data, $\mathbf{K}(\mathbf{X}, \mathbf{X}_*)$ is the cross covariance between the training data and the test data and $\mathbf{K}(\mathbf{X}_*, \mathbf{X}_*)$ is the prior covariance matrix for the test data.

The predictive distribution is given by the conditional,

$$p(\mathbf{f}_* \mid \mathbf{X}, \mathbf{y}, \mathbf{X}_*, \theta) = \mathcal{N}\left(\mathbb{E}[\mathbf{f}_*], \mathbb{V}[\mathbf{f}_*]\right).$$

Closed-form expressions for the predictive mean and variance exist due to convenient properties of the multivariate Gaussian distribution. If two sets of variables are jointly Gaussian, and one set is conditioned on another, the resultant distribution is also a Gaussian. Using this property, the predictive mean and covariance are given by Equation (2.7) and Equation (2.8) respectively.

$$\mathbb{E}[\mathbf{f}_*] = \mathbf{K}(\mathbf{X}_*, \mathbf{X})\left[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} \mathbf{y} \tag{2.7}$$

$$\mathbb{V}[\mathbf{f}_*] = \mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) - \mathbf{K}(\mathbf{X}_*, \mathbf{X})\left[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} \mathbf{K}(\mathbf{X}, \mathbf{X}_*) \tag{2.8}$$

Note that although a zero-mean prior is assumed, the predictive distribution defined by Equation (2.7) and Equation (2.8) produces a GP with a specific, non-zero mean

function. The predictive covariance only depends on the correlations induced by the covariance function at the training and test input locations. Although the predictive covariance does not depend on observed values (training inputs) explicitly, the observed values are considered during model selection.

2.1.2.1 Covariance Functions

In this thesis the squared exponential covariance function is used:

$$\mathrm{Cov}\left(\mathbf{x}_p, \mathbf{x}_q\right) = \sigma_f^2 \exp\left(-\frac{1}{2}\left(\mathbf{x}_p - \mathbf{x}_q\right)^T L^{-1}\left(\mathbf{x}_p - \mathbf{x}_q\right)\right). \tag{2.9}$$

In this covariance function, distant inputs have a covariance that tends towards zero and close inputs have a covariance that approaches σ_f^2. The degree to which a pair of function values is correlated depends on free parameters within the covariance function. For the squared exponential covariance function and additive noise, these free parameters, known as hyperparameters, are $\theta = \{L, \sigma_f^2, \sigma_n^2\}^T$. The characteristic length scale L corresponds to a measure of distance across all dimensions where two function values are expected to become uncorrelated. The variance of the underlying function is described by σ_f^2. Additive Gaussian noise on the observations is modelled by the variance term σ_n^2.

2.2 Gaussian Process Classification

In binary classification, the target values are assumed to be Bernoulli distributed independent random variables, given the latent function, such that $y_i \in \{-1, +1\}$. To allow for this, the latent function must be constrained to lie on the interval $[0, 1]$. This allows the probability of class membership $p(\mathbf{y} = 1 \mid \mathbf{X})$ to have a valid probabilistic interpretation. Where this differs significantly from probabilistic regression is that the the likelihood, $p(\mathbf{y} \mid \mathbf{X})$, is non-Gaussian. This effect trickles down during inference.

When invoking Bayes rule

$$p(\mathbf{f} \mid \mathbf{X}, \mathbf{y}) = \frac{p(\mathbf{y} \mid \mathbf{f})\, p(\mathbf{f} \mid \mathbf{X})}{p(\mathbf{y} \mid \mathbf{X})} \tag{2.10}$$

the posterior over the latent values, $p(\mathbf{f} \mid \mathbf{X}, \mathbf{y})$, becomes non-Gaussian thanks to the non-Gaussian likelihood. This departure from conjugacy makes classification using GPs analytically intractable. Since an analytic solution for the predictive class membership probability does not exist, approximation methods are required. A number of methods have been proposed to allow inference.

2.2.1 Approximations

Markov chain Monte Carlo (MCMC) [63] produces a numerical approximation of the predicted latent function values and the predictive class membership probabilities. In the limit of a large number of samples, MCMC methods are guaranteed to produce the true distributions. The MCMC solution is considered the gold standard and is often used as a ground truth for comparison purposes. The quality of the approximation comes at a very high computational price.

An alternative approach is to make a Gaussian approximation to the posterior distribution over the latent function values Equation (2.10). Several approximations have been proposed in the literature including variational methods [26], the Laplace approximation [97], expectation propagation [61] and probabilistic least squares classification [75].

Of these approximation methods, expectation propagation is considered state of the art and is the method of choice when compared to the reference MCMC solution [47, 65]. An interesting outcome of comparing the approximation methods is that the quality of the approximation to the marginal likelihood, which is important for reliable model selection and creating dependable class probabilities, is not necessarily critical for producing accurate class predictions. All of the approximations mentioned are capable of producing similar classification error rates.

2.2.2 Probabilistic Least Squares Classification

The probabilistic least squares classifier (PLSC) is a simple method of producing a Gaussian approximation to the posterior. Rather than assuming the latent function is constrained to lie on the interval $[0, 1]$, the PLSC returns to the regression assumption that the latent function can reside in $\mathbb{R}$. Approaching classification as a regression problem is not limited to GPs and has been used in other models where it is known, rather appropriately, as label regression or least squares classification [65, 77]. The model has been used effectively in several machine learning and robotics applications [41, 67, 76, 78].

Treating classification as a regression problem implies the odd assumption that the observed discrete targets have a Gaussian noise model. Although this assumption is counter intuitive, in the large limit, the predictive class probabilities will converge on the true value [75]. For a finite collection of data this is not guaranteed and the latent function can take on any real value. To enforce a probabilistic interpretation of the latent function, it must be passed through a function which squashes any real value onto the interval $(0, 1)$. The squashed output represents a valid predictive probability.

The PLSC can be thought of as another way of producing a Gaussian approximation to the posterior distribution over the latent function values. The attraction of this method is that, since the problem is cast as a regression problem, the conjugacy of GP regression is maintained and the GP principles in Section 2.1 can be applied. This makes the PLSC less computationally intensive than the other approximation methods and simpler to implement.

After training the regression model on the class labels, the only additional step required is to train the squashing function to ensure the predictive output are sensible and valid probabilities. Any sigmoid function can be used to squash the latent function. One function for performing the squashing is the parameterised cumulative Gaussian sigmoid function [72, 75],

$$p(\mathbf{y}_i \mid \mathbf{X}_i, \mathbf{y}_i, \theta) = \Phi\left(\frac{\mathbf{y}_i\left(\alpha\mu_i + \beta\right)}{\sqrt{1 + \alpha^2\sigma_i^2}}\right) \tag{2.11}$$

where Φ is the standard normal cumulative distribution function and is essentially a linearly transformed error function. In the parameterised cumulative Gaussian sigmoid function, the posterior variance over the latent function is used to ensure that the probability tends towards 0.5 as uncertainty in the latent function grows. The two tunable parameters, α and β, must be determined. They can be optimised using a leave one out (LOO) cross-validation procedure where the sum of the log LOO probabilities

$$L_{LOO}\left(\mathbf{X}, \mathbf{y}, \theta\right) = \sum_{i=1}^{n} \log p(\mathbf{y}_i \mid \mathbf{X}, \mathbf{y}_{-i}, \theta) \tag{2.12}$$

is maximised with respect to α and β. The LOO predictive mean and variance are given as

$$\mu_i = \mathbf{y}_i - \frac{[\mathbf{K}^{-1}\mathbf{y}]_{-i}}{[\mathbf{K}^{-1}]_{-ii}} \tag{2.13}$$

$$\sigma_i^2 = \frac{1}{[\mathbf{K}^{-1}]_{-ii}} \tag{2.14}$$

where the subscript $-i$ refers to the element that is removed from the target data and $-ii$ refers to the row and column that is removed from the covariance matrix. Once the sigmoid hyperparameters have been determined, inference can be performed.

2.2.3 Multiclass Classification

Standard GP classification models can only solve binary classification problems. The Laplace approximation [75, 97] and variational approximation [27] have been extended to multiple classes. Other approaches perform multiclass classification by representing each class as a task [9] and using dependent GPs to represent the relations between the different 'tasks' [1]. A simple approach to extend a binary classifier to multiple classes is to use the one-vs-all (OVA) scheme. In OVA classification a binary classifier is trained to classify one class against all others [77]. This is done for each class. Predictions are made by performing inference in each model and combining and normalising the output of each classifier into a single multinomial.

2.3 Summary

This chapter has introduced basic concepts and terminology required to understand GP models. GP models provide a general framework for performing non-parametric regression and classification and are well suited to modelling environmental data. This chapter began by establishing some basic terminology and concepts for training and performing inference in GP regression problems. The chapter concluded by discussing various approximation methods for performing classification. This overview provides a context for research presented in the following chapters.

Chapter 3

Classification with Probabilistic Targets

Robots are often deployed to gather information about the environment. Data collected from multiple platforms or sensors can be combined to produce unique representations of the environment which cannot be obtained from a single deployment or sensor. However, the volume of data collected by modern robotic platforms is often too large for human experts to analyse exhaustively in a practical amount of time or in a cost-effective manner. Under these circumstances it is not possible to efficiently deploy *supervised* machine learning algorithms which rely on processed training data.

In this chapter an unsupervised probabilistic clustering algorithm is used to autonomously create training data for a Gaussian process (GP) classifier. After human approval the training data can be used for classification. This semi-autonomous method allows efficient unsupervised algorithms to categorise large amounts of data and human domain knowledge to be incorporated into the result. Section 3.2 proposes a novel GP classifier designed to preserve all of the information provided by the unsupervised algorithm. This is done by performing classification with probabilistic training targets instead of discrete training targets. The method is demonstrated using a simple one-dimensional example. Section 3.3 investigates the behaviour of the proposed model using a simple two-dimensional example. Finally, Section 3.4 provides concluding remarks.

3.1 Motivation

The objective of classification is to assign input vectors to one of several known categories. Supervised machine learning algorithms achieve this goal by using training data to model the relationship between observed input and target vectors. The intention of establishing a classification model is to make it possible to infer the category at arbitrary locations in the input vector space. Since the input vector is the independent explanatory variable, it is often readily available and does not require intensive analysis to be converted into a usable form.

The category at each input training location must be observed. Given that this data is the dependent variable and the subject of the model, it is likely to be more scarce or difficult and expensive to collect than the independent variable. A reason for modelling the data might be that observing the value of the target variable at the training input locations is difficult or expensive. Another reason might be that producing a useful categorisation of the data requires labour-intensive expert analysis.

A labour-intensive example of processing large data sets is environmental monitoring. It is often possible to observe broad areas of the environment using a cheap, low-resolution sensing modality such as satellite imaging. Alone, this data is unlikely to be sufficient to fulfil modelling objectives such as species distribution estimation. However the data can often be used as a proxy for more expensive, high-resolution observation methods such as gathering in-situ observations through field experiments. Converting data observed in the field into scientifically relevant measures will likely rely on the knowledge of experts such as biologists.

To reduce labour requirements it might be possible to forego human analysis and summarise the field data using an unsupervised clustering model. Many unsupervised models are driven by the density of the data. For instance, the assumption underpinning the efficacy of unsupervised clustering is that distinct modes within the density of the data represent meaningful categorisations. The position advocated in this thesis is that this assumption is correct, when unsupervised clustering is carefully applied.

Labelling the field data autonomously reduces the labour requirement of producing training data. It does not fulfil the original objective of determining the relationship between the low-resolution independent variables and the autonomously derived labels. This is the role of supervised classification. If a probabilistic model has categorised the training data, the probability of class membership is available for each observation. This gives rise to an unconventional classification problem.

Rather than discarding the class probabilities and relying solely on the class labels, this chapter proposes a novel GP classifier designed to accept class probabilities as training inputs. In particular, a novel and intuitive extension to the probabilistic least squares classifier (PLSC) (see Section 2.2.2) is proposed. The new model is called the probabilistic targets least squares classifier (PTLSC). The intention is to preserve all probabilistic information conveyed by the unsupervised model.

3.2 Illustrative Example

To illustrate the PTLSC, the one-dimensional example given in chapter 3 of [75] is adapted. The data consists of 500 observations randomly drawn from a mixture of three Gaussians. Each of the Gaussian components have a standard deviation of 2. From left to right, the Gaussian components are centred around -5, 1 and 5 with mixture weights of 0.20, 0.30 and 0.50. The left-most and right-most components represent one class and the middle component represents a second class, as shown in Figure 3.1. The task is to predict the class probabilities over the domain shown.

3.2.1 Pre-clustering

For illustration purposes, suppose it is unfeasible to rely on human expertise to classify the one-dimensional example. Following the stated goal of reducing labour requirements, the data is clustered autonomously. In this chapter, and the remainder of the thesis, a VDP model [46] is used to categorise the training data into clusters.

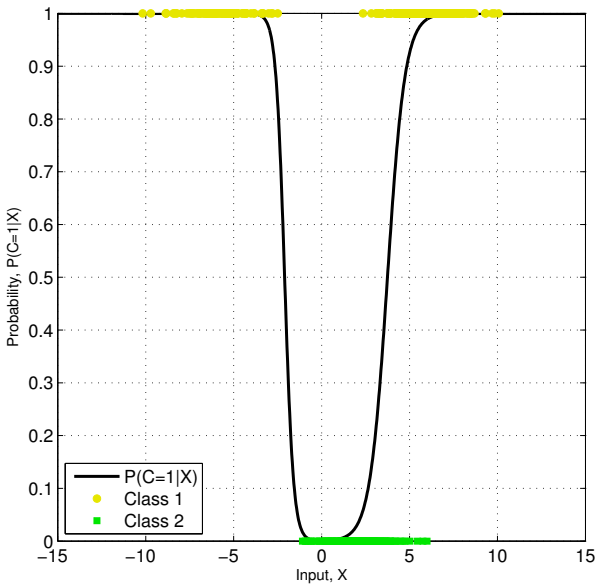
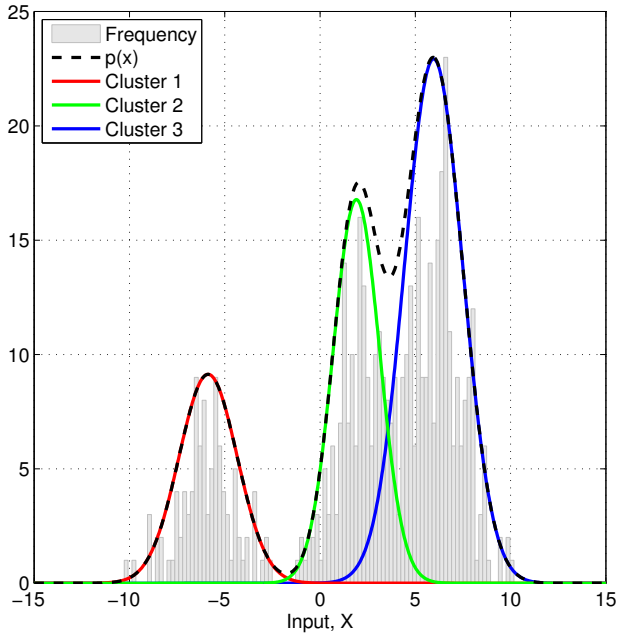


Figure 3.1 – One-dimensional example. The black line represents the underlying probability of observing the first class. The value of observations made on the X-axis are shown as yellow and green dots for the first and second class respectively.

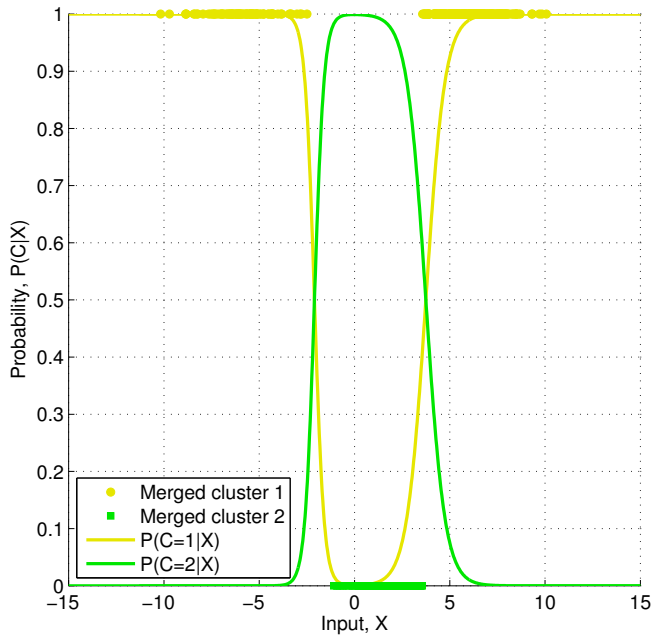
The VDP is a Bayesian, non-parametric model with the attractive property that it does not require the number of clusters to be specified in advance. Clusters are detected in the data by modelling the density of the data as a Gaussian mixture model (GMM). The number of Gaussians in the mixture, the mixing coefficient and the mean and variance of each Gaussian component are inferred from the data during training.

Three clusters were discovered and are shown in Figure 3.2a. For a human expert, analysing the composition of the three clusters is a simpler task than processing all 500 observations. Prior to classification, an expert can review and modify the clusters to ensure they are sensible. To simulate this sort of 'sanity' check, the first and third cluster are consolidated into one group as shown in Figure 3.2b.

The one-dimensional example is sufficiently represented by the VDP after human approval. There is no need to learn a mapping between the input-space and the probabilistic targets supplied by the VDP. In real data sets this relationship is not trivial. With this in mind, the one-dimensional example is continued for illustration purposes.



(a) Cluster density



(b) Cluster probability

Figure 3.2 – Clustering the one-dimensional data set. **(a)** Density estimate of the data after clustering the data with a variational Dirichlet process (VDP) model. The clusters are shown in red, blue and green. **(b)** Predictive probabilities given by the VDP after a human has merged the first and third clusters into one group. The merged clusters are shown in yellow and the remaining cluster is shown in green.

3.2.2 Regression

In PLSC, classification is treated as a regression problem (see Section 2.2.2). Rather than performing regression on discrete cluster labels, $\mathbf{y} \in \{-1, +1\}$, the continuous nature of regression can be exploited by performing regression on continuous cluster probabilities. For the benefit of the 'squashing' stage, the probabilistic targets are scaled from $\mathbf{y} \in [0, 1]$ to $\mathbf{y} \in [-1, 1]$.

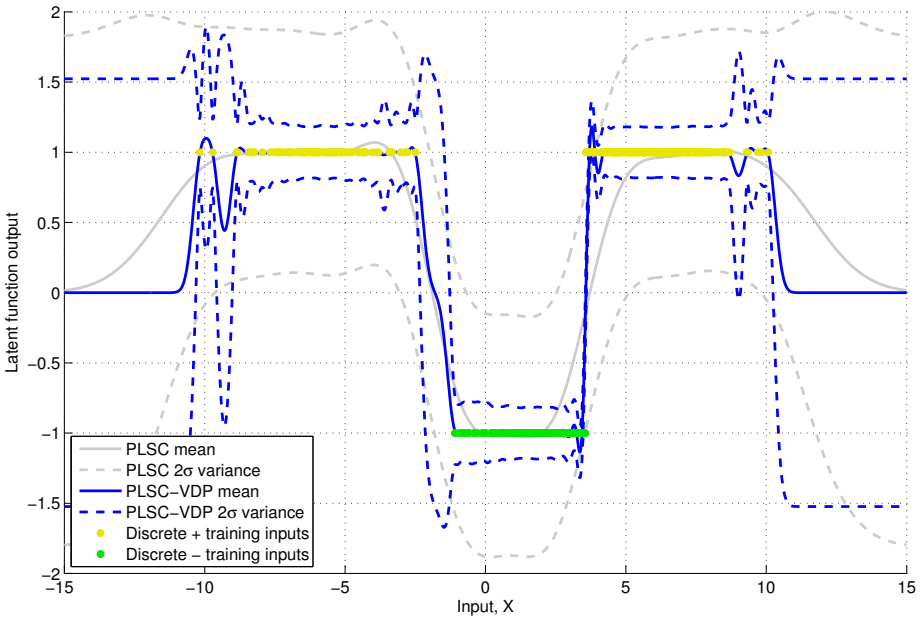
The predictive probabilities produced by a PLSC trained on noisy observed labels, a PLSC naively trained on discrete VDP cluster labels (PLSC-VDP) and a PTLSC trained on continuous VDP cluster probabilities are compared in Figure 3.3 and Table 3.1.

Model	Mean Variance	Length Scale	Signal Noise
PLSC	0.191	1.789	0.424
PLSC-VDP	0.050	0.258	0.087
PTLSC	0.000	0.558	0.000

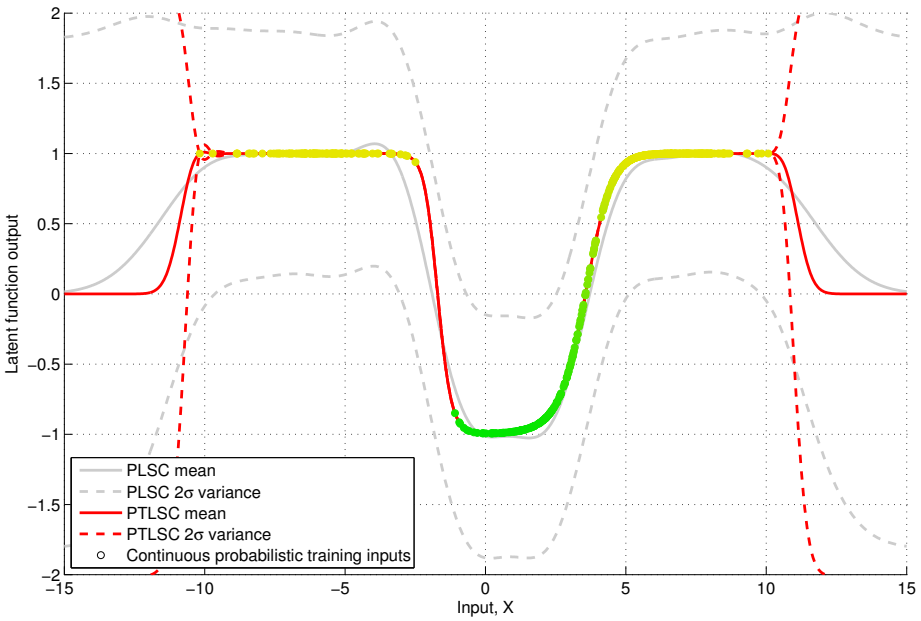
Table 3.1 – Comparison of the optimised squared exponential kernel parameters and mean variance for the classifier latent functions. The mean variance is calculated on the domain $[-10, 10]$ where there is data to support inference.

The PLSC-VDP latent function, shown in Figure 3.3a, must model a step change in the discrete training inputs. This is a difficult task for a regression model which assumes smoothness in the data and is achieved by learning a short length scale and a medium noise term. By only using the VDP cluster labels the PLSC-VDP has discarded valuable information.

Uncertainty in the underlying function is represented in the VDP as cluster probabilities. The rescaled cluster probabilities, used to train the PTLSC, preserve all of the information provided by the VDP. Since the probabilistic training data is continuous, it is better suited to the continuous nature of regression enabling the PTLSC latent function to fit the data almost exactly, as shown in Figure 3.3b. The model is able to learn a longer length scale than the PLSC-VDP, a low mean variance and low signal noise. These results indicate that the PTLSC is able to produce more general and confident predictions than the PLSC-VDP.



(a) Comparison of PLSC and PLSC-VDP latent functions



(b) Comparison of PLSC and PTLSC latent functions

Figure 3.3 – Comparison of classifier latent functions using variational Dirichlet process (VDP) training data. **(a)** Latent mean and variance functions of a PLSC trained on VDP cluster labels (PLSC-VDP). **(b)** Latent mean and variance functions of a PLSC trained on VDP cluster probabilities (PTLSC). The latent mean and variance functions of a PLSC trained on the true labels are shown in (a) and (b) for comparison.

3.2.3 Sigmoid Function

The objective of the 'squashing' stage is to constrain the latent function to lie on the interval $[0, 1]$. Whilst the sigmoid function, given by Equation (2.11), fulfils this objective, it does not preserve the shape of the latent function. To preserve the shape of the latent function a new sigmoid function is proposed. The proposed sigmoid function is linear within a specified region of the input domain and a logistic sigmoid outside of this region. The linear portion preserves the shape of the input and the logistic sigmoid portion ensures extreme values are restricted to the interval $[0, 1]$.

More precisely, the linear-logistic sigmoid function is defined as a piecewise function for $x \geq 0$ in Equation (3.1).

$$\phi(x, C) = \begin{cases} mx + 0.5 & \text{if } x \leq C \\ \dfrac{1}{1 + \exp\left(-\beta\left(x + Z - C\right)\right)} & \text{if } x > C \end{cases} \tag{3.1}$$

The region of the input domain which is linearly transformed is specified by the linear cut-off parameter, C. The scaling parameter, β, and translation parameter, Z, of the logistic sigmoid must be solved to satisfy the condition,

$$mC + 0.5 = \frac{1}{1 + \exp\left(-\beta Z\right)} \ , \tag{3.2}$$

where the gradient of the linear segment, m, is given as,

$$m = \beta \frac{1}{1 + \exp\left(-\beta Z\right)} \left(1 - \frac{1}{1 + \exp\left(-\beta Z\right)}\right). \tag{3.3}$$

The conditions given by Equation (3.2) and Equation (3.3) force the linear and logistic sigmoid segments to form a smooth function where they meet. The scaling parameter, β, controls the gradient of the linear segment and the logistic sigmoid segment at the linear cut off C. The translation parameter, Z, is simultaneously optimised to ensure that the two segments meet at the same location on the Y-axis at the linear cut off. The parameters β and Z are not chosen by the user, they are optimised in response to

a particular choice of C. The parts of the linear-logistic sigmoid function are shown in Figure 3.4a. Symmetry is used to transform input values less than 0.

Finally, the parameterised linear-logistic sigmoid function is given by,

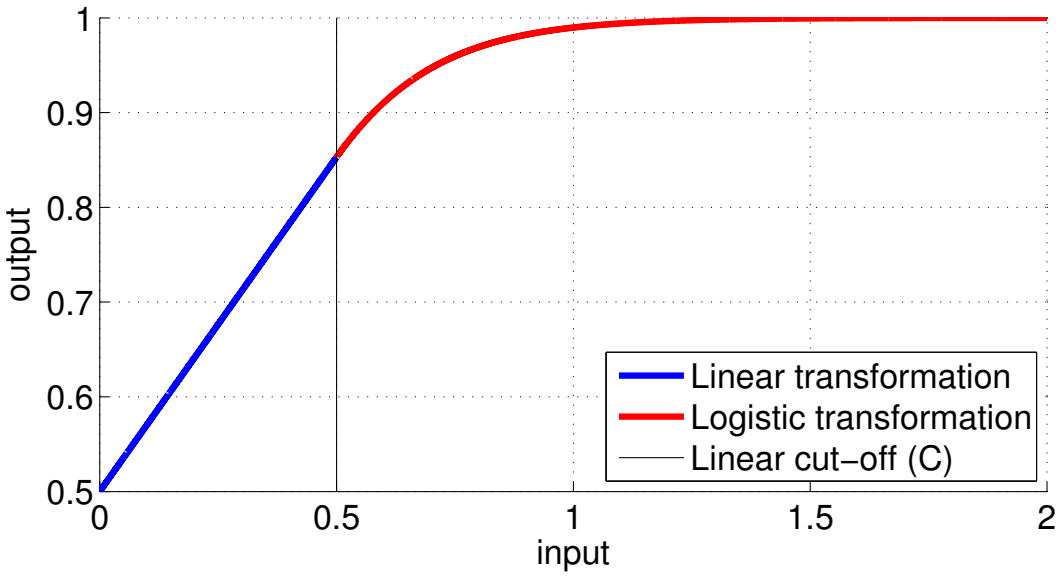
$$p(y_i \mid \mathbf{X}, \mathbf{y}, \theta) = \phi\left(\alpha\mu_i, \alpha C\right), \tag{3.4}$$

where the 'squashing' function, ϕ, is given by Equation (3.1) and μ_i is given by Equation (2.13). The effect of varying the parameters, α and C, is shown in Figure 3.4b. When the cut-off parameter is zero (red lines) the parameterised linear-logistic sigmoid function acts like a logistic sigmoid function. Increasing the scaling parameter, α, decreases the gradient of the linear segment and decreases the rate of signal attenuation in the logistic sigmoid segment.

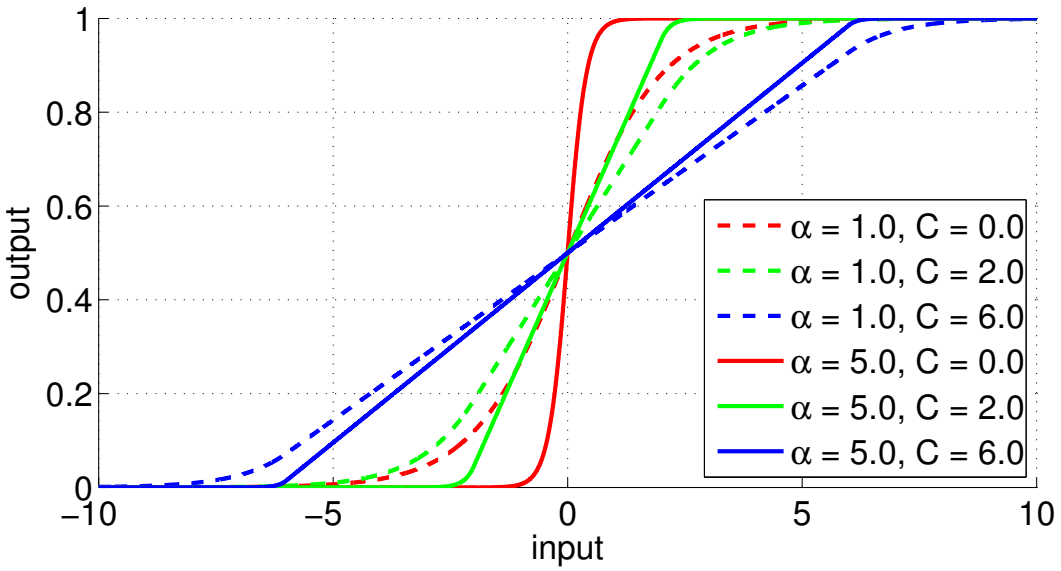
3.2.4 Training the Sigmoid Function

In the PLSC, the parameters of the sigmoid function are optimised by maximising the sum of the log leave one out (LOO) probabilities given by Equation (2.12). Under this construction, noise in the labels communicates uncertainty about the location of the decision boundaries. Noisy target data distributed around the decision boundary prevents the LOO objective function from creating a step change in the predictive probabilities.

In a PTLSC, the location of the decision boundaries are represented as degrees of belief in class membership. These continuous values do not contain any label noise. As a result, sigmoid function parameters which create a step change in the predicted probabilities are not penalised. Maximising the sum of the log LOO probabilities under these conditions rewards sigmoid function parameters where the transition between classes is as short as possible. This behaviour forces the predictive probabilities towards their extremes as shown in Figure 3.5a. Whilst the class labels may be correct, the predictive density does not appropriately preserve the shape of the latent function or represent the data generating process.

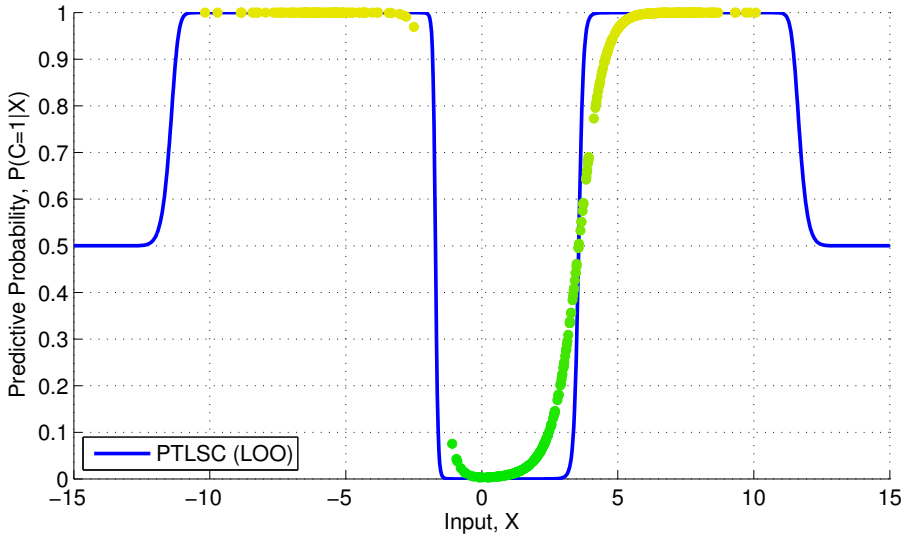


(a) Construction of linear-logistic sigmoid function

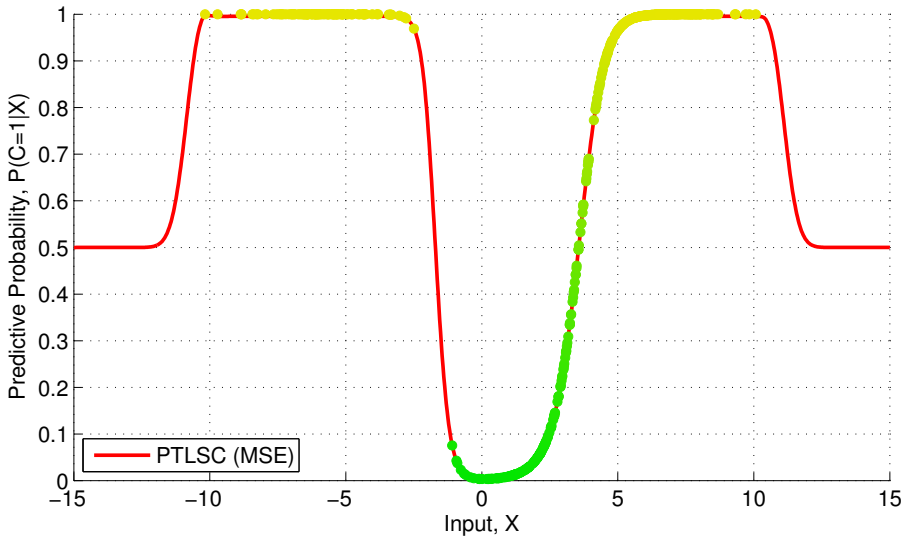


(b) Comparison of linear-logistic sigmoid parameters

Figure 3.4 – Linear-logistic sigmoid function. **(a)** Construction of the linear-logistic sigmoid function. The blue line shows the linear transformation and the red line shows the logistic transformation. The black line is the linear cut-off point where the signal transitions from a linear transformation to a logistic transformation. **(b)** Effect of varying the parameterised linear-logistic sigmoid parameters. Lines of the same colour are linear over the same portion of the input space. Solid lines attenuate faster than dashed lines.



(a) Sum of the log leave one out probabilities objective function



(b) Mean squared error objective function

Figure 3.5 – Comparison of linear-logistic sigmoid function objective functions. The coloured circles, in both subplots, represent the probabilistic training targets. The colour varies from yellow (first class) to green (second class) according to the target input probabilities. **(a)** The blue line represents predictive probabilities after maximising the sum of the log leave one out (LOO) probabilities with respect to the parameters of the linear-logistic sigmoid function. **(b)** The red line represents predictive probabilities after minimising the mean squared error (MSE) of the squashed probabilities with respect to the parameters of the linear-logistic sigmoid function.

An objective function better suited to preserving the shape of a PTLSC latent function is to minimise the mean squared error (MSE) of the 'squashed' probabilities with respect to the free parameters of Equation (3.4). This objective function rewards sigmoid function parameters which result in a predictive probability which follows the probabilistic target inputs as closely as possible.

The MSE is given by,

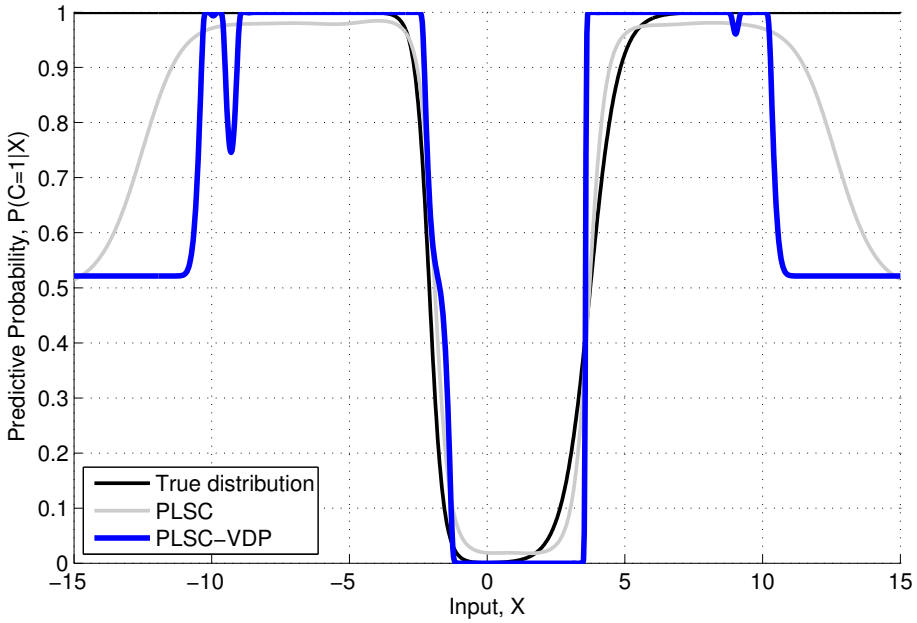
$$MSE\left(\mathbf{X}, \mathbf{y}, \theta\right) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - p(y_i \mid \mathbf{X}, \mathbf{y}, \theta)\right)^2,$$

and is measured between the 'squashed' probabilities and the probabilistic target inputs. This objective function rewards values for α and C which result in predictive probabilities which follow the probabilistic target inputs as closely as possible as shown in Figure 3.5b. The predictive probability is only subtly different from the original latent function (Figure 3.3b). The majority of the latent function has been linearly transformed and the shape has been preserved. The extremes of the latent function have been attenuated by the logistic portion of the linear-logistic sigmoid function to ensure they remain within the interval $[0, 1]$.

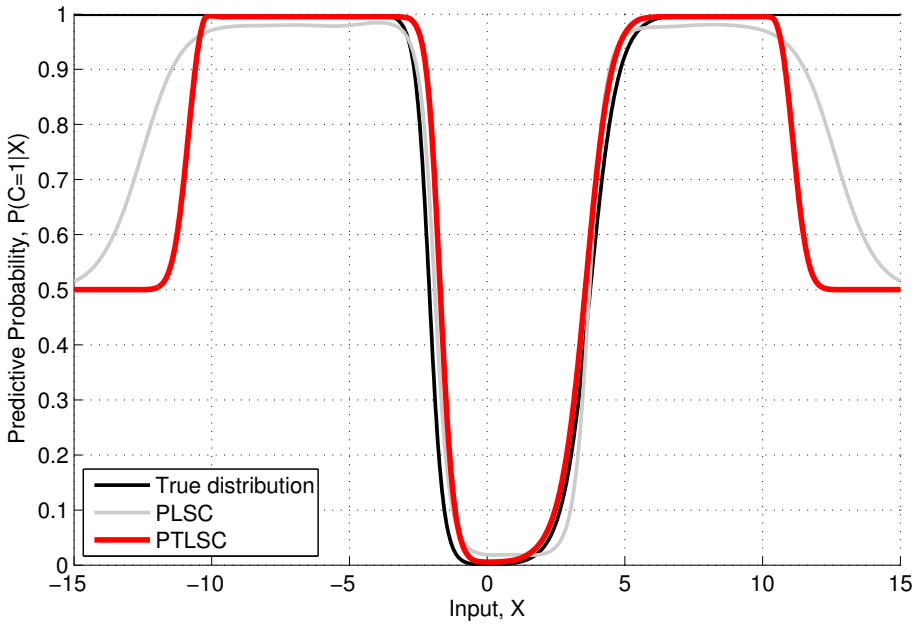
The predictive probabilities produced by a PLSC trained on the true noisy labels, a PLSC naively trained on discrete VDP cluster labels (PLSC-VDP) and a PTLSC trained on continuous VDP cluster probabilities are compared in Figure 3.6 and Table 3.2.

Model	Accuracy (%)	MSE
PLSC	98.80	0.002
PLSC-VDP	97.15	0.014
PTLSC	97.15	0.006

Table 3.2 – Comparison of classifier predictive probabilities. Accuracy and mean squared error (MSE) are calculated relative to the true distribution (shown in Figure 3.6) on the domain $[-10, 10]$, where there is data to support inference.



(a) Comparison of PLSC and PLSC-VDP predicted probabilities



(b) Comparison of PLSC and PTLSC predicted probabilities

Figure 3.6 – Comparison of classifier predicted probabilities using VDP training data. **(a)** Predictive probabilities of a PLSC trained on VDP cluster labels (PLSC-VDP). **(b)** Predictive probabilities of a PLSC trained on VDP cluster probabilities (PTLSC). The true distribution and the predictive probabilities of the PLSC are shown in (a) and (b) for comparison.

The PLSC has access to the true labels where uncertainty in the underlying function is communicated via noisy labels at the decision boundaries. The label noise occurs over a large region and causes the model to learn a long length scale and a large signal noise (see Table 3.1). This produces a smooth function which is able to achieve the highest label accuracy and lowest MSE.

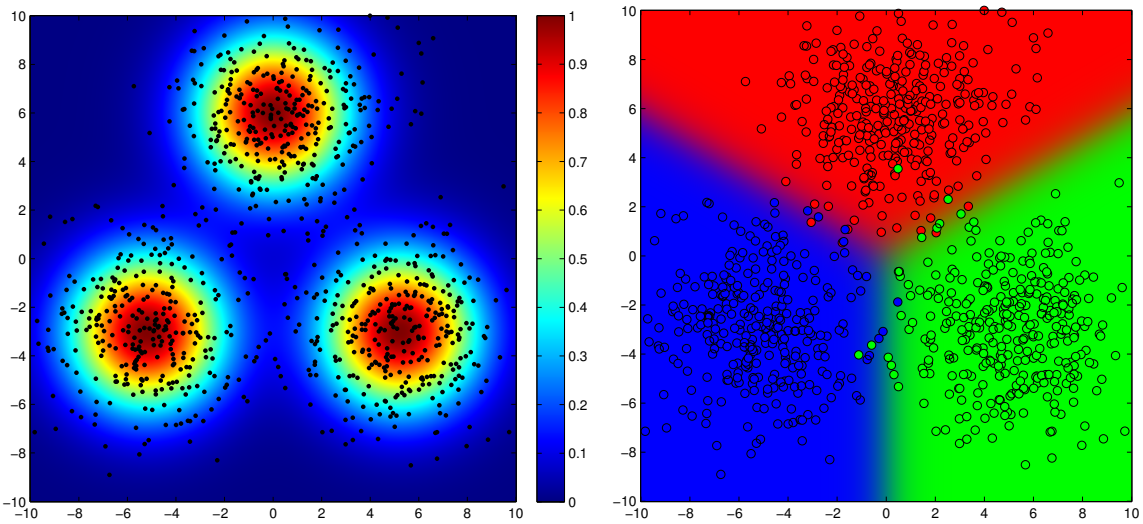
The discrete VDP cluster labels used to train the PLSC-VDP, only capture the locations of the decision boundaries. Once the PLSC-VDP latent function, shown in Figure 3.3a, has been 'squashed' into predictive probabilities all probabilistic information originally captured by the VDP has been lost. All that remains in the predictive probabilities produced by the model is the location of the decision boundaries as shown in Figure 3.6a. This defeats the purpose of using a probabilistic model and is reflected by the poorest MSE of all models.

The continuous VDP cluster probabilities used to train the PTLSC encode uncertainty about cluster membership as probabilistic degrees of belief. This fine-grained approach allows the PTLSC latent function to preserve more information contained in the VDP data than the PLSC-VDP. When the latent function is 'squashed' into predictive probabilities the shape of the latent function is preserved as shown in Figure 3.6b. The predictive probabilities result in a lower MSE than the PLSC-VDP.

Despite achieving a similar predictive accuracy, each model produces different predictive probabilities. In Section 2.2.1 evidence from the literature [47, 65] showed that the approximation to the marginal likelihood is not necessarily critical for producing accurate class predictions. This observation is replicated in the comparison between the PLSC, PLSC-VDP and PTLSC. If data labelled by an expert is available, the PLSC is able to produce both accurate class predictions and accurate predictive probabilities. If there is too much data for an expert to label it may be possible to categorise the training data using an unsupervised probabilistic model. The results from the illustrative example provided in this section show that the PTLSC is able to deliver more dependable predictive probabilities, with a negligible difference in predictive accuracy, than a PLSC-VDP. For situations where the predictive probability is considered important, the PTLSC is the preferred model.

3.3 Model Behaviour

To investigate the behaviour of the PTLSC in response to changes in cluster separation and cluster variance a controlled experiment was conducted in two dimensions. A GMM composed of three equally weighted clusters is placed in the environment. The mean of each cluster is placed a constant distance from the origin of the test domain. The means are separated from one another by 120 °. Observation locations are drawn randomly from the GMM according to the probability density function (PDF) of the GMM. At each observation location the observed cluster label is randomly drawn from the multinomial defined by the posterior probability of observing each GMM cluster at the observation location.



(a) Probability density function of the GMM and random observation locations

(b) Posterior probability of the GMM clusters and observed classes

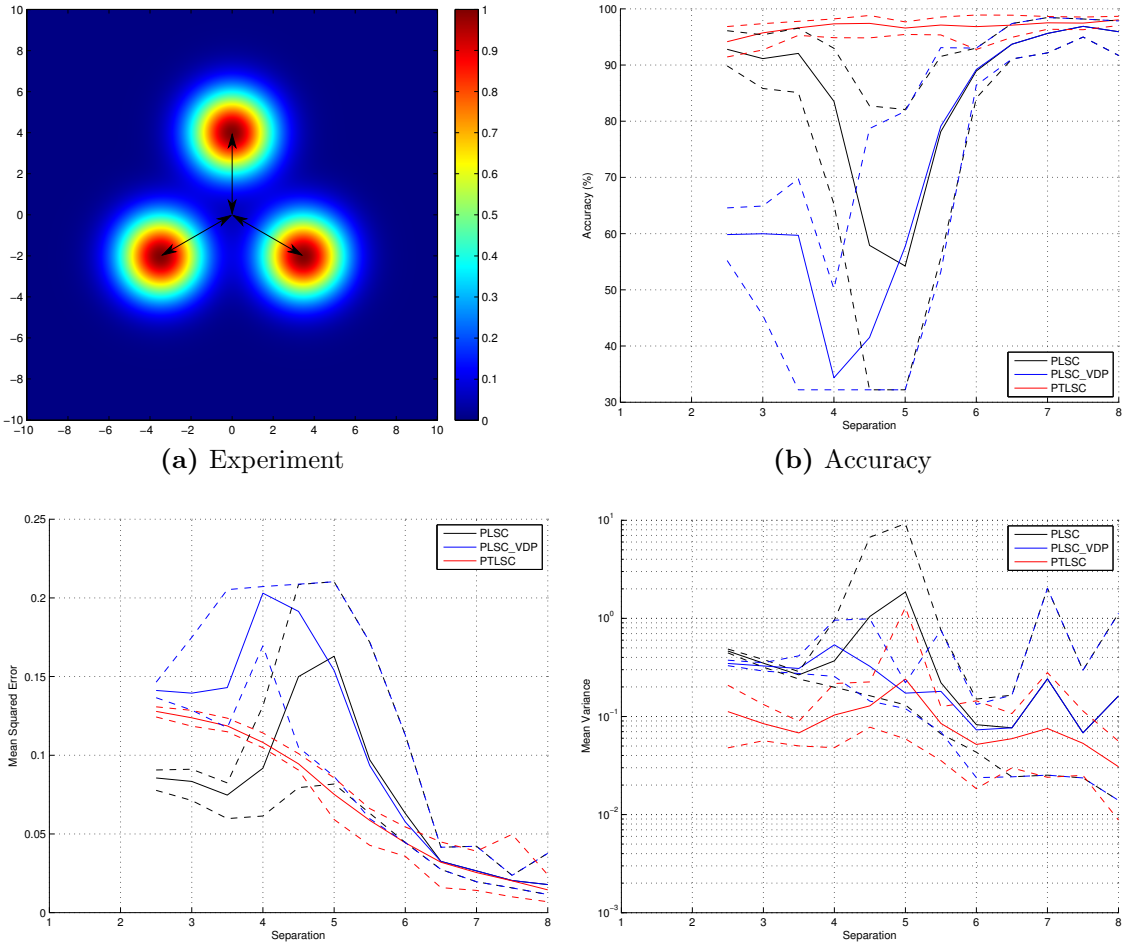
Figure 3.7 – Two-dimensional experiment. **(a)** probability density function (PDF) of the Gaussian mixture model (GMM). The intensity image represents the PDF of the GMM. The black dots represent 1000 random observation locations drawn from the GMM. **(b)** Posterior probability of belonging to a GMM cluster. The intensity image represents the posterior probability of belonging to a GMM cluster. The probability is represented as a mixture of the colours red, green and blue. Each of the three colours corresponds to a cluster. The coloured dots represent the observed class at the random observation locations. The observed cluster label is drawn randomly from the posterior probability at each observation location.

The effect of cluster separation on model performance is investigated by holding cluster variance constant and by modifying the distance of the clusters from the origin of the test domain. Cluster separation (Figure 3.8) is varied from one unit to eight units while the covariance matrix of each Gaussian component is fixed to a diagonal of two units. At cluster separations closer than 2.5 units, VDP clustering failed to recognise three clusters due to the close clusters being seen as inseparable.

The effect of varying cluster variance is investigated by holding cluster separation constant and by modifying the variance of the clusters as shown in Figure 3.9a. Cluster variance is modified by varying the diagonal of the covariance matrix of each Gaussian component from one to eight units. Cluster separation is held at a constant six units from the origin of the test domain.

Each configuration is tested across ten random folds of 1000 observations. Accuracy, MSE of the predicted probabilities and mean variance are calculated across the entire test domain. Accuracy and the MSE of the predicted probabilities are calculated relative to the ground truth distribution. The performance of the PLSC, PLSC-VDP and PTLSC in response to increasing cluster separation are shown in Figure 3.8. The performance of the models in response to increasing cluster variance are shown in Figure 3.9. Both experiments are a way of testing the performance of the models in the presence of increasing cluster overlap.

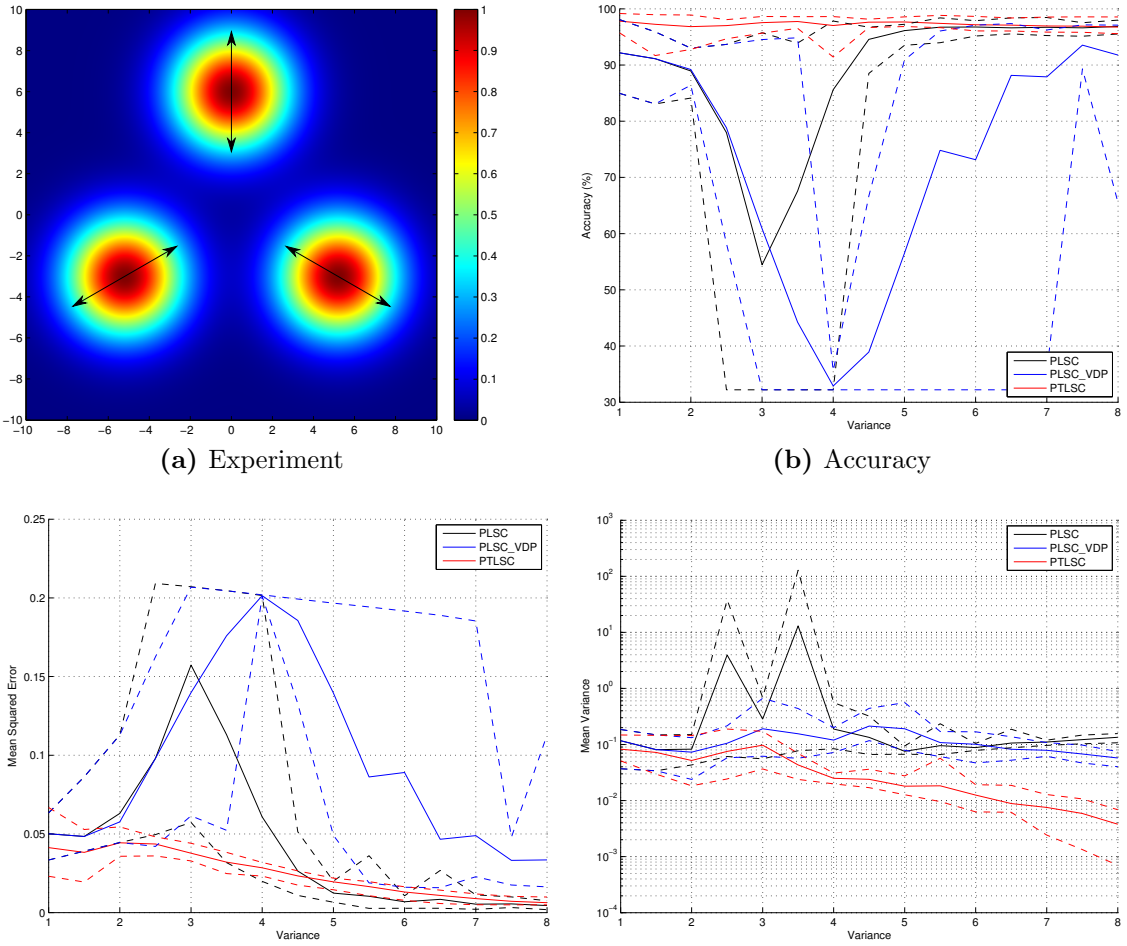
Model accuracy as a function of cluster separation is shown in Figure 3.8b. Model accuracy as a function of cluster variance is shown in Figure 3.9b. As cluster separation increases, there is a slight upward trend in accuracy of the PTLSC. Accuracy of the PTLSC was invariant to changes in the cluster variance. Most significantly, in both experiments, the PTLSC has the lowest variation between random draws. Additionally, the accuracy is largely unaffected by changes in cluster separation or variance. This indicates that the PTLSC is more robust to noise and class overlap in the observed data. By using probabilistic training data in confusing areas, the adverse affect of inconsistent label assignment is diminished. Instead of relying on label noise in ambiguous areas, probabilities which correctly identify uncertainty in the data are used. This prevents the model from learning incorrect decision boundaries.



(a) Experiment **(b)** Accuracy

(c) Mean squared error (MSE) **(d)** Mean variance

Figure 3.8 – Classifier performance as a function of cluster separation. The solid line represents the mean performance calculated over 10 random folds. The dashed lines represent the best and worst values achieved during cross-validation. **(a)** Experiment setup. The effect of varying cluster separation is investigated by holding cluster variance constant and by modifying the distance of the clusters from the origin of the test domain. The covariance matrix of each Gaussian component is fixed to a diagonal of two units. An example separation of 4 units is shown. **(b)** Model accuracy. **(c)** Mean squared error (MSE) of the predicted probabilities. **(d)** Mean variance of the predictions. Model accuracy, MSE and mean variance of the predictions are calculated across the entire domain. Model accuracy and MSE are calculated relative to the true distribution.



(a) Experiment **(b)** Accuracy

(c) Mean squared error (MSE) **(d)** Mean variance

Figure 3.9 – Classifier performance as a function of cluster overlap. The solid line represents the mean performance calculated over 10 random folds. The dashed lines represent the best and worst values achieved during cross-validation. **(a)** Experiment setup. The effect of varying cluster overlap is investigated by holding cluster separation constant and by modifying the variance of the clusters. Cluster separation is held at a constant six units from the origin of the test domain. An example covariance of four units is shown. **(b)** Model accuracy. **(c)** Mean squared error (MSE) of the predicted probabilities. **(d)** Mean variance of the predictions. Model accuracy, MSE and mean variance of the predictions are calculated across the entire domain. Model accuracy and MSE are calculated relative to the true distribution.

The PLSC and the PLSC-VDP display U-shaped accuracy curves in response to increasing separation (Figure 3.8b) and increasing variance (Figure 3.9b). Clusters with a small amount of separation produce data with a large degree of overlap. Similarly, clusters with a large amount of variance produce data with a large degree of overlap. If the data is separable by the VDP, overlap in the observed data helps improve the performance of the PLSC. The PLSC relies on label noise to faithfully model the location and predictive probability at boundary locations. As the cluster separation increases and cluster variance decreases, the amount of overlapping data decreases. This adversely affects the performance of the PLSC until the clusters are completely separated. Once full separation occurs model performance improves as there is little ambiguity concerning the location and predictive probabilities of the decision boundaries.

The same behaviour is displayed in the PLSC-VDP. Since label noise is removed by the discrete VDP labels, the PLSC-VDP has less information to determine the location of the decision boundaries. It is also more susceptible to poor estimates of the cluster means and covariances produced by the VDP. These factors result in lower accuracy than the PLSC or PTLSC.

The MSE of the predictive probabilities as a function of cluster separation is shown in Figure 3.8c. When the clusters are not distinctly separated, the PLSC benefits from access to the observed labels. As the separation between the clusters increases, the performance of the models converge to a similar result. The PLSC and PLSC-VDP also display U-shaped curves when the MSE of the predictive probabilities is plotted against cluster separation. Both models suffer in scenarios where there is enough overlap to introduce noise into the data but not enough data in the overlap to average out errors introduced by the noise.

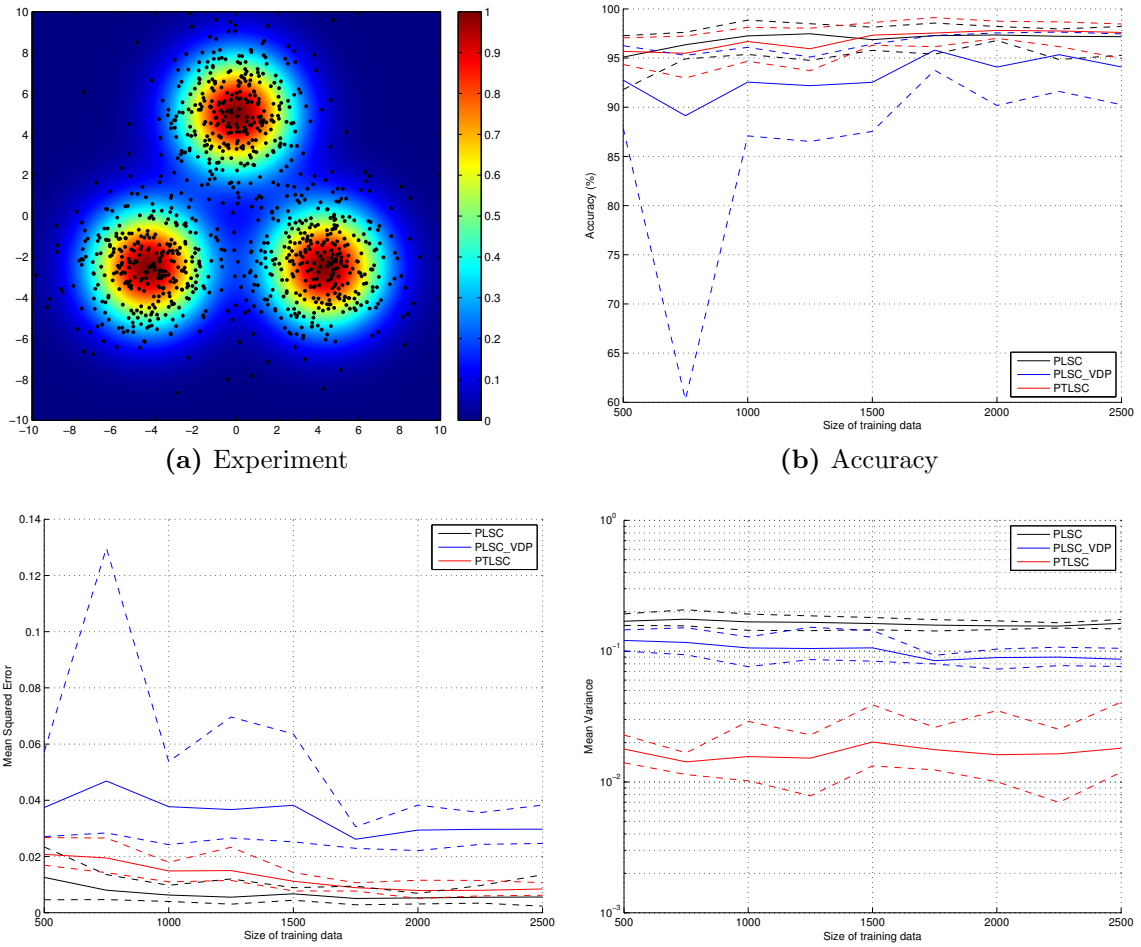
The MSE of the predictive probabilities as a function of cluster variance is shown in Figure 3.9c. As the amount of cluster variance is increased, the MSE of the predictive probabilities improves for all models. Again the PLSC and PLSC-VDP display U-shaped curves. A counter intuitive result is that the predictive probabilities of the models improve as the amount of data overlap increases.

GPs are data-driven models. The mean function returns to a prior of zero in locations far from observed data. This is accompanied by an increase in the predicted variance. In PLSC-based models, these properties create an uninformative predicted probability. Clusters with a small variance do not distribute observations widely across the environment. Data is concentrated at the mean of each cluster. Although this produces good separation, there are no observations to support inference in other areas of the environment. As a result the PLSC-based models will produce correct labels in areas of the environment far from observations but their predictive probabilities are conservative. This causes a decrease in performance when measured by the MSE relative to the true distribution.

As the cluster variance increases, more data is spread across the environment. This allows the MSE of the models to improve. It is also accompanied by a decreasing mean variance in the PTLSC as shown in Figure 3.9d. To model noise in the overlapping boundary regions, the PLSC and PLSC-VDP must learn relatively high noise terms. As a result the mean variance of the PLSC and PLSC-VDP remain largely consistent across all cluster variances tested. As the cluster separation increases, the mean variance of the models remains largely the same. The changes shown in Figure 3.8d are exaggerated by the log scaling on the Y-axis. The PTLSC produces more confident predictions than the competing models in both experiments.

The mean variance of the PLSC peaks when model performance is poor. Although poor performance is undesirable, this illuminates the advantage of using a probabilistic model. It is possible to evaluate model confidence by investigating the predicted variance. Predictions with a high variance indicate the model has inferred the class with a low confidence. Less faith should be placed in these predictions. High-variance predictions also suggest that more data or better features should be gathered.

The effect of the size of the training data on model performance is investigated by holding cluster separation constant at 5 units and cluster variance constant at 6 units. The size of the training data (Figure 3.8) is varied from 500 observations to 2500 observations. At training data sizes less than 500 observations, VDP clustering failed to recognise three clusters due to a paucity of data.



(a) Experiment **(b)** Accuracy

(c) Mean squared error (MSE) **(d)** Mean variance

Figure 3.10 – Classifier performance as a function of training data size. The solid line represents the mean performance calculated over 10 random folds. The dashed lines represent the best and worst values achieved during cross-validation. **(a)** Experiment setup. The effect of varying training data size is investigated by increasing the number of random samples used for training and holding cluster separation and variance constant at five and six units respectively. An example training data size of 1500 is shown. **(b)** Model accuracy. **(c)** Mean squared error (MSE) of the predicted probabilities. **(d)** Mean variance of the predictions. Model accuracy, MSE and mean variance of the predictions are calculated across the entire domain. Model accuracy and MSE are calculated relative to the true distribution.

The models show little correlation with the sizes of the training data tested. The PLSC and PTLSC offer similar accuracy and outperform the PLSC-VDP. The accuracy for all models increases as the number of training examples increases. Similarly, the MSE of the predictive probabilities reduce as the number of training examples increases. In both performance measures the models perform well with a small number of observations. Gains in performance are small as the number of training examples increases. This indicates that the problem is simple and does not require a large amount of data to be modelled effectively. The mean variance of all models effectively remains constant as the number of training instances is varied. The PTLSC is able to provide the lowest mean variance of all the models. This is due to the use of continuous training targets. The model does not have to accommodate large jumps from one label to another.

3.4 Summary

This chapter advocates using an unsupervised clustering algorithm to provide training data for classification. By using an unsupervised clustering algorithm to categorise the training data, the amount of human effort required to process raw observations into a training data is reduced.

To make full use of all the probabilistic data provided by the unsupervised clustering algorithm a new GP based classification model called the PTLSC is proposed. Rather than learning a mapping from an input space to discrete labels, it is designed to learn a mapping from the input space to continuous probabilities. This modification allows the model to preserve more information contained in the probabilistic training data than simply using the most likely class.

The PTLSC was demonstrated on a simple one-dimensional example and its performance in response to various amounts of data overlap in a two-dimensional example was presented. These experiments show that the PTLSC outperforms naive PLSC classification with the most likely cluster label (PLSC-VDP) in all conditions. In many of the parameters tested the PTLSC was able to outperform the PLSC which

has access to the observed data. This result indicates that the PTLSC can provide similar performance to PLSC.

The key observation is that when the training data has been clustered, naively using the label of the most likely cluster does not produce equivalent performance to a manually classified data set when the same classification model is used. By using the cluster probabilities, the PTLSC is able to produce a higher level of performance whilst not relying on a human to manually create a training data set.

In Chapter 4 the PTLSC is applied to real marine data. The objective is to generate a habitat model using benthic data collected in Tasmania, Australia. The model is used to learn the correlation between features extracted from a digital elevation model (DEM) of the seafloor, and habitat categories derived from in-situ images. The DEM is collected using ship-borne multibeam echo-sounder (MBES) data and the in-situ images are collected by an ocean going autonomous underwater vehicle (AUV). Manually classifying all the imagery collected by the AUV is a daunting task and is a good candidate for automation.

Chapter 4

Benthic Habitat Modelling

Benthic habitat maps make *"use of spatially continuous environmental data sets to represent and predict biological patterns on the seafloor (in a continuous or discontinuous manner)"* [12]. In the past decade remote sensing and in-situ sampling techniques have improved the extent and accuracy of broad-scale benthic habitat maps [10]. Current survey technologies allow oceanographers to develop broad-scale, digital terrain models of the seafloor at a relatively low cost. By intelligently combining this data with in-situ samples of the seafloor makes it possible, in principle, to create a useful representation of the distribution of biological and substrate patterns on the seafloor.

This chapter provides a review of habitat mapping techniques and presents a semi-autonomous method for converting a large number of observations into a habitat map. The chapter is arranged as follows. Section 4.1 describes methods used in the literature to create bathymetric habitat maps. Various techniques for collecting bathymetric data are discussed in Section 4.2. Section 4.3 describes common bathymetric features which are used to create habitat maps. Section 4.4 describes a data set collected from Tasmania, Australia including ship-borne, multibeam bathymetry and optical imagery collected by an autonomous underwater vehicle (AUV). A semi-autonomous habitat mapping pipeline, based on the probabilistic targets least squares classifier (PTLSC) from Chapter 3, is proposed and demonstrated using this data. Section 4.6 summarises and concludes the chapter.

4.1 Models for Bathymetric Habitat Mapping

Creating seafloor habitat maps is a cross-disciplinary problem which draws from the fields of marine biology, ecology, geology, hydrography, oceanography and geophysics. Given the wide range of fields which have studied bathymetric habitat mapping, many different solutions have been proposed in the literature. Brown et al., 2011 [12] provide a comprehensive review of methods used to produce seafloor habitat maps and break down seafloor habitat mapping into three broad strategies.

Abiotic surrogate mapping uses unsupervised methods for clustering the bathymetric data into similar groups. Abiotic factors are non-living environmental factors which affect ecosystems such as light, meteorological conditions, available chemistry and topography. Benthic species show preferences for certain seafloor geology and morphological characteristics [37, 101]. By clustering bathymetry based on surface morphology features, it is possible to capture the abiotic preferences which are a function of the seabed morphology. Since this method does not rely on ground-truthing, the regions produced during clustering will not have a precise descriptive semantic meaning.

Assemble first, predict later is a strategy which models the correlation between segmented environmental data and classified ground truth data. Ground truth data is established by collecting in-situ measurements and using expert judgement or statistical methods to classify the data. Although this strategy only models the correlation between ground-truth classes and segments in the environmental data, Brown et al. find it to be the most common mapping strategy presented in the literature. This is also know as the 'top-down' approach [86].

Predict first, assemble later is a more fine-grained approach than the 'assemble first, predict later' method. Rather than simply modelling the correlation between ground-truth classes and segments in the environmental data, the classes are modelled as a function of the environmental data. These methods rely on supervised classification techniques to model the functional dependence of

ground-truth classes on the continuous environmental data. The advantage of using a supervised method is that knowledge of the domain and human expertise can be included in the modelling process at the expense of higher labour costs. This is also known as the 'bottom-up' approach. [80].

Many models have been used to classify bathymetry. Commercial products are used within the literature to segment bathymetry [11, 60]. A popular example is Quester Tangent Corporation of Canada (QTC) multiview software. Generic unsupervised clustering techniques such as K-means [8, 13] are also used widely. Many solutions proposed in the literature reduce the number of dimensions in the environmental data through the use of principal component analysis (PCA) [13, 22, 62] before applying unsupervised clustering.

A commonly used supervised classification algorithm is decision trees [34, 37, 80]. Other supervised techniques include neural networks [59], support vector machines (SVMs) [31] and kriging [86]. Several supervised methods are compared in [52] where random forests are found to be effective. In [36], several models are used to predict the distribution of sponge assemblages given bathymetry and chemical data. Non-parametric machine learning models are found to perform better than the traditional regression models. Supervised classification techniques are amenable to the predict first, assemble later strategy. This 'bottom-up' approach is better able to create relevant correlations between observed ground truth data and the environmental data [80, 86]

In this thesis, the 'predict first, assemble later' approach is adopted. The model selected for habitat mapping in this thesis is the Gaussian process (GP). Since GPs are non-parametric models, no strict assumptions need to be made about the relationship between the environmental data and the observed ground truth data. As a probabilistic model, GPs are also able to produce confidence estimates during prediction. This extra data functions as an important diagnostic tool which can be used to improve the model.

4.2 Data Acquisition

4.2.1 Broad-Scale Bathymetry

There are two methods for acquiring large bathymetric data sets remotely: airborne light detection and ranging (LIDAR) and ship-borne sound navigation and ranging (SONAR). Each method has its own advantages and limitations, depending on the depth, resolution and coverage required.

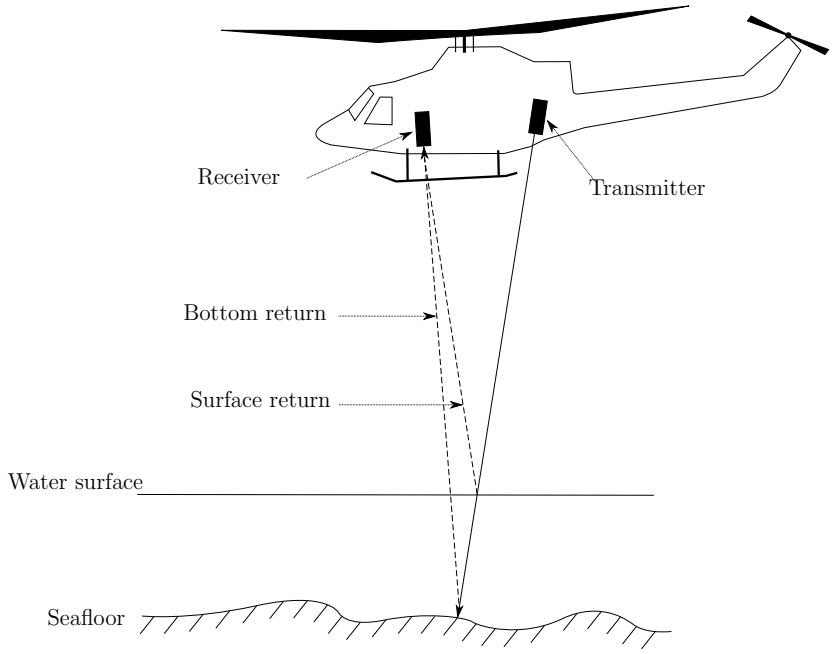


Figure 4.1 – LIDAR operating principle. Figure adapted from [38].

Airborne bathymetric LIDAR is a relatively new technology [38]. The system operates by transmitting laser pulses towards the earth's surface. Part of the light energy is reflected by the water surface, the remainder penetrates the water column and is reflected by the seafloor as shown in Figure 4.1. The water depth can be calculated from the time elapsed between the two reflections. The main disadvantage of bathymetric LIDAR is that electromagnetic signals are attenuated rapidly via absorption, scattering and refraction. As a result bathymetric LIDAR is limited to shallow and clear waters. The sensor is effective up to a depth of approximately 50 metres and 70 metres in ideal conditions [17, 19].

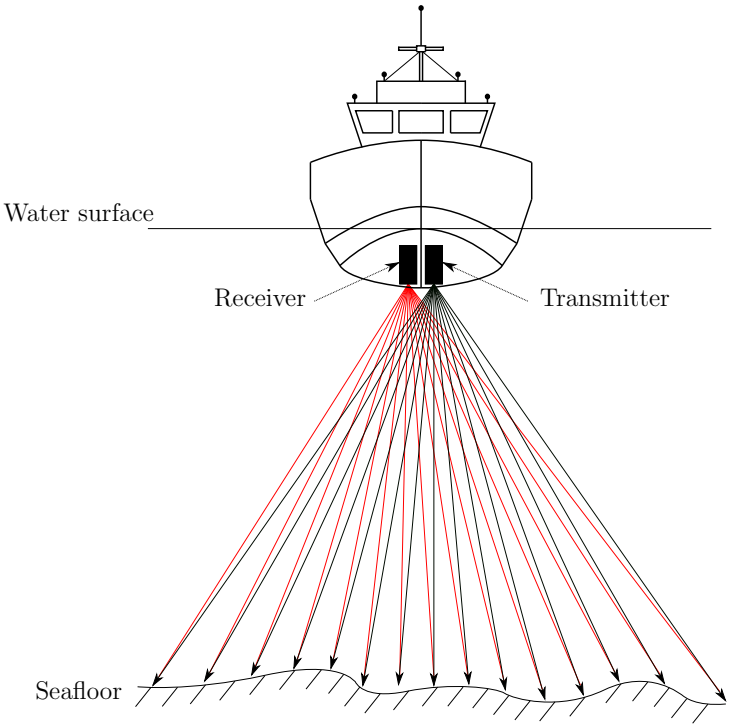


Figure 4.2 – Multibeam echo-sounder (MBES) operating principle.

To acquire bathymetric data at greater depths or in turbid waters, remote ship-borne acoustic surveys of the seafloor can be used to collect broad-scale data sets. The two most commonly used systems are side scan SONAR and multibeam echo-sounder (MBES) [50]. Rather than using electromagnetic signals, which attenuate rapidly in water columns, acoustic systems are able to penetrate greater depths by emitting a pulse of sound called a "ping". The time elapsed between transmitting a ping and receiving its reflection is used to calculate the depth to the seafloor. A swath of pings demonstrating the way in which MBES systems operate is shown in Figure 4.2. It is possible to achieve resolutions from tens of centimetres to tens of metres [42] depending on the depth of the seafloor and the design and technology of the acoustic systems. MBES systems are becoming the preferred method of acquiring acoustic benthic data due to their ability to collect bathymetry and backscatter information simultaneously [10]. MBES systems can be designed to resolve objects down to a size of one metre and penetrate water depths up to a kilometre [42].

Bathymetric LIDAR has several advantages over ship-based MBES systems in *shallow* environments. The resolution of MBES systems is a function of the seafloor depth. As the depth increases, the resolution of an MBES system decreases. LIDAR systems provide a signal which is *almost* independent of depth [19]. LIDAR systems are also capable of seamlessly integrating coastal bathymetry and land topography. However, in deep environments, airborne LIDAR systems cannot provide the water penetration and signal clarity that ship-based MBES systems are able to produce.

Ship-borne MBES data is used in this thesis. The ability to image the seafloor at depths greater than 70 metres allows habitat maps, generated using bathymetry, to cover the entire depth of the photic zone up to 200 metres [21]. Although MBES is used in this research, the methods presented apply equally to other digital elevation model (DEM) data sets.

4.2.2 In-situ Observations

Many methods for collecting observations of the seafloor are available. These methods are reviewed and evaluated by van Rein et al., 2009 [96]. The methods are categorised as operating over fine ($< 10m$), meso ($10m - 1km$) and broad ($> 1km$) scales. Methods for collecting data over broad-scales was discussed in the previous section. This section focuses on collecting direct observations of the seafloor at meso and fine scales. To provide an overview of the methods available at meso and fine scales, the findings of van Rein et al. are summarised in this section and Table 4.1.

At a fine scale, common methods for providing in-situ observation of soft substratum are grab samples and sediment cores which can be conducted from a ship. These methods disturb the environment by physically retrieving samples from the seafloor which are analysed by experts. These samples will consist of various portions of sediments such as clay, mud, silt, sand and gravel. Grab samples collect sediment from the surface of the seafloor ($< 20cm$) whereas benthic cores are able to collect sediment deeper from the surface. In scenarios where minimal disruption to the environment is desired or when the environment contains hard and complex substrates such as coral

Scale	Method	Substratum	Impact	Cost	Deployment
Fine	Grabs	Soft	Moderate	Moderate	Ship
	Cores	Soft	Moderate	Moderate	Ship
	Quadrat	Hard	Low	Low	Scuba
	Photoquadrat	Hard	Low	Moderate	Scuba
	Transect	Hard	Low	Low	Scuba
	Video transect	All	Low	Moderate	Scuba
Meso	Trawls	Soft	High	Moderate	Ship
	Dredges	Soft	High	Moderate	Ship
	Towed camera sled	Soft	Moderate	Moderate	Ship
	Dropdown camera	All	Low	Moderate	Ship
	ROV	All	None	High	Ship
	Manta towed video	All	None	Low	Ship
	DPV	All	None	Moderate	Scuba
All	AUV	All	None	High	Ship

Table 4.1 – Summary of methods for making direct observations of the seafloor based on Table 1 of [96]. Impact is defined as the level of disturbance to the monitored biota and cost refers to the financial cost of the monitoring method.

reefs, scuba divers are deployed. Although human divers are able to provide detailed data through quadrats and transects, the surveys are limited by safe diving practices and tidal and weather conditions.

Monitoring methods at meso scales can also be divided by their impact on the environment. Methods such as trawling and dredging retrieve physical samples from the environment which can be inspected by experts. Other less destructive methods of monitoring can observe hard substrates. Dropdown cameras are optical systems that are deployed to hover over the seafloor and take point observations. Towed camera sleds are optical systems that are dragged over the seafloor. Manta towed systems are optical systems that are towed at a constant altitude above the seafloor. Of these methods, remotely operated vehicles (ROVs) offer the greatest precision in control and can operate in difficult terrain but are a costly platform to deploy.

AUVs were not evaluated in the van Rein et al. study. However van Rein et al. acknowledge that when instrumented appropriately, AUVs are able to collect high-resolution data across all spatial scales, including broad-scales. Although AUVs are a relatively expensive and scarce tool for observing the seafloor, they offer several advantages over current methods as summarised in Table 4.1.

The primary advantage of AUVs over other methods is their ability to operate in close proximity to the seafloor and follow rugged terrain. Unlike human divers they are not constrained by either depth or dive duration. Being decoupled from the surface also allows AUVs to operate in sea conditions which preclude the use of ROVs and rugged benthic terrain which is inaccessible to towed video systems. AUVs are also a flexible platform and offer the ability to support a wide range of sensors, making AUVs a productive resource in a variety of contexts such as near-bottom SONAR-based surveys [6, 79] and high-resolution optical imaging [100].

Accurately geo-located, in-situ data is an important aspect of bathymetric habitat mapping. Poor geo-location introduces a registration error between categorised in-situ observations and broad-scale bathymetry. Depending on the severity and frequency of these mismatches, the benthic habitat model may learn incorrect associations and ultimately produce poor predictions. Ground truth data collected by AUVs introduces less registration error into habitat maps than traditional methods of ground truthing. The ability to perform dense, targeted surveys using state of the art, terrain aided navigation [3, 57, 99] makes AUVs a unique high-resolution tool for gathering in-situ data.

The in-situ data used in this thesis is obtained using an AUV. Precise navigation, consistent illumination and large volumes of high-resolution data make AUVs amenable to autonomous data processing systems. This raises interesting prospects as well as challenges which will be discussed in Section 4.4.2.

4.3 Bathymetric Features

It is widely understood in the marine science community that benthic species show a preference for specific seafloor characteristics [37, 42, 43]. It is this correlation with seabed morphology that allows marine scientists to produce benthic habitat maps. Interpreting the bathymetric data is a subtle task. The ability to separate habitats will depend on the resolution of the bathymetry and the scale at which the bathymetry has been analysed [101].

The bathymetric data may serve as a direct measure of biogenic structures such as reefs [102], kelp forests [60] or mussel beds [62]. In other cases, differences in the bathymetric feature vectors will not always correspond to a change in benthic habitat [18]. In these challenging cases, there may not be enough information contained in the bathymetry to properly disambiguate between habitats.

DEMs only provide depth to the seafloor. While it is clear marine life is correlated with depth [12, 43], it is possible to process DEMs into a variety of descriptors which can capture biological preferences. These derivative features provide a richer description of the environment and help to disambiguate between habitats that might otherwise appear similar. Wilson et al., 2007 [101] break down terrain analysis methods into slope, aspect, curvature and terrain complexity. Acoustic properties of the returned SONAR signals are also increasingly being used to categorise benthic habitats. These bathymetry features are described in the remainder of this section.

Slope describes the incline of a local region [14, 22, 37, 80]. The slope of a region affects how sediment can accumulation on the terrain. The presence of sediment provides different ecological niches to rocky formations devoid of sediment. As a consequence, flat areas tend to exhibit different rock formations and communities to steeply sloping regions of the seafloor.

Aspect or orientation measures the direction a slope faces [14, 22, 37]. The orientation of a location on the seafloor will determine its exposure to local water currents

and lighting conditions. These factors can determine the availability of nutrients and energy to organisms in the area.

Curvature measures the rate of change of slope in a defined direction [14, 22]. More precisely, curvature measures the second spatial derivative of the seabed terrain. The two most commonly calculated measures of curvature are profile and plan curvature. Profile curvature measures the rate of change in slope parallel to the direction of maximum down slope. Plan curvature measures the rate of change in slope perpendicular to the direction of maximum downward slope. These measures can help define surface features in the terrain such as ridges and valleys. Like aspect, curvature can provide information on water flows over areas of the bathymetry. Another commonly used measure of curvature is the bathymetric position index (BPI) [14, 22, 37]. This measure calculates whether a particular region forms a crest or trough by inspecting the elevation of neighbouring pixels.

Terrain complexity measures the amount of structure or roughness of an area. Complex environments tend to contain features which provide shelter and locations for organisms to settle. Simple environments often accumulate sediment providing different ecological opportunities. Terrain complexity can be measured in various ways. The terrain ruggedness index (TRI) measures variation of pixels surrounding a central pixel [22, 55]. Another widely used measure is rugosity [14, 22, 55, 80] which is defined as the ratio between the area of the surface and the area of the surface projected onto another plane. The horizontal planar area intersecting with the centre of the surface is usually chosen as the reference plane. Other terrain descriptors assess terrain complexity using fractals [44].

Backscatter measures the acoustic properties of the reflected multibeam echos. By analysing the variations in the echo strength, it is possible to develop an indication of acoustic "hardness" and infer the underlying geological material [13]. This data can be used in combination with morphological properties to detect relevant patterns in the seafloor [30, 37, 60].

4.3.1 Feature Calculation

In this thesis depth, rugosity and slope are used to describe the bathymetry. These features are easy to derive from bathymetry rasters and are widely used in benthic modelling. Terrain features are calculated by centring a window over each data point in the DEM. At each placement, the data point at the centre of the window is described in terms of the neighbouring data points which fall within the window. These data points form a local surface which can be processed by various functions to produce terrain descriptors. For raster based DEMs this operation is similar to the concept of convolution in image processing.

Typically rugosity is calculated by dividing the area of the local surface by the area of its footprint when projected onto the horizontal plane. Projecting the local surface onto this plane couples rugosity with slope [25, 80]. To decouple rugosity from slope Friedman et al., 2012 [25] propose projecting the local surface onto a plane of best fit. The same approach is adopted in this thesis. To provide an overview of the technique, the key equations used in [25] are summarised in this section.

For a data point in the bathymetry rugosity is calculated using

$$r = \frac{A}{A'} \tag{4.1}$$

where A is the area of the local surface surrounding the data point and A' is the area of the the local surface projected onto a plane of best fit. The benefit of this approach is that rugosity is decoupled from slope.

The bathymetry is represented using a triangle mesh where each observation of bathymetry forms a vertex in the mesh. This allows the surface area A to be represented by summing the area of all the triangles within a local surface. This is given by

$$A = \sum_{i=1}^{N} a_i \tag{4.2}$$

where N represents the number of triangles which fall within the local window.

In three-dimensional Cartesian space a point can be represented by the vector $\mathbf{x} = (x, y, z)$. In this space a triangle is defined by three points $\mathbf{x_A}$, $\mathbf{x_B}$ and $\mathbf{x_C}$. The area of the triangle can be calculated efficiently using a vector product.

$$a = \frac{1}{2}\left|\overrightarrow{AB} \times \overrightarrow{AC}\right| \tag{4.3}$$

where the edges of the triangle are given by the vectors

$$\begin{aligned}\overrightarrow{AB} &= (x_B - x_A, y_B - y_A, z_B - z_A)\\ \overrightarrow{AC} &= (x_C - x_A, y_C - y_A, z_C - z_A)\,.\end{aligned}$$

To calculate the projected area A', the plane of best fit must be determined. A plane can be fitted to the data by performing orthogonal regression using PCA. For three-dimensional data, three principal components are returned. The first two principal components form a basis for the plane which passes through the multidimensional mean of the data and minimises the sum of the squared distances of each point to the plane. The third principal component, $\hat{\mathbf{c}}$, defines the normal vector of the plane. The direction of the normal is forced to point upwards using

$$\hat{\mathbf{p}} = \begin{cases} \hat{\mathbf{c}} & \text{if } \hat{\mathbf{c}} \cdot \hat{\mathbf{k}} \geq 0 \\ -\hat{\mathbf{c}} & \text{if } \hat{\mathbf{c}} \cdot \hat{\mathbf{k}} < 0 \end{cases}$$

where $\hat{\mathbf{k}}$ is the upward facing unit vector. Finally the projected area, A', is given by

$$A' = \sum_{i=1}^{N} a_i \left(|\hat{\mathbf{p}} \cdot \hat{\mathbf{n}}_i|\right) \tag{4.4}$$

where

$$\hat{\mathbf{n}}_i = \frac{\overrightarrow{(AB)}_i \times \overrightarrow{(AC)}_i}{\left|\overrightarrow{(AB)}_i \times \overrightarrow{(AC)}_i\right|}.$$

Rugosity can then be calculated by applying Equations (4.3) and (4.4) to Equation (4.1).

Once the plane of best fit is known, calculating slope is trivial. Slope is defined as the angle between the normal vector of the plane of best fit and the horizontal plane. Slope is given by

$$\theta = \cos^{-1}\left(\hat{\mathbf{p}} \cdot \hat{\mathbf{k}}\right). \tag{4.5}$$

The feature calculation process is demonstrated graphically in Figure 4.3. The local surface is shown in red and the plane of best fit is shown in blue. The surfaces are defined by triangle meshes. Each vertex in the mesh is defined by a pixel in the bathymetry raster.

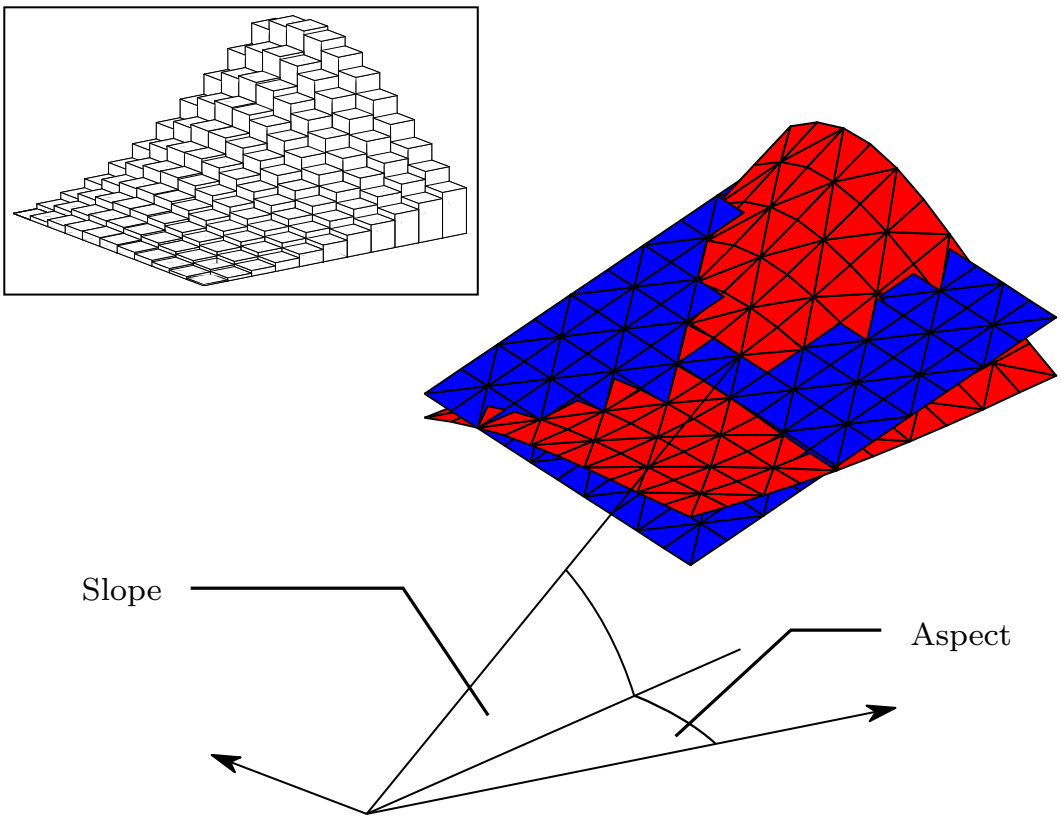
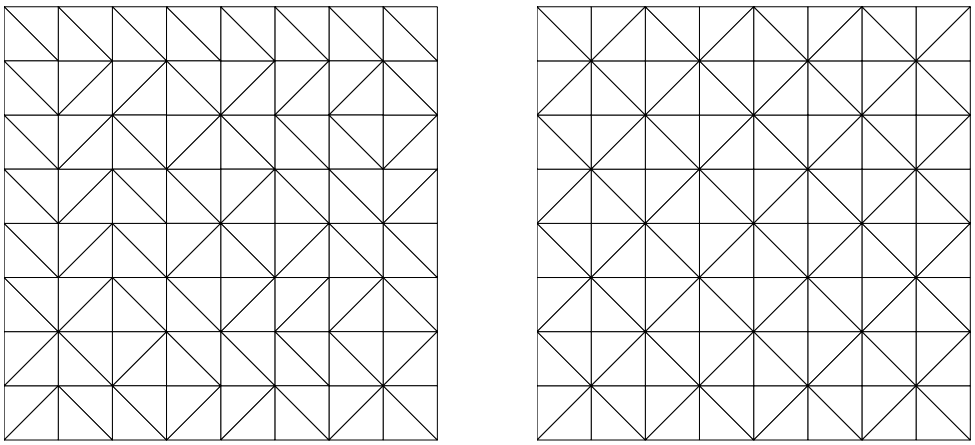


Figure 4.3 – Demonstration of bathymetry feature calculation. The box in the upper left corner represents a local window of data extracted from a DEM. The red surface represents the triangulated surface of the local window. The blue surface represents the local data projected onto the plane of best fit. Rugosity is given by the ratio between the surface area of the red surface and the surface area of the blue surface.

The method used for calculating rugosity and slope in this thesis differs from the method used in [25] in the way that triangle meshes are generated. In this thesis bathymetry is represented as a raster image. Since the data is arranged on a regular grid, there is no need to fit a mesh to the data using Delaunay triangulation as in [25]. Instead a predefined triangulation pattern can be repeated at regular intervals. Generating surfaces using this method has a negligible computational overhead and ensures that meshes of the same size will contain identical triangulations. A comparison of the two methods is shown in Figure 4.4.



(a) Delaunay triangulation **(b)** Regular triangulation

Figure 4.4 – Demonstration of creating a mesh from a DEM. **(a)** Delaunay triangulation on regular 9-by-9 grid of points. Note that although the data is arranged in a structured grid, Delaunay triangulation produces an irregular pattern. **(b)** Regular triangulation. A predefined triangulation is repeated at regular intervals defined by the grid of data. This method produces a neat, symmetric lattice.

Square windows are used where a specified 'radius' of neighbours defines the size of the window. Bathymetry features calculated at a scale N will have N pixels directly to the top, bottom, left and right of a central pixel as shown in Figure 4.5. The scale N can be converted from a pixel radius to a width in metres using:

$$W = 2Nr + r \tag{4.6}$$

where W is the square width of the window in metres, N is the pixel radius used during feature calculation (see Figure 4.5) and r is the size of each pixel in metres.

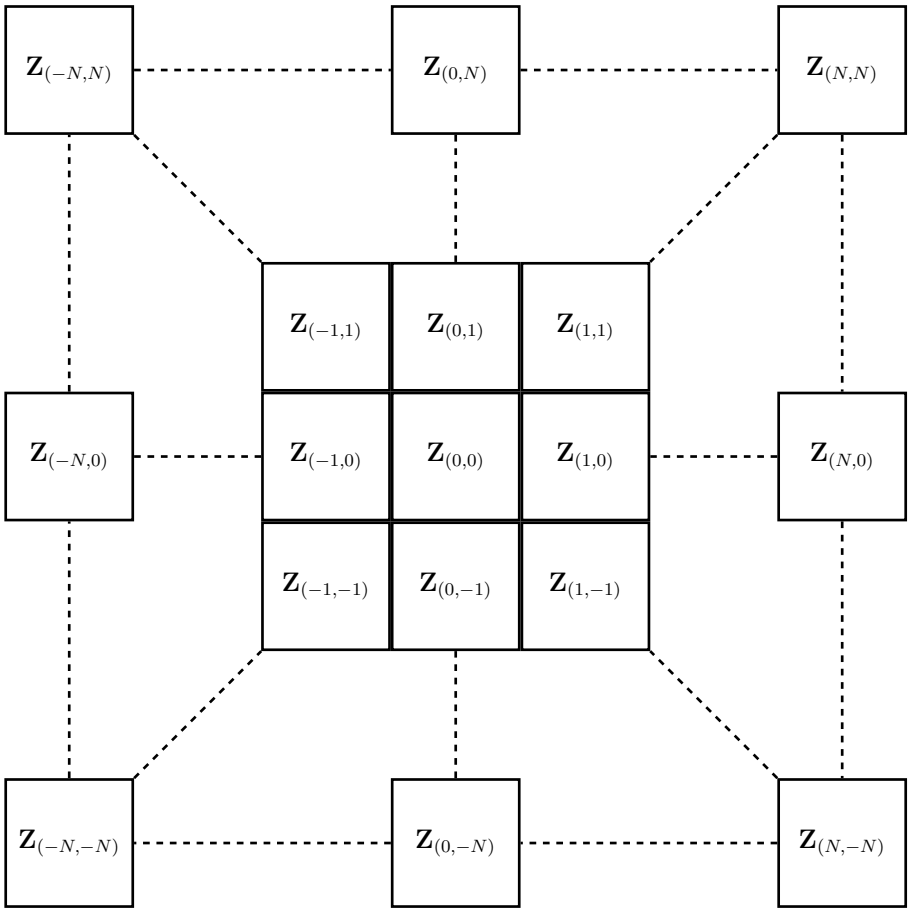


Figure 4.5 – Illustration of multi-scale analysis from [101]. A local co-ordinate system is centred around a point of interest. The location of neighbouring points are defined relative to this origin and are denoted by the subscripts. The point of interest, $\mathbf{Z}_{(0,0)}$, is assigned the result of the local analysis.

To calculate bathymetry features for the entire environment, the centre of the local window is moved from pixel to pixel. Only windows which contain a full set of data are processed. Windows which traverse holes or the edge of the bathymetry are ignored. This ensures that no edge artefacts are introduced into the data.

Knowing which scales are relevant to the habitats of interest is difficult to determine prior to modelling. Each habitat in the model is likely to be most strongly correlated with an individual scale. Wilson et al. show that models which use multiple bathymetric feature vectors at different scales produce better habitat maps, quantitatively and qualitatively, than models which rely on a single feature vector or scale.

4.4 Southeastern Tasmania Data Set

In 2008 a monitoring program was undertaken in southeastern Tasmania as a collaboration between the Tasmanian Aquaculture and Fisheries Institute (TAFI), at the University of Tasmania, Geoscience Australia (GA) and the Australian Centre for Field Robotics (ACFR) at the University of Sydney (USyd). The purpose of the program was "*to collect high-quality, accurately co-located physical and biological data to enable the robust testing of a range of physical parameters as surrogates of patterns of benthic biodiversity*" [64]. Broad-scale bathymetry was collected using ship-borne multibeam SONAR and in-situ observations were collected by towed-video and an AUV.

4.4.1 Bathymetry

Ship-borne MBES bathymetry was collected in June, 2008 by the research vessel *Challenger* operated by the TAFI. The vessel was equipped with a Simrad EM3002(D) 300 kHz multibeam SONAR system and an Applanix position and orientation system, coupled with a C-Nav GPS system [64]. The survey region is shown in Figure 4.6. The bathymetry is rasterised to a pixel size of 1.6 metres and is $16.7km$ tall and $10.8km$ wide. The bounding box of the survey region spans a rectangular region of approximately $181.4km^2$. There are 38,052,323 valid data points in the raster which cover an area of approximately $97.4km^2$.

Rugosity and slope for the southeastern Tasmania data set were calculated at scales of 2 ($8m$), 4 ($14.4m$), 8 ($27.2m$) and 16 ($52.8m$) and are shown in Figures 4.7 and 4.8. Log rugosity and log slope are used to increase 'contrast' in the data. The multibeam data collected in Tasmania was corrected for tides, vessel motion and elevation errors caused by dynamic draft of the vessel [64]. Although visual inspection of the bathymetry does not reveal any artefacts (Figure 4.6), small variations in the bathymetry are magnified when converted into bathymetry features. These variations are visible as artefacts in the bathymetry features (Figures 4.7 and 4.8) and are likely caused by surveying the same area in different wave conditions.

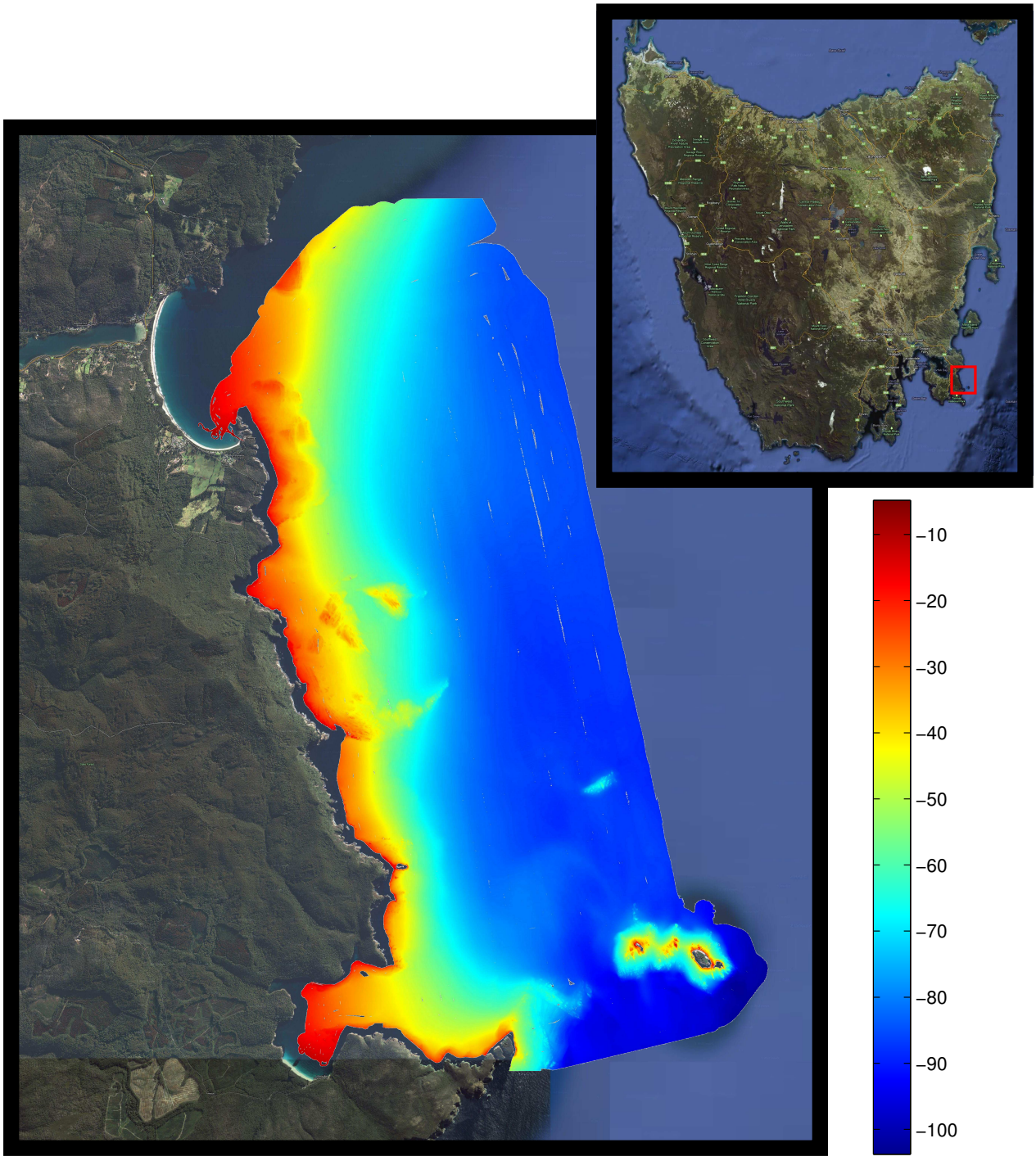
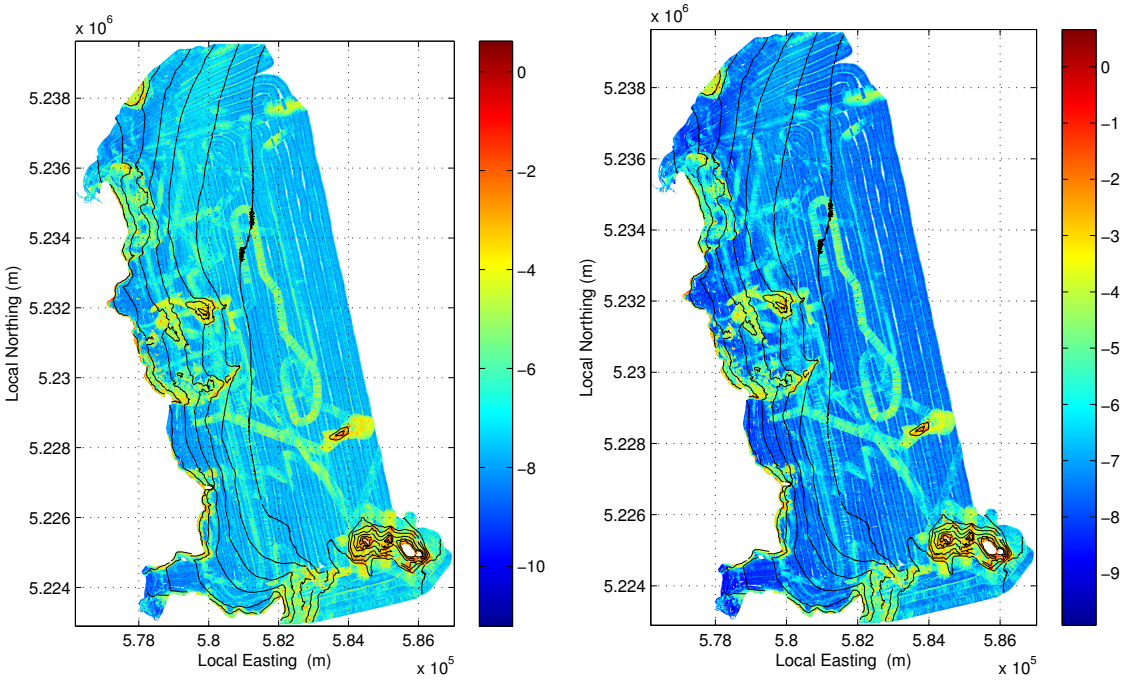
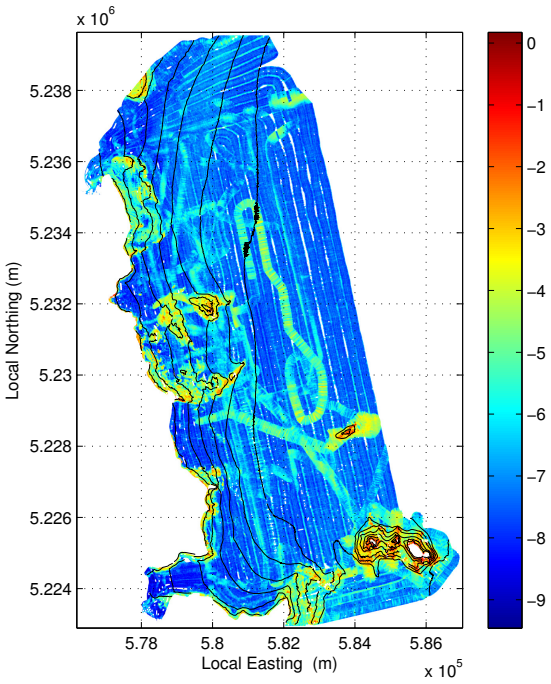


Figure 4.6 – Bathymetry of southeastern Tasmania survey region layered on top of satellite imagery obtained from Google Maps. The map in the upper right corner highlights the survey region in red and provides a context for the location and size of the survey box with reference to the Tasmanian land mass.

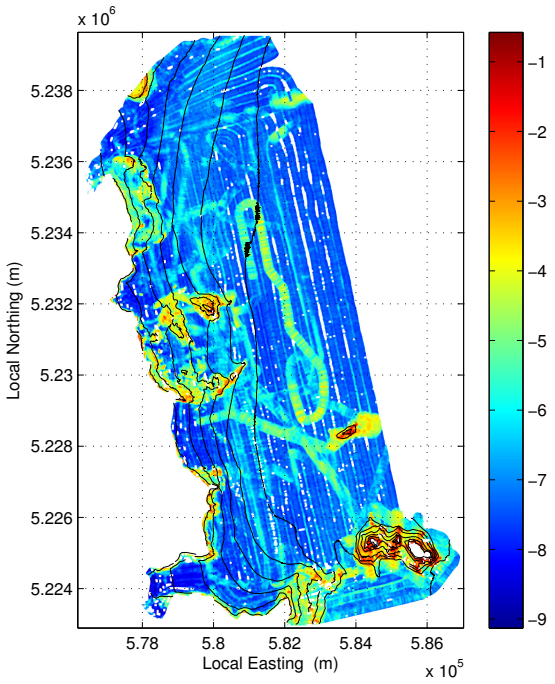


(a) Log rugosity, scale 2 ($8m$)

(b) Log rugosity, scale 4 ($14.4m$)

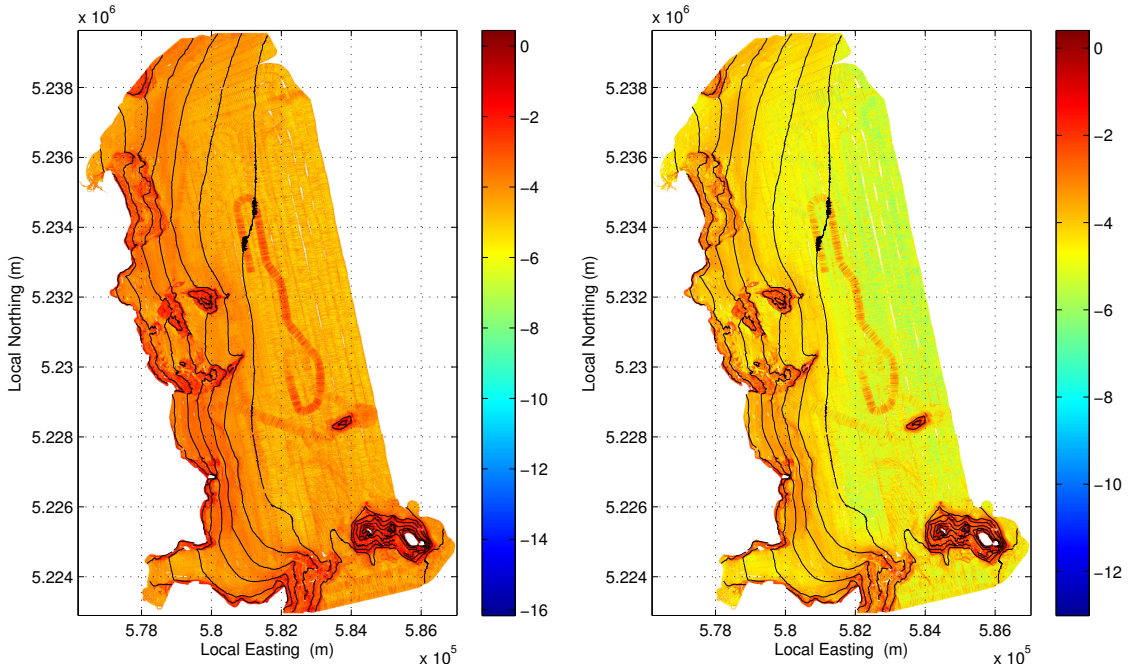


(c) Log rugosity, scale 8 ($27.2m$)



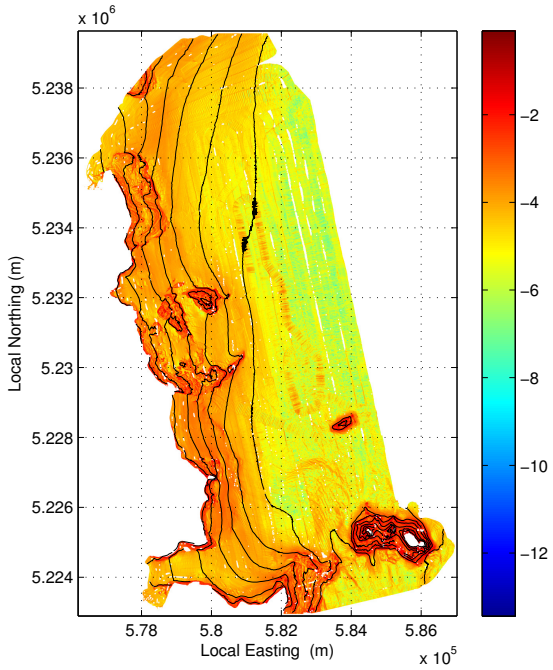
(d) Log rugosity, scale 16 ($52.8m$)

Figure 4.7 – Log rugosity for southeastern Tasmania calculated at four scales. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.

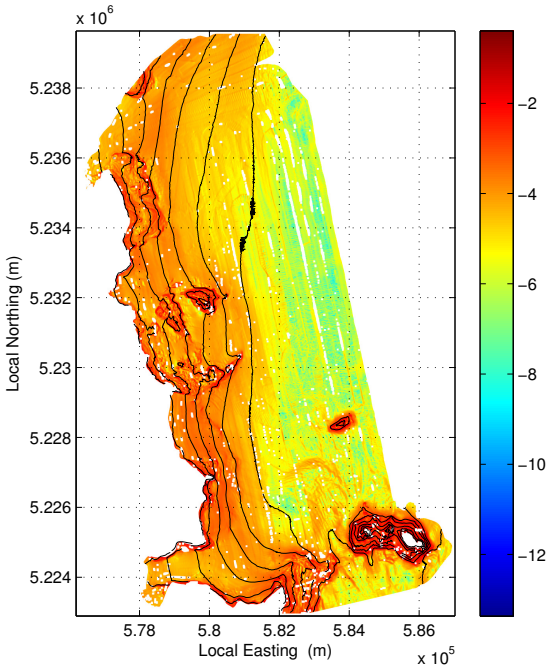


(a) Log slope, scale 2 ($8m$)

(b) Log slope, scale 4 ($14.4m$)



(c) Log slope, scale 8 ($27.2m$)



(d) Log slope, scale 16 ($52.8m$)

Figure 4.8 – Log slope for southeastern Tasmania calculated at four scales. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.

4.4.2 Surveys

As a part of the southeastern Tasmania monitoring program, high-resolution photographs of the seafloor were collected using an ocean-going AUV. One of the aims of the deployment was to "*examine the fine-scale relationships between the marine flora and fauna and the physical nature of these seabeds*" [64].

The AUV, called *Sirius*, is operated by the ACFR at the University of Sydney. The vehicle is a modified version of the SeaBED AUV [89], built and developed at the Woods Hole Oceanographic Institution. *Sirius* is approximately 2.0m long, 1.5m tall, weighs about 200kg and was designed for high-resolution, geo-referenced imaging. The payload of sensors *Sirius* currently uses is shown in Figure 4.9.

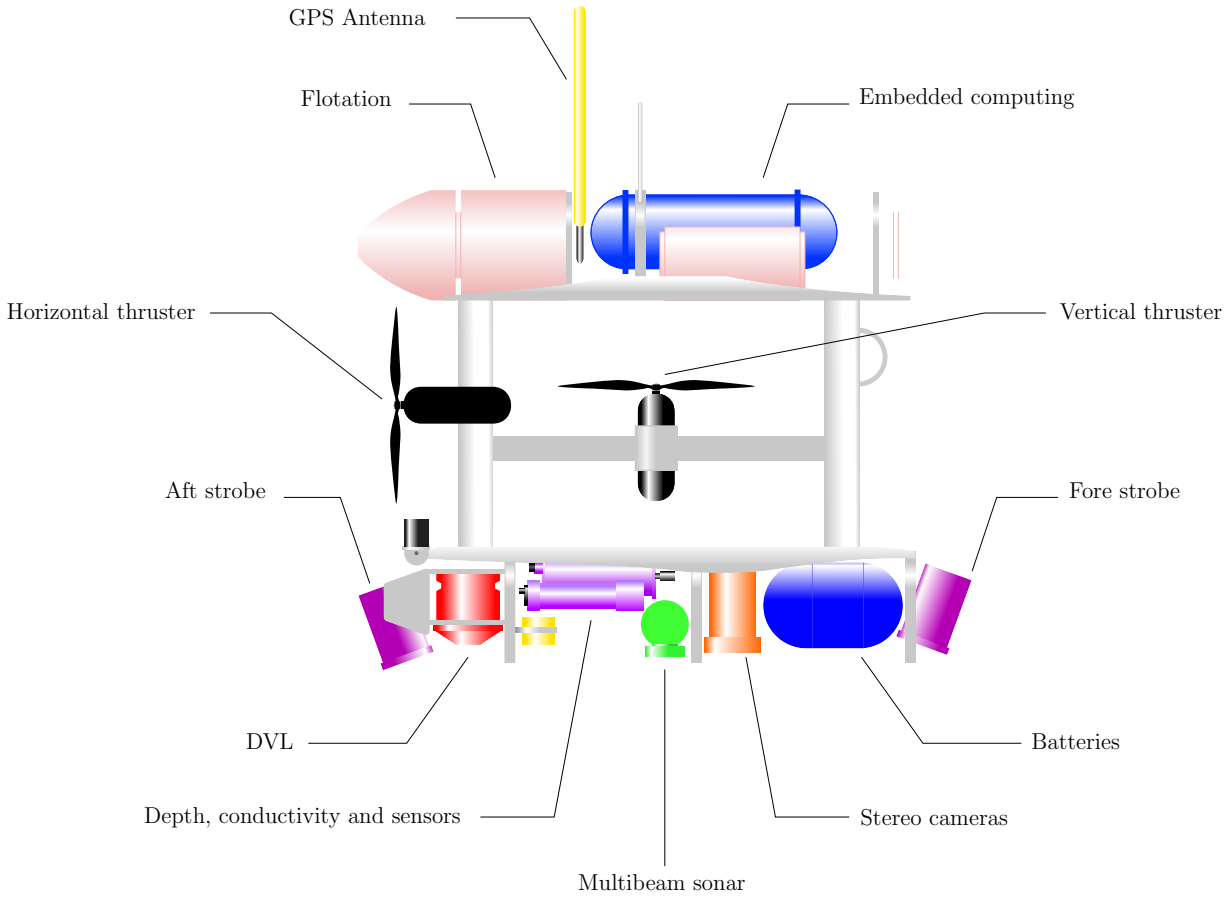


Figure 4.9 – Exposed view of *Sirius* with the fairings removed. Internal components of the vehicle are labelled and coloured differently.

Sirius generates a navigation solution in real time using observations obtained through its GPS receiver, Doppler velocity log (DVL), depth sensor and its ultrashort baseline (USBL) acoustic positioning system. Measurements of heading and attitude are provided by a digital compass with integrated roll and pitch sensors. The navigation solution is produced by fusing the sensor data using an extended Kalman filter [99].

Observations provided by the DVL, depth sensor and compass are fused into the navigation solution directly. The support vessel collects range and bearing observations to *Sirius* using the USBL system. These observations are combined with the position and orientation of the support vessel and sent back to *Sirius*, using the USBL modem, where they are fused into the navigation solution. This navigation suite allows *Sirius* to be positioned within one meter of its intended survey location [98]. The accuracy and self-consistency of the vehicle trajectory is refined offline using visual simultaneous localisation and mapping (SLAM). The sensors used in navigation are listed in Table 4.2.

Sensor	Description
Attitude and heading	TCM2 compass/tilt sensor
Depth	Digiquartz pressure sensor
DVL	Teledyne RDI 1200Khz Navigator
Altitude	RDI navigator
USBL	TrackLink 1,500 high accuracy (HA)
GPS receiver	U-Blox receiver
Optical camera	Two Prosilica 12-b cameras in a stereo configuration

Table 4.2 – Summary of navigation sensors used on *Sirius*. Table reproduced from [98].

The navigation system used by *Sirius* allows observations to be accurately geolocated. This accuracy allows the AUV imagery to be more precisely coupled with the ship-borne MBES data than the competing methods listed in Section 4.2.2. Pairing navigation error on the order of one metre and a bathymetry raster with a resolution of 1.6 metres (Section 4.4.1) will lead to registration errors no more than several pixels. On a large-scale, this small error will not introduce significant errors into subsequent habitat maps.

(a) *Sirius* on board the *R/V Challenger*

(b) Recovering *Sirius*

Figure 4.10 – *Sirius* and the *R/V Challenger* in the southeast Tasman peninsular. **(a)** *Sirius* ready for deployment on board the *R/V Challenger*. **(b)** *Sirius* being recovered after a successful deployment.

Deployments were conducted over a 10 day period in October 2008 and are shown in Figure 4.11. Detailed views of the surveys are shown in Figures 4.12 and 4.13. *Sirius* was programmed to perform surveys at an altitude of $2m$ and at a speed of $0.5m/s$. Summary statistics for each survey conducted in the southeast Tasmanian peninsula are shown in Tables 4.3 and 4.4.

Survey Name	Duration (h)	Distance (km)	Images Captured
waterfall_05	3.32	5.20	11940
waterfall_06	1.98	2.91	7131
ohara_07	3.13	4.78	11278
ohara_20	1.82	2.81	6564
patchreefnorth_08	1.93	2.69	6942
littlehippoN_11	1.87	2.66	6727
littlehippoSE_12	1.89	2.77	6787
ChevronRockN_10	1.78	2.73	6394
ChevronRockS_14	2.15	3.02	7737
hippoN_09	1.71	2.58	6139
hippoS_13	1.91	2.81	6860
Total	**23.47**	**34.96**	**84499**

Table 4.3 – Summary statistics of *Sirius* surveys in southeastern Tasmania.

Survey Name	Depth (m)			Altitude mean (m)
	min	mean	max	
waterfall_05	30.90	48.15	66.50	2.21
waterfall_06	31.76	50.17	64.56	2.50
ohara_07	31.56	54.48	75.80	2.26
ohara_20	30.24	54.77	74.79	2.29
patchreefnorth_08	59.36	74.29	83.86	2.49
littlehippoN_11	42.74	66.50	81.28	2.21
littlehippoSE_12	25.86	58.34	84.05	2.98
ChevronRockN_10	39.29	73.71	86.45	2.48
ChevronRockS_14	25.52	54.31	87.47	2.93
hippoN_09	41.69	71.14	88.82	2.32
hippoS_13	33.03	69.96	94.29	2.89

Table 4.4 – Depth statistics of *Sirius* surveys in southeastern Tasmania.

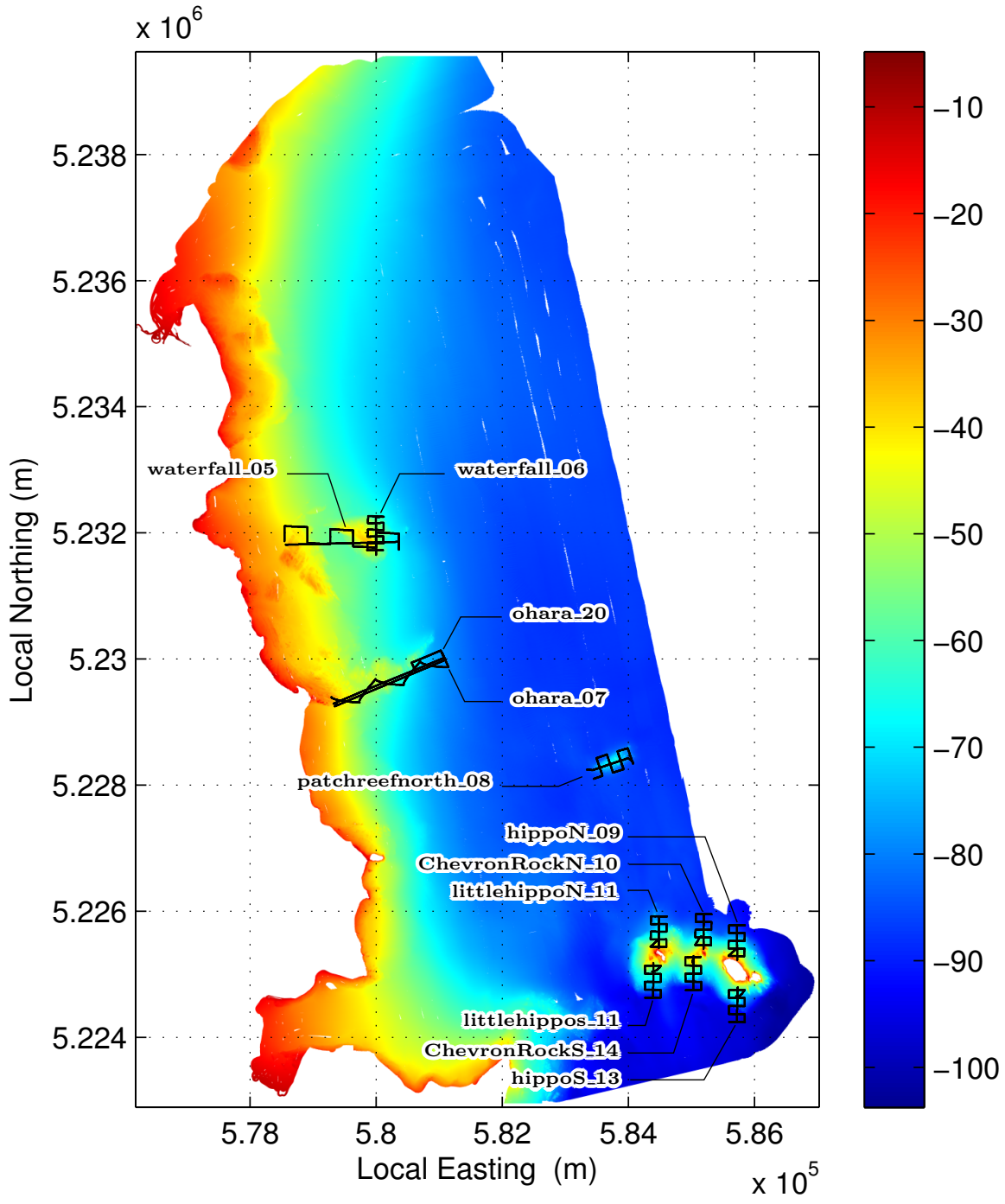
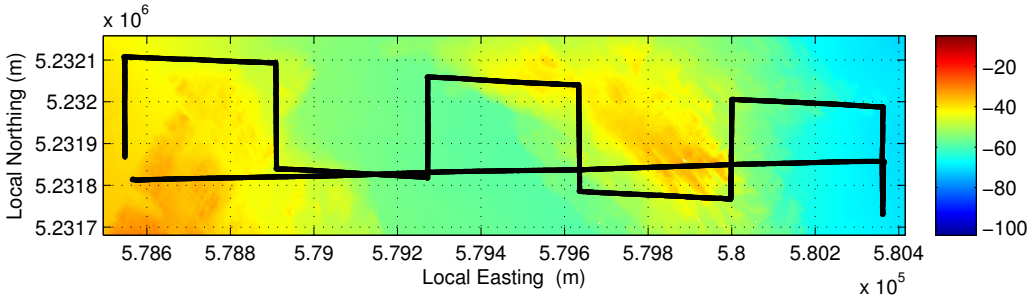
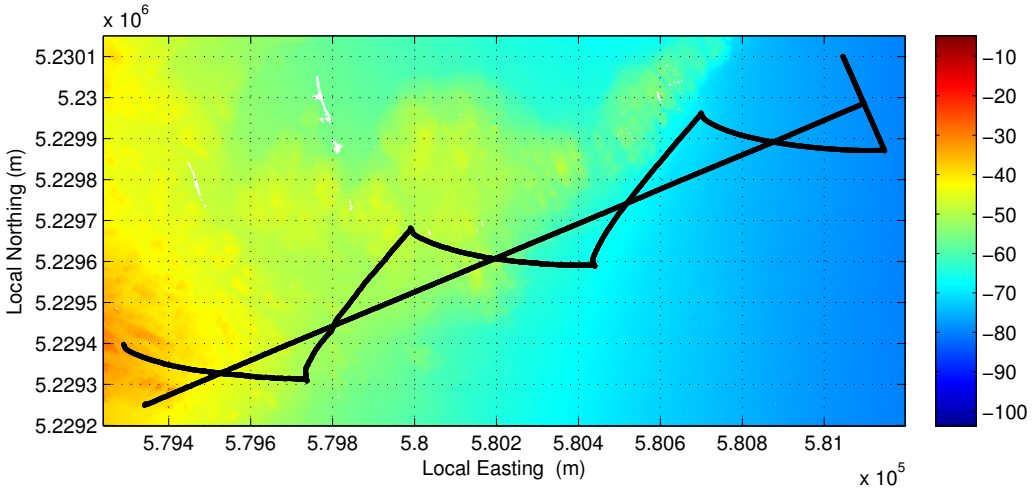


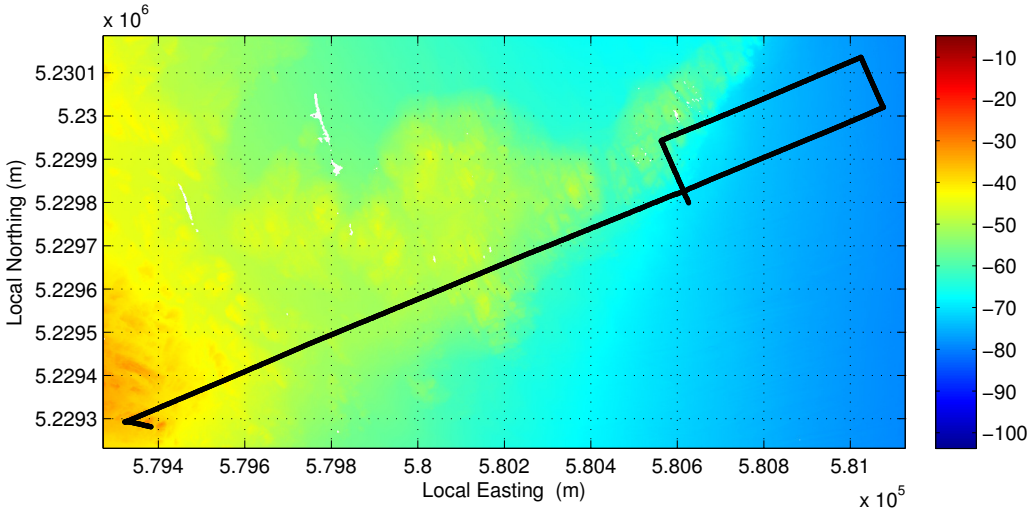
Figure 4.11 – Surveys conducted by *Sirius* in southeastern Tasmania. Each survey is labelled with their survey name. The survey names are linked to the corresponding detailed figures in Figures 4.12 and 4.13. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.



(a) waterfall_05

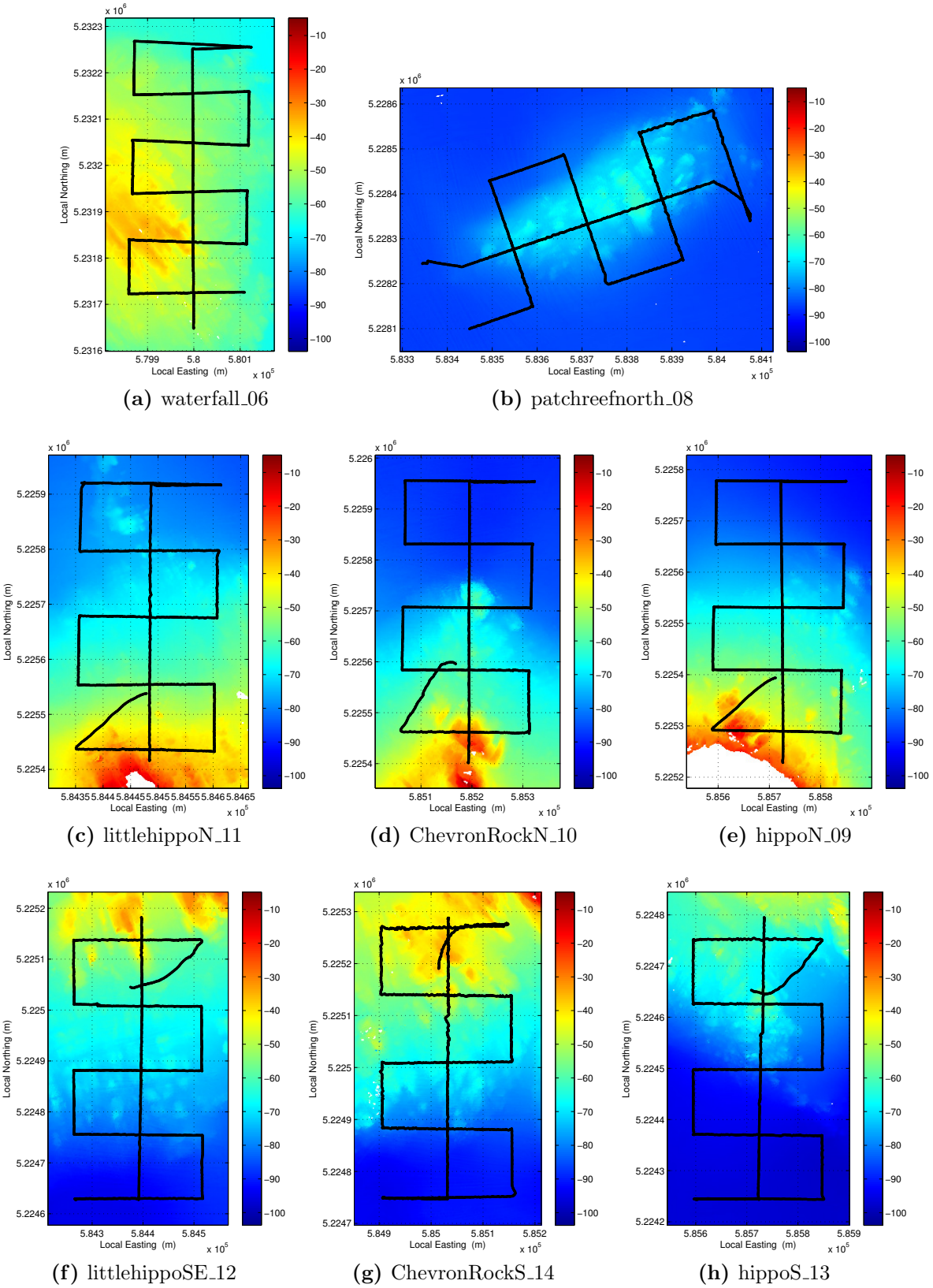


(b) ohara_07



(c) ohara_20

Figure 4.12 – Detail of AUV surveys conducted in southeastern Tasmania. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.



(a) waterfall_06 **(b)** patchreefnorth_08

(c) littlehippoN_11 **(d)** ChevronRockN_10 **(e)** hippoN_09

(f) littlehippoSE_12 **(g)** ChevronRockS_14 **(h)** hippoS_13

Figure 4.13 – Detail of AUV surveys conducted in southeastern Tasmania. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.

4.4.3 Categorising the In-situ Imagery

Although AUVs can collect high-resolution and accurately geo-located in-situ data, the volumes of data they collect can quickly overwhelm human analysts. As shown in Table 4.3, nearly 85,000 images were captured. In principle, the AUV imagery could be classified by a human expert. However, the labour requirements of analysing all the images is prohibitive.

In this thesis, the labour requirements of processing large volumes of imagery is replaced with an unsupervised variational Dirichlet process (VDP) model [46]. The VDP model is Bayesian and non-parametric with the attractive property that it does not require the number of clusters to be specified in advance. The VDP has been shown to be effective at producing clusters which provide a useful summary of large volumes of marine images [91]. The method presented in [91] is used in this thesis to cluster AUV imagery.

Prior to clustering the AUV imagery, each image must be represented as a single data point. In [91] each AUV image is represented using both image appearance features and terrain complexity features. For convenience, the image features used in [91] are repeated in this section.

The overall visual content of an AUV image is described using measures of texture and colour. Image texture is measured using Local Binary Patterns (LBPs), which are invariant against monotonic transformations in illumination. Colour is measured using summary statistics from performing mean shift image segmentation in the L*a*b colour space. The summary statistics include the average segment size, the mean of the segment colours and the standard deviation of the segment colours. The average segment size is calculated in pixels and normalised by the number of pixels in the image. The mean and standard deviation of the segment colours are produced by calculating the L*a*b colour mode for each segment and computing the mean and standard deviation using the resultant colour modes.

Since *Sirius* is precisely navigated and collects stereo imagery, it is possible to create detailed three-dimensional models of the seafloor in the form of textured polygonal meshes [39]. Terrain complexity features can be derived from these Delaunay triangulated meshes using the method proposed in [25]. In particular, rugosity and slope are used to quantify the three-dimensional content of an image.

Combining the image appearance and terrain complexity features produces an AUV image feature vector containing 23 dimensions. The AUV image feature vector is summarised in Table 4.5. For more detail on this set of features refer to [91].

The bathymetry feature vector detailed in Section 4.4.1 contains similar features to the image features used during clustering. Both feature vectors contain log rugosity and log slope. The pairwise Spearman correlation coefficients for log rugosity and log slope, derived from both bathymetry and stereo-imagery, are shown in Tables 4.6a and 4.6b respectively.

The scales of rugosity are all highly correlated when bathymetry or image features are considered in isolation. Rugosity derived from bathymetry is only weakly correlated with rugosity derived from the stereo-imagery. The same is true of slope. The polygonal meshes which define the three-dimensional reconstruction of stereo-imagery provide a spatial resolution of less than a meter. The bathymetry raster is limited to a spatial resolution of $1.6m$. While both data sets are observing the same phenomenon, they do so at very different resolutions and scales. This leads to only a weak correlation between the two data sets.

To avoid poor illumination in the imagery, images below an altitude of 0.2m and above an altitude of 3m are removed. For the AUV data shown in Figure 4.11, approximately 70,000 images remain when high and low altitude images are removed. To speed up the clustering process, the AUV image feature vector is reduced to 5 dimensions using PCA and standardised by subtracting the mean from each dimension and dividing by the standard deviation. The VDP detected 25 clusters in the AUV imagery. Example images randomly sampled from each of the 'habitat' clusters are shown in Figure 4.14. Rather than manually classifying all 70,000 images, a human expert is simply required to check that a relatively small set of clusters represents valid habitat 'proxies'.

Feature	Feature scale	Number of dimensions
log(rugosity - 1)	image	1
log(rugosity - 1)	$5 \times 5m$	1
log(slope)	$5 \times 5m$	1
log(rugosity - 1)	$10 \times 10m$	1
log(slope)	$10 \times 10m$	1
mean(L*a*b* segment modes)	image	3
st. dev(L*a*b* segment modes)	image	3
log(mean(L*a*b* segment size))	image	1
st. dev(Grey-scale image pixels)	image	1
LBPs(radius of 1, 8 samples)	image	10

Table 4.5 – Summary of image appearance and terrain complexity features used to describe AUV images. Table reproduced from [91].

Bathymetry ($8m$)	1.0000	0.9351	0.8758	0.8323	0.2447	0.2261	0.2365
Bathymetry ($14.4m$)	0.9351	1.0000	0.9596	0.9140	0.2975	0.2706	0.2810
Bathymetry ($27.2m$)	0.8758	0.9596	1.0000	0.9696	0.3275	0.3005	0.3121
Bathymetry ($52.8m$)	0.8323	0.9140	0.9696	1.0000	0.3403	0.3167	0.3275
Image (image)	0.2447	0.2975	0.3275	0.3403	1.0000	0.7454	0.6800
Image ($5m$)	0.2261	0.2706	0.3005	0.3167	0.7454	1.0000	0.8975
Image ($10m$)	0.2365	0.2810	0.3121	0.3275	0.6800	0.8975	1.0000

(a) Pairwise Spearman correlation coefficients for log rugosity features

Bathymetry ($8m$)	1.0000	0.9221	0.8399	0.7708	0.2991	0.3076
Bathymetry ($14.4m$)	0.9221	1.0000	0.9325	0.8432	0.3111	0.3237
Bathymetry ($27.2m$)	0.8399	0.9325	1.0000	0.9233	0.3161	0.3272
Bathymetry ($52.8m$)	0.7708	0.8432	0.9233	1.0000	0.3313	0.3455
Image ($5m$)	0.2991	0.3111	0.3161	0.3313	1.0000	0.8212
Image ($10m$)	0.3076	0.3237	0.3272	0.3455	0.8212	1.0000

(b) Pairwise Spearman correlation coefficients for log slope features

Table 4.6 – Pairwise Spearman correlation coefficients for bathymetry and image features. **(a)** Pairwise Spearman correlation coefficients for log rugosity features. **(b)** Pairwise Spearman correlation coefficients for log slope features. The first four features are derived from the bathymetry. The remaining features are derived from the stereo-imagery. Red entries are pairwise correlations between bathymetry features. Blue entries are pairwise correlations between image features. Black entries are pairwise correlations between bathymetry and image features.

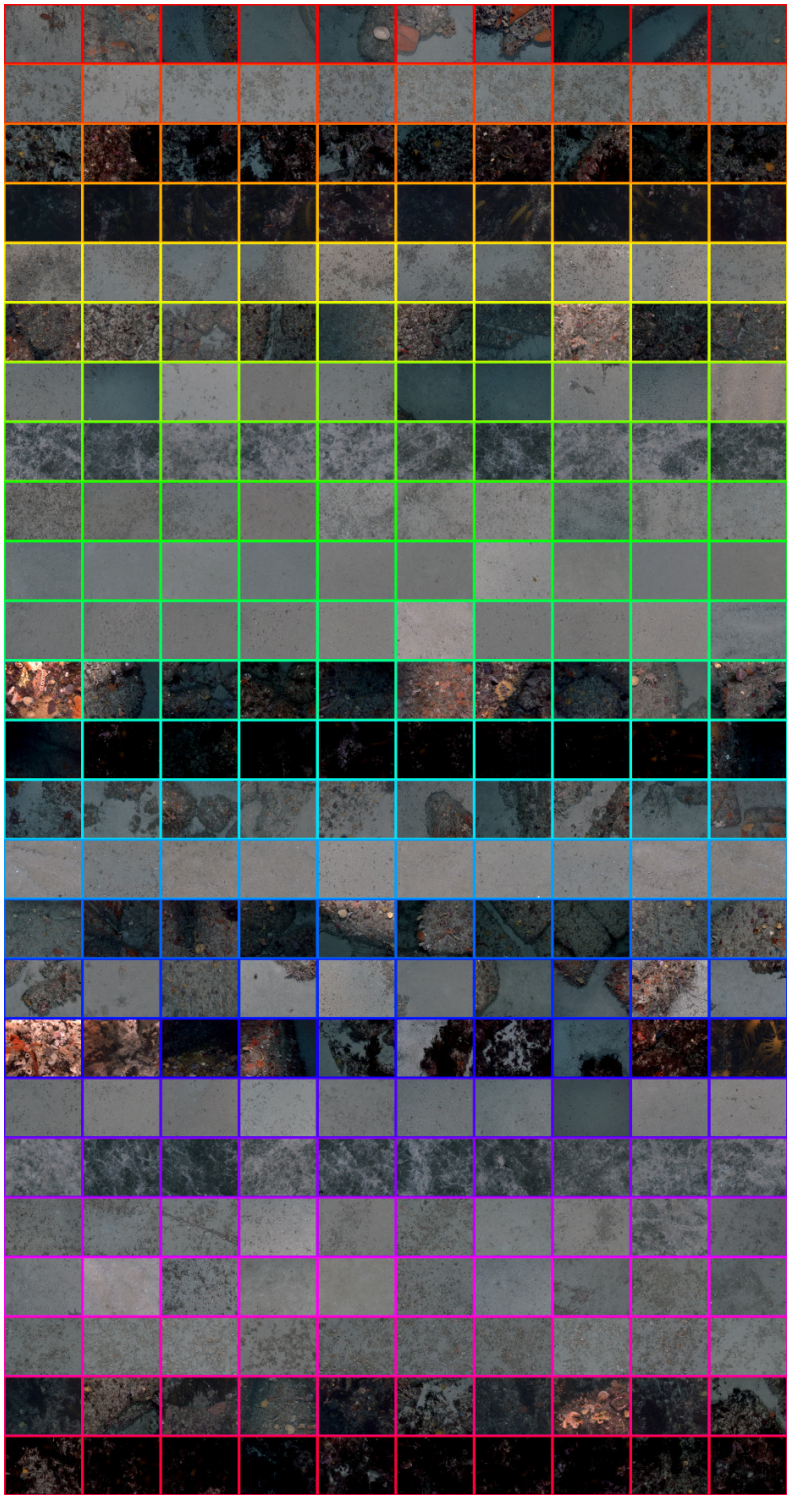


Figure 4.14 – Example images drawn randomly from the habitat clusters. Prior to human verification and merging, the VDP clusters are not associated with semantic labels.

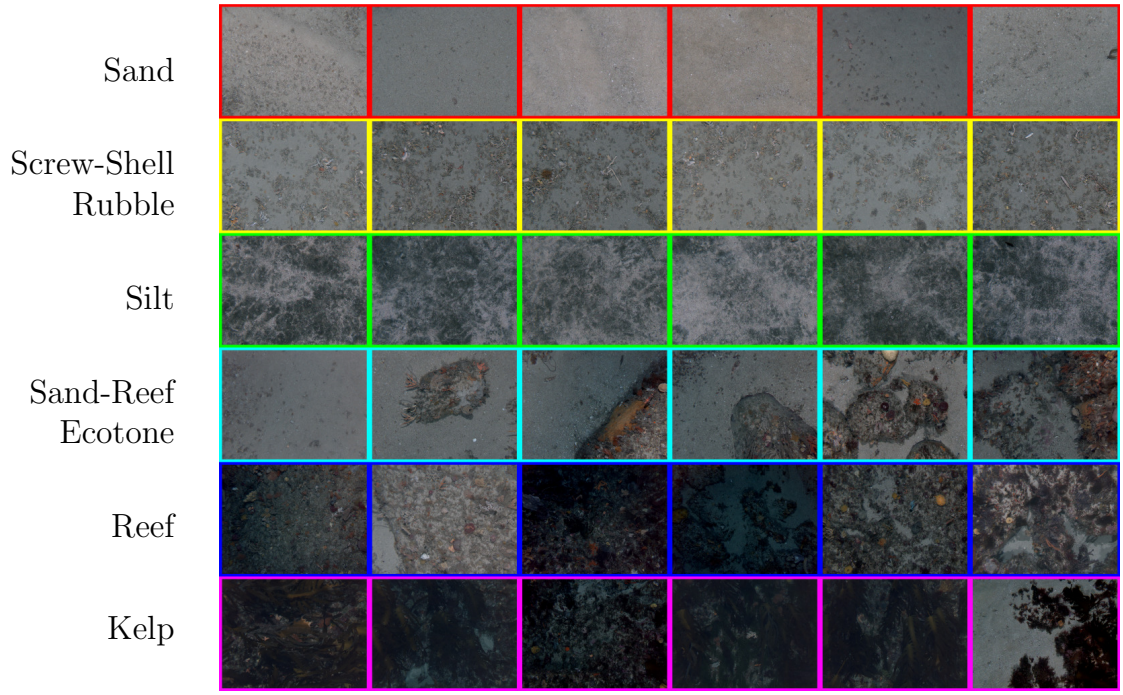


Figure 4.15 – Example images drawn randomly from the habitat classes. Once the VDP clusters have been verified and merged by a human, they can be assigned a semantic label and considered classes.

For the purposes of creating bathymetric habitat maps, the AUV imagery has been over clustered. Clusters which appear distinct in the image feature space may be inseparable in the bathymetry feature space. For example, clusters which are differentiated by colour or textures which occur on a scale less than the bathymetry resolution are not likely to produce a distinct signal in the bathymetry. To reduce the computational demands on subsequent modelling stages and reduce confusion in the training data, similar clusters are merged.

Six habitat categories remain after the clusters shown in Figure 4.14 have been reviewed and consolidated. The consolidated habitat categories can now be given semantic labels and considered habitat classes as shown in Figure 4.15. The spatial contiguity (Figure 4.16) and visual content (Figures 4.15, 4.17 and 4.18) of the habitat classes suggests that after human approval, the VDP has captured a distinct and meaningful categorisation of biotic and abiotic groups within the environment.

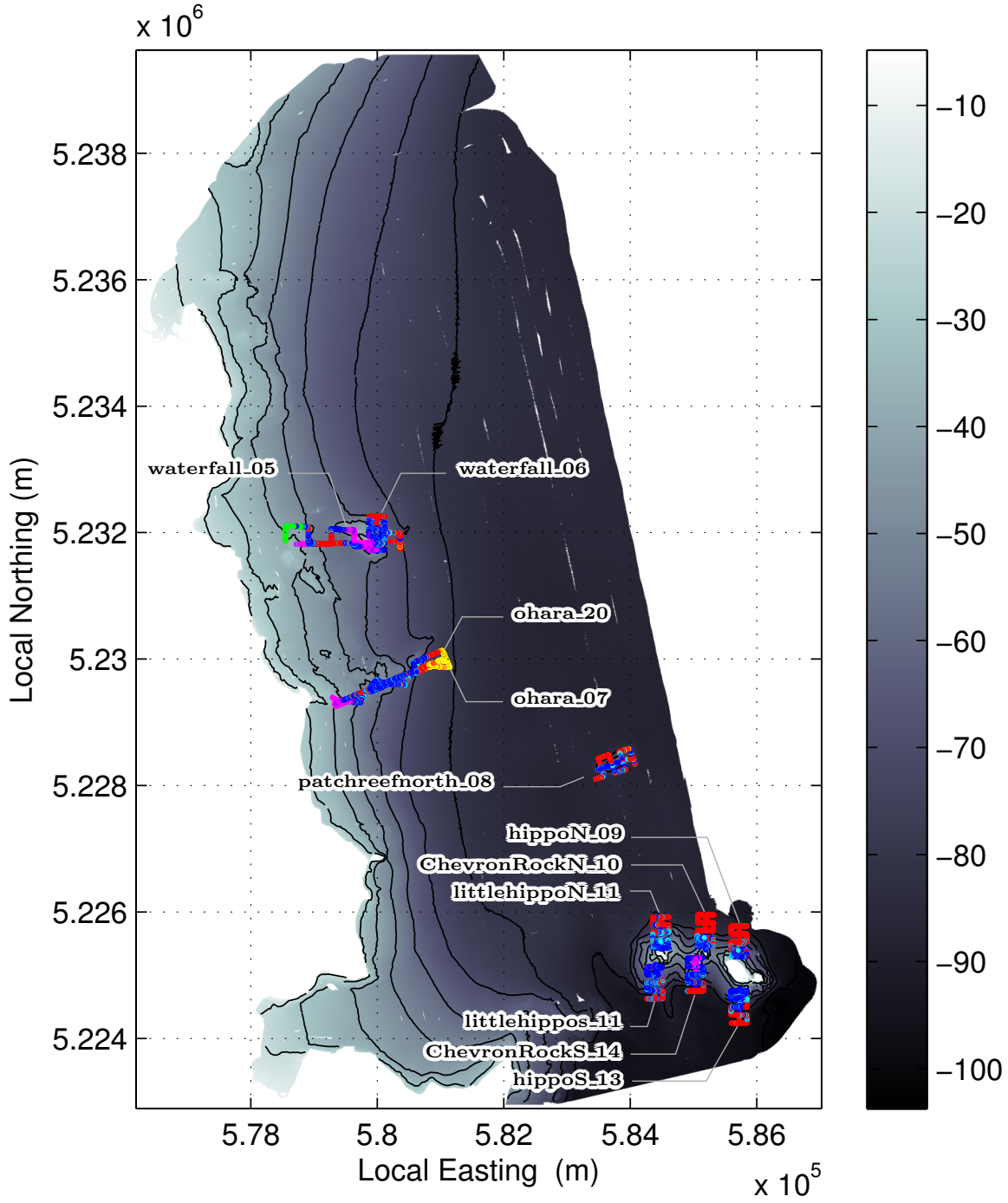
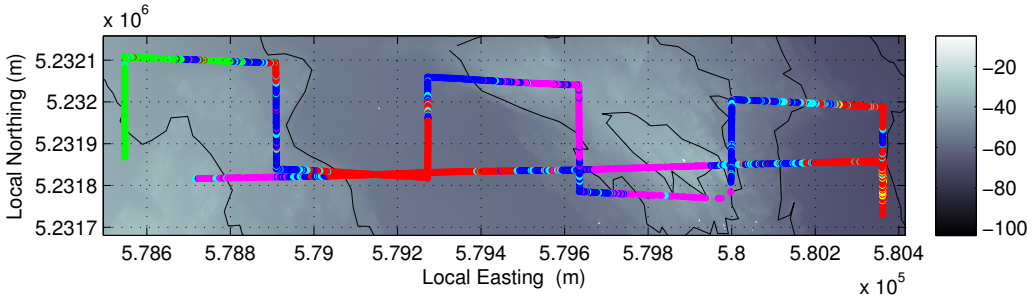
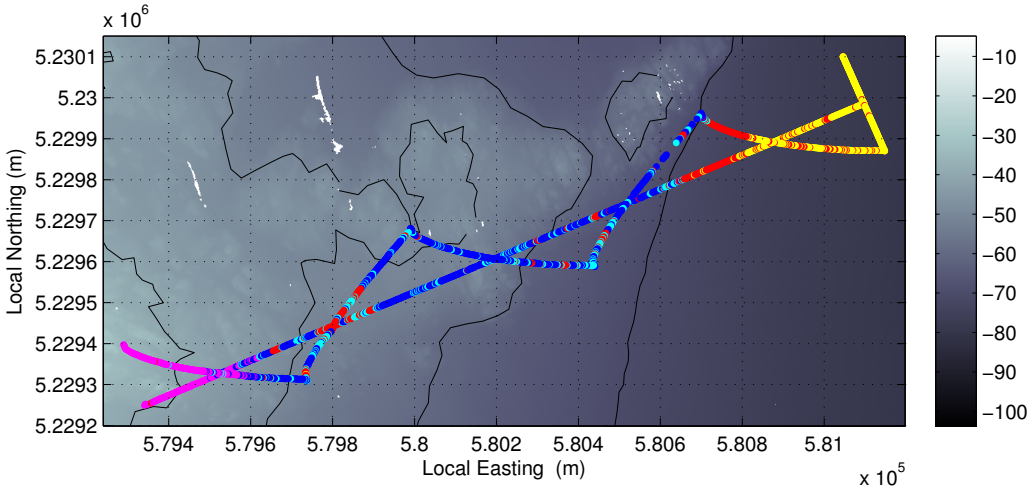


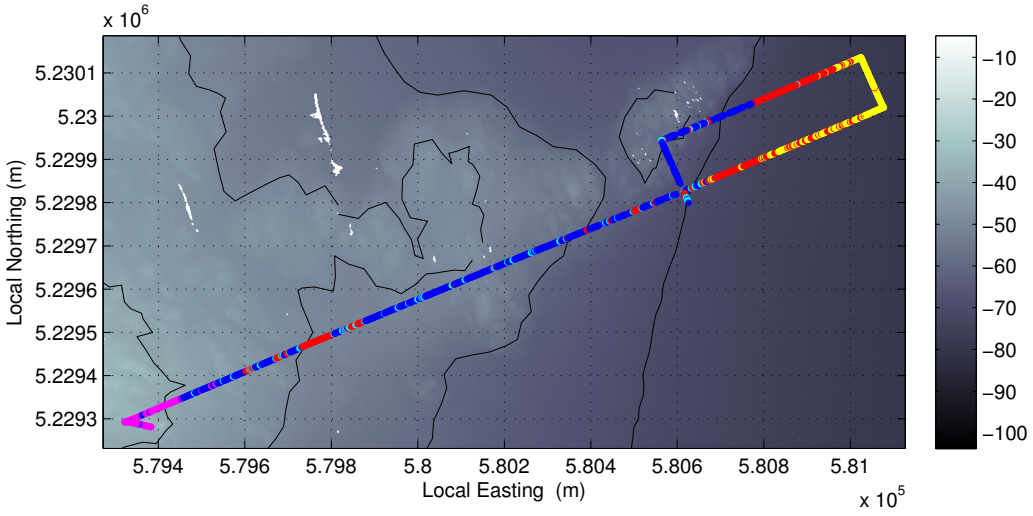
Figure 4.16 – Classified AUV surveys conducted in southeastern Tasmania. The intensity plot and black lines represent the bathymetry and depth contours. Each AUV pose is shown as a coloured dot where the colours indicates a specific habitat class. The colours for each habitat class are shown in Figure 4.15. Each survey is labelled with their survey name. The survey names are linked to the corresponding detailed figures in Figures 4.17 and 4.18. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.



(a) waterfall_05

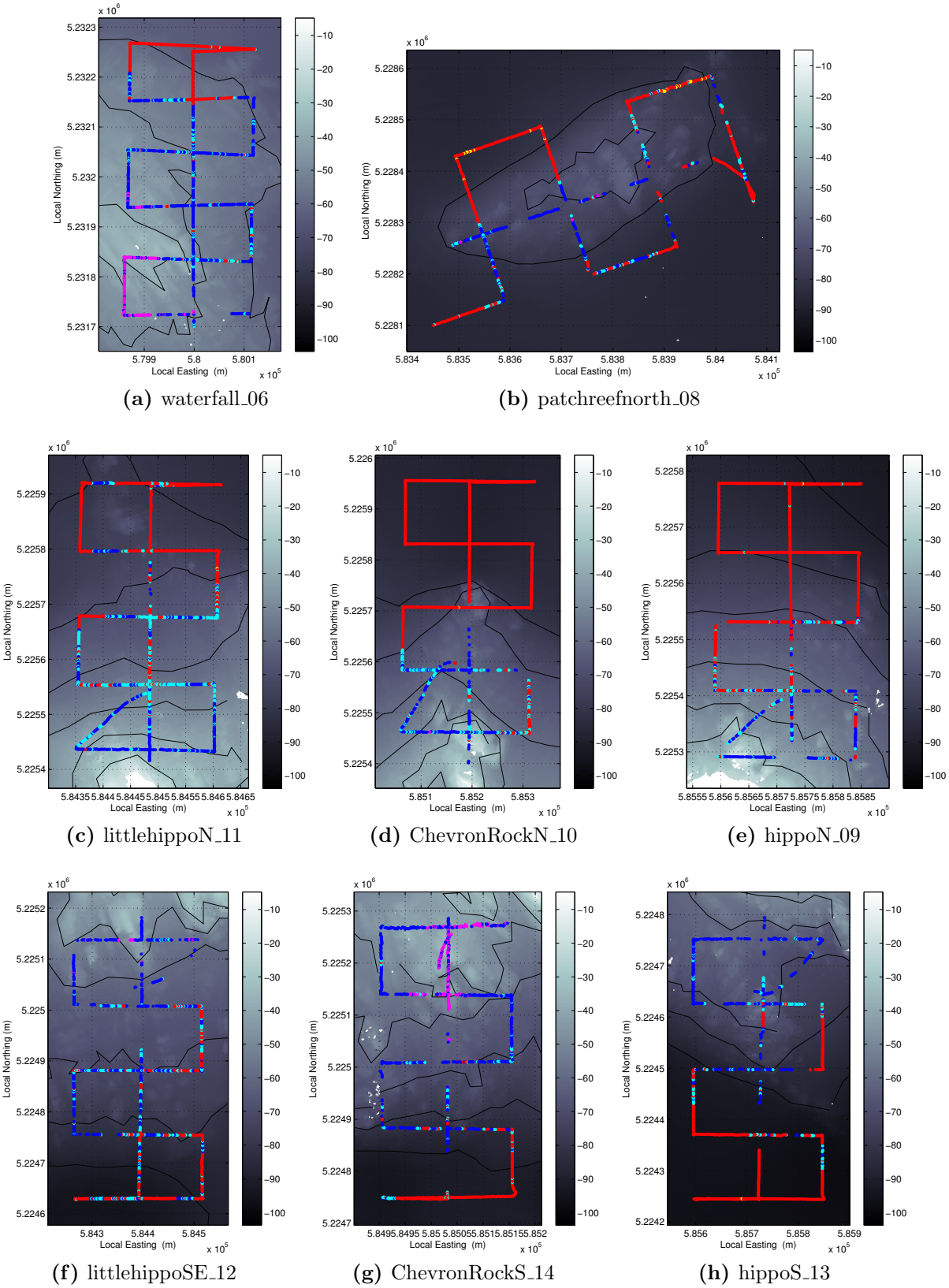


(b) ohara_07



(c) ohara_20

Figure 4.17 – Detail of classified AUV surveys conducted in southeastern Tasmania. Each AUV pose is shown as a coloured dot where the colours indicate a specific habitat class. The colours for each habitat class are shown in Figure 4.15. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.



(a) waterfall_06 **(b)** patchreefnorth_08

(c) littlehippoN_11 **(d)** ChevronRockN_10 **(e)** hippoN_09

(f) littlehippoSE_12 **(g)** ChevronRockS_14 **(h)** hippoS_13

Figure 4.18 – Detail of classified AUV surveys conducted in southeastern Tasmania. Each AUV pose is shown as a coloured dot where the colours indicate a specific habitat class. The colours for each habitat class are shown in Figure 4.15. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.

4.4.4 Categorising the Bathymetry

To create a habitat model of the bathymetry, the in-situ observations need to be coupled with the multibeam bathymetry. Aligning the AUV observations and the multibeam bathymetry is trivial as both data products are geo-referenced. Before it is possible to create a habitat model, each bathymetry pixel visited by the AUV must be assigned to a habitat class. A graphical example of assigning bathymetry pixels, observed by an AUV, to habitat classes is shown in Figure 4.19.

After autonomously categorising the images, the AUV observations are assigned a probability of belonging to each of the habitat categories. The probability of the AUV observations shown in Figure 4.19 belonging to one of six habitat classes is shown in Table 4.7.

AUV pose	Sand (class 1)	Screw-Shell Rubble (class 2)	Silt (class 3)	Sand-Reef Ecotone (class 4)	Reef (class 5)	Kelp (class 6)
1	0.817	0.000	0.000	0.183	0.000	0.000
2	0.749	0.000	0.000	0.251	0.000	0.000
3	0.000	0.000	0.000	1.000	0.000	0.000
4	0.000	0.000	0.000	0.987	0.013	0.000
5	0.000	0.000	0.000	0.005	0.995	0.000
6	0.000	0.000	0.000	0.005	0.995	0.000
7	0.000	0.000	0.000	0.023	0.977	0.000
8	0.000	0.000	0.000	0.115	0.885	0.000

Table 4.7 – Probability of the AUV observations shown in Figure 4.19 belonging to one of six habitat classes. The AUV pose numbers correspond to the labels shown in Figure 4.19.

The habitat label associated with each image is given by the most likely class probability for each observation. In the example given in Table 4.7, the first two observations share the label *Sand*, the following two observations share the label *Sand-Reef Ecotone* and the remaining four observations share the label *Reef*.

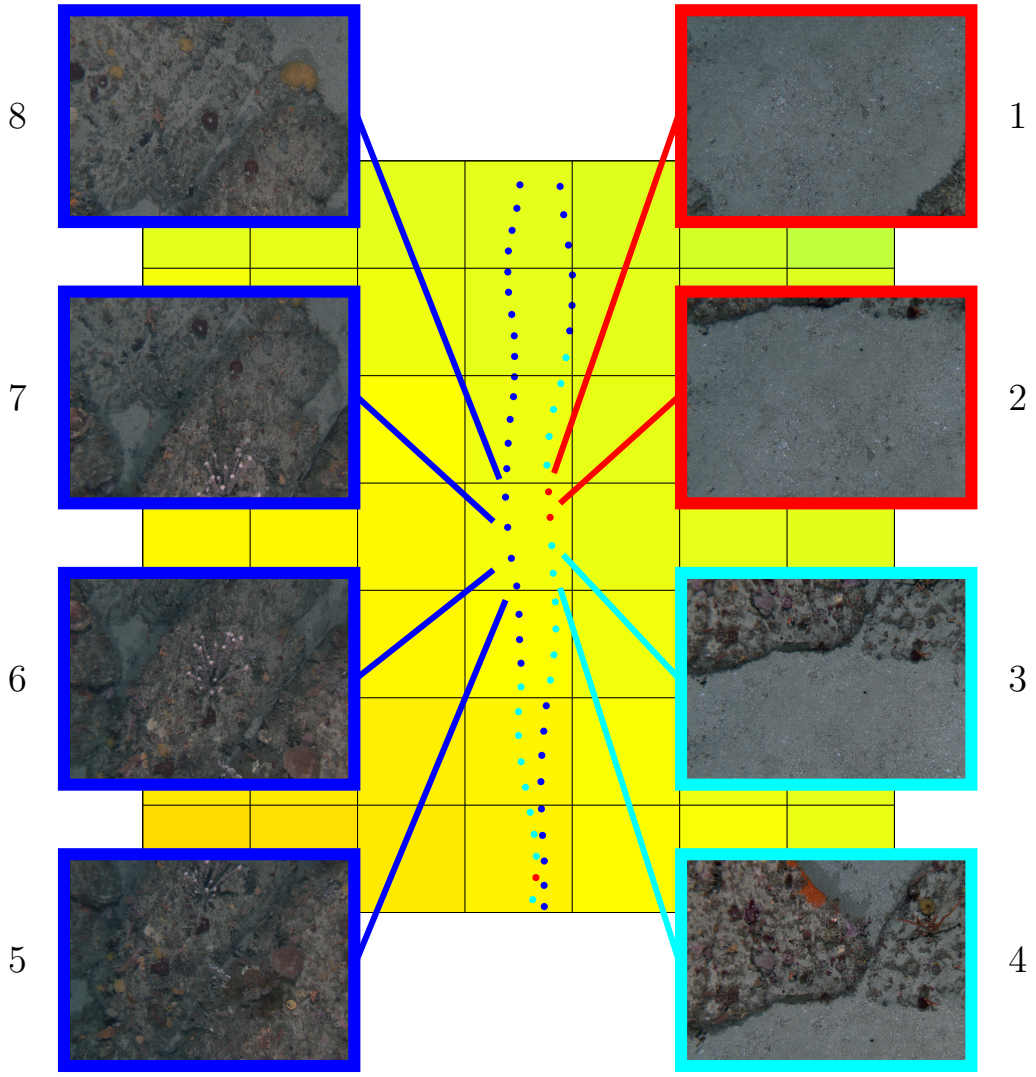


Figure 4.19 – Demonstration of registration between AUV in-situ observations and bathymetry. The data shown is from an intersecting region of the *waterfall_05* (Figure 4.12a) and *waterfall_06* (Figure 4.13a) surveys. The bathymetry raster is shown in the background. The grid lines represent the boundaries between bathymetry pixels. The coloured dots represent AUV observations. Each AUV observation consists of an illuminated image of the seafloor. Observations of the central pixel are shown on the left and right of the figure. The labels next to each image identify the AUV poses which fall into the central pixel. The colours of each AUV pose and image correspond to different habitat classes. The red class represents sand, the cyan class represent a sand-reef boundary and the dark blue class represents reef. To generate a habitat map, each bathymetry pixel observed by the AUV must be assigned to a habitat class from multiple, possibly conflicting, observations.

To label the bathymetry pixel observed by the eight AUV observations shown in Table 4.7 and Figure 4.19, consensus between the multiple conflicting observations must be found. If the habitat label for each observation is used to determine consensus, the mode of the observed labels can be used. The mode of the habitat labels would yield a label of *Reef*.

The lower the spatial resolution of the bathymetric data, the more likely it will contain a mixture of observed classes. The spatial resolution of the Tasmania bathymetry ($1.6m$) is large enough to span multiple habitat classes. As Figure 4.19 shows, the observed habitat class will vary depending on which part of the bathymetry pixel is observed. Describing a bathymetry pixel with a single label ignores the ambiguity introduced by conflicting accounts of what was observed. This is particularly true of bathymetry pixels which observe the transition from one habitat to another.

An alternative to using the mode of the habitat labels to settle conflicting accounts is to use the class probabilities. Averaging the class probabilities of the observed data in a bathymetry pixel produces an estimate of the mix of habitat classes. The average class probabilities for all eight observations shown in Table 4.7 are shown in Table 4.8.

Sand (class 1)	Screw-Shell Rubble (class 2)	Silt (class 3)	Sand-Reef Ecotone (class 4)	Reef (class 5)	Kelp (class 6)
0.1958	0.0000	0.0000	0.3209	0.4833	0.0000

Table 4.8 – Mean probability of the central bathymetry pixel, shown in Figure 4.19, belonging to one of six habitat classes.

Table 4.8 shows that the most likely class is *Reef*. However, the average class probability for *Reef* is low and fairly closely followed by *Sand-Reef Ecotone*. This averaged probability captures the conflicting accounts presented by the eight observations of the bathymetric pixel. Rather than belonging to a single habitat class, the bathymetric cell contains a mixture of habitat classes.

4.5 Experimental Results

In this section a probabilistic least squares classifier trained on discrete class labels (PLSC-VDP) and a probabilistic least squares classifier trained on continuous class probabilities (PTLSC) are compared. Both models are described in Chapter 3. The models are applied to the task of generating a bathymetric habitat map using the data set described in Section 4.4.

The bathymetric feature vector contains depth, log rugosity and log slope extracted from the DEM using window sizes of 2 ($8m$), 4 ($14.4m$), 8 ($27.2m$) and 16 ($52.8m$) (see Figure 4.7, Figure 4.8). The bathymetric feature vector is standardised by subtracting the mean from each dimension and dividing by the standard deviation. The AUV observations are clustered and consolidated as described in Section 4.4.3. The bathymetry pixels observed by the AUV are labelled by averaging the class probabilities of all AUV observations within a single pixel as described in Section 4.4.4. After this procedure, 19,450 labelled bathymetry pixels are available for training a model.

4.5.1 Cross-Validation

The performance of the PLSC-VDP and the PTLSC is measured using accuracy, mean squared error (MSE) of the predictive density and mean variance of the latent functions. The accuracy is calculated by recording the portion of correctly classified instances in the test data, relative to the VDP cluster labels. The MSE of the predictive density is measured between the 'true' human verified VDP class probabilities and the predicted probabilities. Mean variance is calculated by averaging the variance measured across all training instances and classes.

To provide an indication of model generalisation and limit the training data to a tractable size, performance is evaluated using a variant of cross-validation. Repeated random sub-sampling is used where a subset of data is randomly selected from the pool of observations for training, and the remaining data are used for validation.

Stratified sampling is used to randomly sample the training data. Each class is allocated the same number of training samples. Samples drawn from a class are also stratified according to the probability of class membership. Observations within a class are grouped into probability bins of 0.01. The number of observations allocated to the class are distributed such that an equal number of observations are sampled randomly from each probability bin. Any remaining observations, due to integer rounding errors or probability bins with a low numbers of observations, are sampled randomly from the pool of remaining observations in the class. This is done for each class. The goal is to provide a training data set with equal numbers of observation for each class and probability.

After a fold of training data has been selected, testing data is selected from the remaining observations. This ensures that no training and testing fold pair share any common observations. Three methods are used to select testing data, uniform random sampling, the most likely data and the least likely data. Selecting the most likely and the least likely data are deterministic sampling methods. Since the testing data is selected randomly, the pool of remaining observations is also random leading to variation in the deterministic sampling methods. The most likely and least likely observations are determined by the human verified VDP class probabilities. The probability of the most likely class is used to order the observations.

The performance of the PLSC-VDP and the PTLSC is quantified using 2000 observations for training and 10,000 observations for testing. The mean performance calculated over the ten folds is shown in Table 4.9 for the most likely observations, in Table 4.10 for uniform randomly sampled testing data and in Table 4.11 for the least likely observations.

In all testing data scenarios the PTLSC is able to outperform the PLSC-VDP. The most likely data contains probabilities with almost no ambiguity. They are effectively discrete labels and the models offer comparable levels of performance. The PTLSC has a small 1% accuracy advantage and offers more confident predictions with a lower mean variance.

Model	Accuracy (%)	MSE	Mean Variance
PLSC-VDP	64.472	0.082	0.353
PTLSC	65.452	0.082	0.216

Table 4.9 – Ten-fold cross-validation performance for the most likely data samples from the Tasmania data set.

Model	Accuracy (%)	MSE	Mean Variance
PLSC-VDP	57.704	0.075	0.347
PTLSC	61.255	0.072	0.210

Table 4.10 – Ten-fold cross-validation performance for uniform, randomly sampled testing data

Model	Accuracy (%)	MSE	Mean Variance
PLSC-VDP	50.269	0.073	0.341
PTLSC	57.557	0.064	0.203

Table 4.11 – Ten-fold cross-validation performance for the least likely data sampled from the Tasmania data set.

In uniform randomly sampled test data the test locations contain a mix of high and low probability data. Predictions in the PTLSC are produced with a lower MSE and mean variance than the PLSC-VDP. The PTLSC is also able to offer a 3% increase in classification accuracy.

For test data sampled from the least likely observations, the PTLSC produces significantly better performance than the PLSC-VDP. The least likely observations contain many ambiguous class probabilities. In these regions the class probabilities contain subtle information which is not preserved in the discrete class labels. The PTLSC is able to make use of this information and produce a more accurate predictive distribution. A beneficial side effect of producing a more accurate predictive distribution is that the PTLSC produces a superior accuracy with a performance gain of 7%.

The cross-validation results presented in Tables 4.9 to 4.11 provide evidence that the PTLSC is able to make use of the probabilistic training data in regions where there are ambiguous class assignments. As a result the PTLSC produces a more accurate and confident predictive distribution than the PLSC-VDP. This results in a higher label accuracy. The difference in performance between the two models increases as the test data is drawn from more ambiguous areas of the environment. In the presence of uncertain test data, the performance of the PTLSC degrades gracefully and at a slower rate than the PLSC-VDP.

4.5.2 Habitat Map

Due to the $\mathcal{O}(N^3)$ complexity of training a GP [75], it is not feasible to train a GP model on all 19,450 observations. To make training tractable, the stratified sampling method is used to select 6000 observations for training. The stratified sampling method is also used to sample 15,000 observations for interpolation. It does not matter if the training and interpolation data sets share common observations. The objective is to create an accurate model using as much data as possible.

The PTLSC habitat map for the southeastern Tasmania data set is shown in Figure 4.20. The intensity plot shows the most likely habitat where the intensity of the colour is proportional to the probability. As the predictions become less certain, the colour fades to white. The mean variance of the class latent functions is shown in Figure 4.21.

The habitat map predictions closely match the observed data shown in Figure 4.16. However, in locations far from observed data predictions become weaker. This is particularly obvious at depths below $90m$. The deep, eastern region of the bathymetry raster has been predicted with a low confidence.

Intuition suggests that the deep eastern region is likely to be a sandy substrate. Most of this region has been predicted as *Screw-Shell Rubble* with some low probability *Reef*. There are no observations in this region available to support inference. As shown in Section 4.4.1, artefacts are present in the bathymetry. These artefacts contribute to

an artificial roughness in the bathymetry making the rougher *Screw-Shell Rubble* a more plausible hypothesis than *Sand*. Many of the closest observations of this region are observations of *Screw-Shell Rubble* from *ohara_07* and *ohara_20* which border on the deep and flat eastern region.

The same phenomenon can be seen in shallow regions. Few observations have been collected from shallow, flat terrain. The majority of shallow, flat bathymetry observations are from *waterfall_05* which observes *Silt*. The remaining shallow observations pass over textured *Kelp* and *Reef*. Consequently, most flat terrain shallower than $40m$ has been classified as *Silt*. Whilst some of this region is correctly identified as *Silt* much of it will likely be *Sand*.

At this point it is worth noting that the AUV observations represent a biased sample of the environment. The AUV surveys deliberately "targeted the coastal and offshore reefs" [64] of the southeastern Tasmanian peninsula. Relatively little attention was given to other habitats in the environment. The length, width and orientation of each survey was designed specifically for the reefs targeted by the survey. This targeted approach is not an efficient sampling strategy for producing broad-scale habitat maps. This is supported by the resulting habitat map. Large regions of the environment have not been explored and inference in these regions remains weak. Transects which traverse a diverse range of bathymetry features are more appropriate for general inference.

Although large sections of the environment are not supported by observations, the probabilistic model provides two levels of self-assessment. The first level of self-assessment is the predicted probability provided by the model. The predicted probability can be interpreted as the likelihood of observing a class at the input location. If the predicted probability is in its most entropic state, the input location is equally likely to generate one of the possible classes. Alternatively it can be interpreted as an indication that the input location is a mixture of the possible classes. Under this interpretation an entropic observation has been generated by an even mixing of the classes.

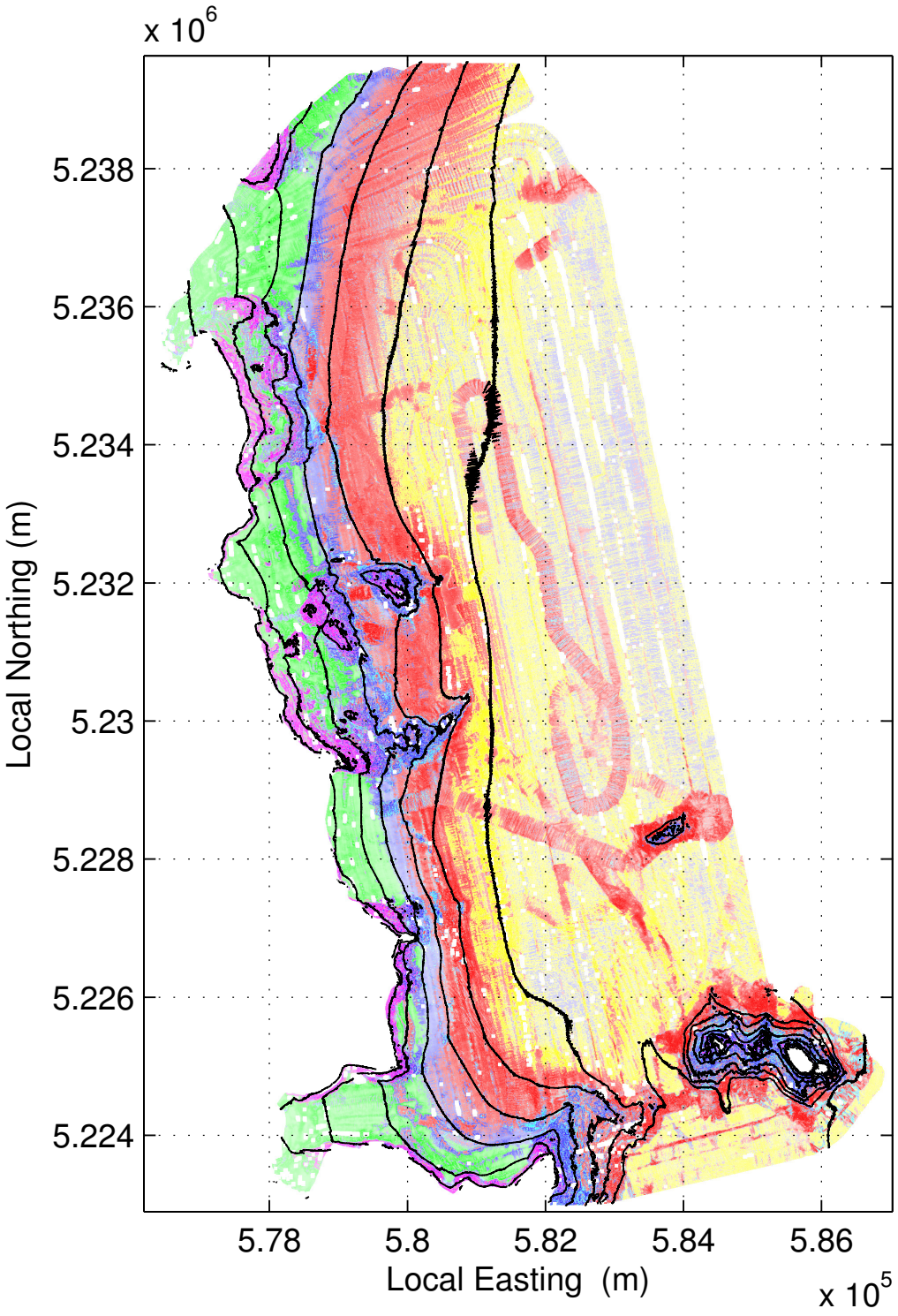


Figure 4.20 – Southeastern Tasmania habitat map generated by the probabilistic targets least squares classifier (PTLSC). The bathymetry contours are shown as black lines. The intensity plot shows the most likely habitat where the intensity of the colour is proportional to the probability of the most likely habitat. As the predictions become less certain, the colour fades to white. Colours correspond to the habitat classes shown in Figure 4.15. The holes in the habitat map are due to missing pixels in Figure 4.6 where it was not possible to calculate bathymetry features. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.

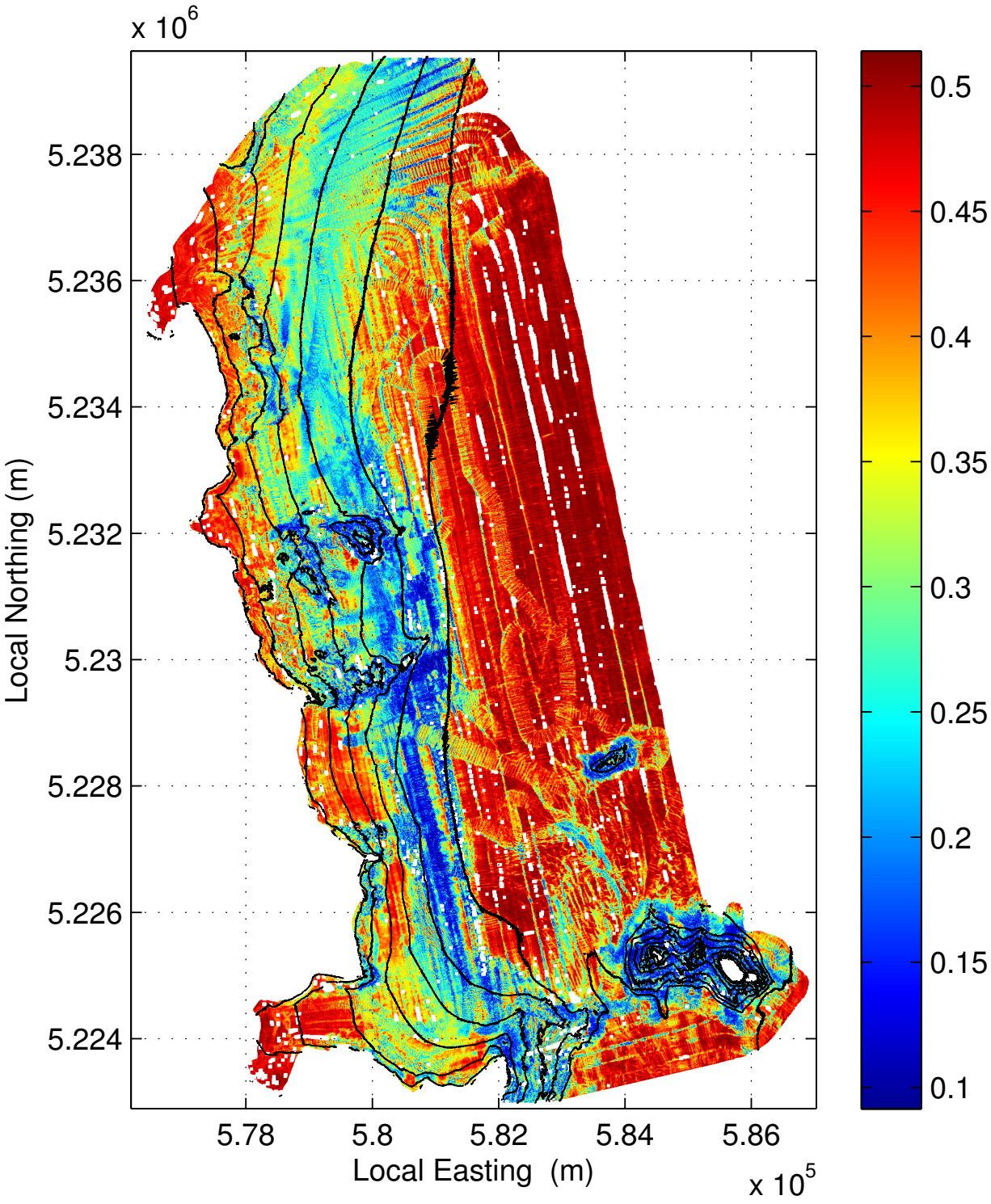


Figure 4.21 – Variance of southeastern Tasmania habitat map generated by the probabilistic targets least squares classifier (PTLSC). The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. The holes in the habitat map are due to missing pixels in Figure 4.6 where it was not possible to calculate bathymetry features. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.

An entropic state might be due to multiple conflicting observations of the same bathymetry cell or due to classes which are inseparable in the bathymetry feature space. For instance, two reef classes might be distinct in AUV imagery due to colour but appear identical in bathymetry texture. These classes are inseparable in the bathymetry feature space and will be predicted with an equal likelihood.

If the predictive probabilities are uncertain and the query locations are well supported by training data, it is likely the input vector is not descriptive enough to separate the observed classes. This can be determined by referring to the second level of self-assessment, the variance of the GP latent function. The latent function variance is low in locations where there is ample training data to support inference and high in locations where there is no training data available to support inference.

As an input location moves further away from training data, the predicted variance grows and the mean of the latent function returns to its prior. This causes the predicted probabilities to return to a highly entropic and uninformative state. This simply means that the classes cannot be separated as there is no data to support reasonable inference. On the other hand, if the predicted variance is low and the predicted probability is highly entropic, the data is supporting an ambiguous hypothesis. The model has identified a region in the input space where the training data provides conflicting information. This likely means the training data is not descriptive enough to disambiguate between classes.

In short, the predictive probability provides a measure of class separability and the predicted variance indicates the distance to training data. In Figures 4.20 and 4.21, high-variance locations of the environment are correlated with low probability predictions. As discussed, this means there are no observations to support reasonable inference in these areas.

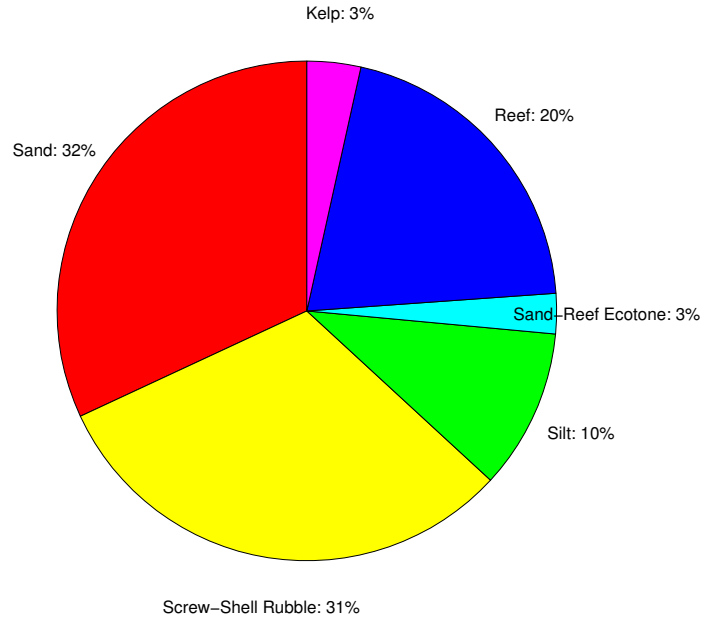
Predictions are confident in locations of the environment where the bathymetry features are close to observed features. The AUV surveys spanned depths from $30m$ to $90m$. On average the surveys collected data between $50m$ and $70m$ depths (see Table 4.4). Low-variance predictions, shown by the blue region in Figure 4.21, occur in this range of depths as there is data available to support inference.

Although weak performance is not desirable, the advantage of using a probabilistic model is in the self-assessments it provides. End users of the predictions can decide how much faith to put in the predictions given the model confidence.

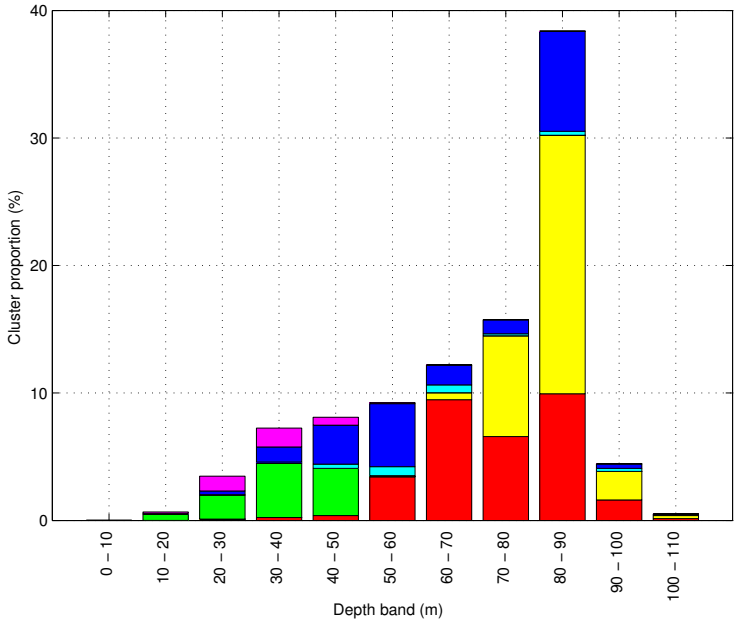
The composition of the environment is shown in Figure 4.22. Figure 4.22a shows the predicted composition of the environment. Given the high predicted variance in shallow and deep areas of the environment the predicted proportions of *Screw-Shell Rubble* and *Silt* in the environment, as shown in Figure 4.22a, are likely to be anomalously high. A finer-grained approach can be adopted by breaking up the environment into depth bands and plotting the composition of habitats found in each depth band as a proportion of the total environment. This is shown in Figure 4.22b.

Assertions about Figure 4.22b can be made for the $30m$ to $90m$ depth bands where there are confident predictions and AUV data to support inference. *Kelp* constitutes a small portion of the predicted environment and is not predicted below depths of $50m$ in significant numbers. *Reef* is predicted to exist in large numbers from $30m$ down to $70m$. Most of the environment beyond $60m$ is estimated to be substrate.

To provide an optimistic estimate of model performance, a confusion matrix was generated using all of the data for interpolation and all of the data for testing. The confusion matrix is shown in Figure 4.23. Most of the classes are estimated well. The class with the poorest performance is *Sand-Reef Ecotone*. More than a quarter of the *Sand-Reef Ecotone* observations were incorrectly classified as sand. A further 13.8% were identified as *Reef*. This confusion is unsurprising as the *Sand-Reef Ecotone* class is a mixture of the *Sand* and *Reef* classes. *Screw-Shell Rubble* was mistaken for *Sand* almost a fifth of the time. These are also two highly related classes.



(a) Predicted proportion of the environment occupied by each habitat



(b) Habitat composition at 10 metre depth intervals

Figure 4.22 – Habitat composition of southeastern Tasmania. **(a)** Predicted proportion of habitats in southeastern Tasmania. **(b)** Habitat composition at 10 metre depth intervals. Each bar in the graph represents a specific depth band. Each bar is composed of habitats found in the depth band as a proportion of the total environment.

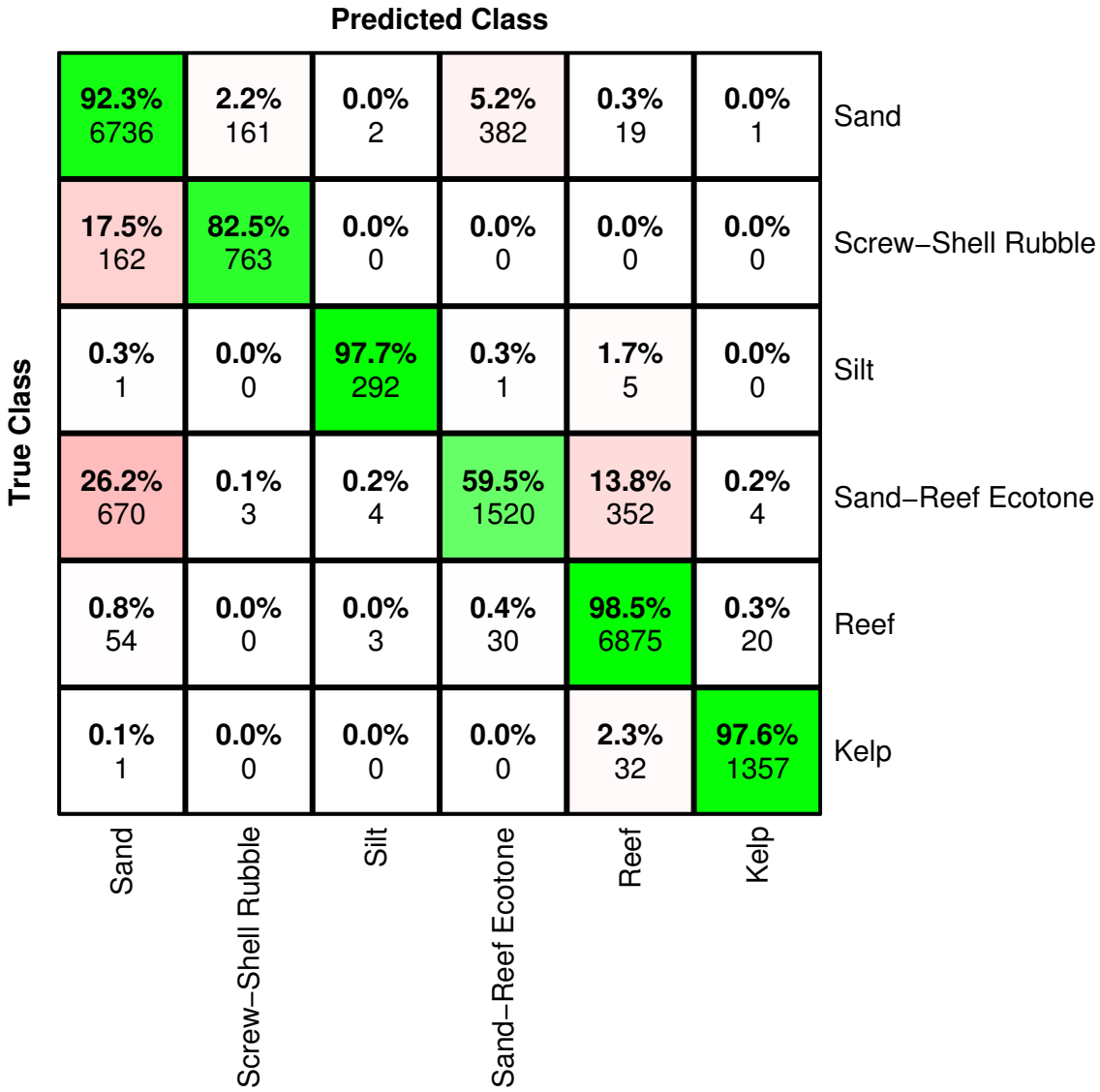


Figure 4.23 – Confusion matrix for the PTLSC. The confusion matrix was calculated by using 6000 observations for training. All of the data was used for interpolation and for testing. Data re-substitution is used to provide an optimistic indication of model performance, not necessarily an indication of model generalisation.

4.6 Summary

This chapter provided a review of benthic habitat mapping. Methods for collecting multibeam bathymetry and in-situ observations of the environment were discussed. A data set including ship-borne MBES bathymetry and AUV observations of the seafloor was introduced. The data set was collected in the southeastern Tasmania peninsula in Australia. A novel semi-autonomous habitat mapping pipeline using a 'predict first, assemble later' approach was proposed and demonstrated using the Tasmania data set.

The habitat mapping pipeline uses an unsupervised clustering algorithm to categorise the in-situ AUV observations into habitat clusters. Foregoing human supervision to classify AUV imagery reduces labour costs and allows large pools of data to be summarised quickly. Rather than relying on a human expert to manually classify large volumes of images, they are required to review a small set of habitat clusters. These proxies can be merged and discarded to produce a set of habitat classes.

To take full advantage of the probabilistic information provided by the unsupervised clustering algorithm, the PTLSC described in Chapter 3 is used to classify features extracted from the ship-borne MBES bathymetry. When compared to an approach that only uses the discrete class labels for training, the results show that the PTLSC is the more accurate and confident model. The benefits of using the PTLSC are strongest when there is ambiguity in the test data.

A habitat map for the Tasmania data was produced using the proposed pipeline. The habitat map produces an accurate estimate of the environment in locations close to observed values. In locations far from observed values the habitat predictions become weaker. The self-diagnostic properties of the GP-based classifier highlight the benefits of using a Bayesian probabilistic model. Weak predictions are associated with a high predicted variance. The prevalence of high-variance predictions in the environment suggests the habitat model would benefit from further observations of the environment.

Efficiently gathering additional observations from the environment is not a trivial problem. Extra surveys must be planned carefully to maximise the amount of information extracted from the environment. In large environments this is complicated by the number of possible survey trajectories which must be considered. A method for planning surveys in large environments is proposed in Chapter 5.

Chapter 5

Autonomous Survey Planning

The problem of efficiently gathering information from large environments applies to a wide variety of applications from planetary exploration [94] to remote sensing and environmental monitoring [90]. Autonomous platforms are well suited to exploring environments. They can collect precisely navigated, high-resolution data while operating beyond the limits of human endurance and safety. However, they are typically limited by finite battery capacities, data storage and computational resources. Additionally, the projects which support these vehicles have limited budgets and time frames to complete their objectives. These constraints make it impractical to sample the environment exhaustively. To use the limited resources effectively, trajectories which maximise the amount of information gathered from the environment must be designed.

This chapter proposes a novel method for exploring large environments. The approach is proposed in Section 5.1 and is designed to address the curse of dimensionality inherent in survey based exploration problems. Section 5.2 reviews objective functions used in active learning and experimental design. Section 5.4 demonstrates and validates the proposed method on real marine data when no prior observations are available. Finally, Section 5.5 provides concluding remarks.

5.1 Exploration via Learning

Gathering and analysing data is often an expensive and labour-intensive task. In most applications it is not possible to exhaustively sample the environment and analyse all the collected data. Instead small samples of data are collected and analysed. These small data sets can then be generalised into a model which relates the observed data to the outcomes of the analysis. For example, in classification problems, the task is to map input vectors to a discrete category from a finite set.

Generalising a small number of in-situ observations into a model as a function of the broadly observed environmental data allows inference to be performed in arbitrary locations. Clearly the efficacy of the model is dependent on the training data that has been analysed. To most effectively utilise the resources at hand, robot operators must collect data which best suits their modelling objectives.

One framework which has gained considerable attention for its ability to gather information from the environment is partially observable Markov decision processes (POMDPs) [40, 95]. Under the POMDP framework, robot actions and observations are considered uncertain. Since the true state is unobservable, the robot must maintain a distribution over the true state and plan in this belief space. Operating in a belief space provides a theoretically rigorous framework for balancing goal acquisition and information gathering.

In POMDPs the model will only engage in information gathering behaviour to the extent that it helps the agent reach its goal state. Information gathering is only a means to an end. The extent to which exploration and exploitation behaviours are exhibited depends on how rewards are specified. Supplying a POMDP with a reward function which seeks information is a possibility. However, the complexity of generating a control policy in the POMDP belief space is an exponentially hard problem and considered PSPACE-complete [69]. Although approximate methods have been proposed to scale POMDPs up to larger problems [70, 73, 82], none have been applied successfully to exploration problems where the environment is unknown. These problems are currently considered intractable.

Unlike many conventional planning problems [49, 71, 92], exploration is an open-ended task. Well-defined objectives such as a known initial and final configuration are complex decisions which may be an integral part of the problem. The only strongly specified objective is to maximise the amount of information gathered from the environment. In this context the primary concern is collecting data which optimises some statistical objective.

The problem of collecting data which optimises some statistical objective is referred to as *experimental design* [5, 16] and was originally formalised by the statistics community. More recently, the problem of collecting data to optimise some criteria has been studied by the machine learning communities and in the field of control theory. In the machine learning community active learning or query learning [84] is a framework where the agent is able to request data from the environment.

The main idea behind active learning is that algorithms which seek informative data will outperform naive learning algorithms and will require less training. However, the information gained from including a single observation, or batches of observations, is commonly assessed independently of the resources required to make an observation. The general assumption is that making observations and submitting the data for analysis is relatively inexpensive. This assumption is not true in the context of robotic exploration as it ignores the constraints of the robot.

Data drawn from the environment based solely on some statistical objective, with no consideration of how that data is distributed in physical space, will lead to inefficient or impossible survey trajectories. To apply experimental design effectively to robotic exploration, a constraint must be included to ensure each survey can be completed by the robot. Robot operators often naturally impose constraints on vehicle deployments. For instance, in the majority of autonomous underwater vehicle (AUV) deployments, AUV operators and marine scientists use prespecified survey trajectories which are composed offline. The dynamic limitations of the AUV and the scientific objectives of the mission are encoded into these trajectories. The surveys are often lawn mower patterns or are simple linear designs which follow gradients or depth contours. An example of a human-designed survey template is shown in Figure 5.1.

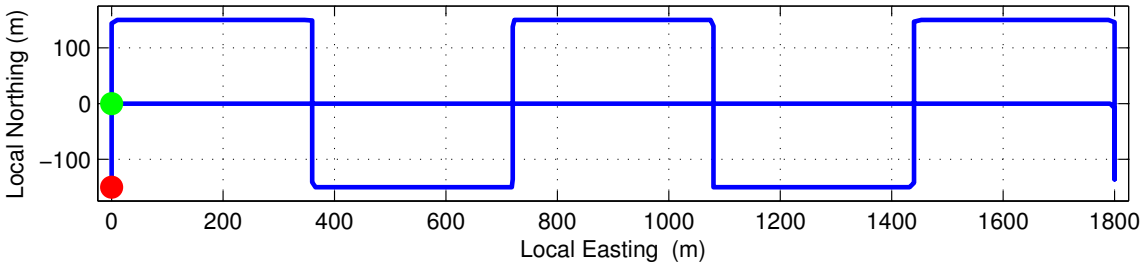


Figure 5.1 – Example of a survey template. The green indicates the starting location of the survey. The red dot indicates the ending location of the survey. The blue line represents the vehicle trajectory.

By limiting the vehicle to a prespecified survey template, only batches of observations which can be observed by placing the survey template somewhere in the environment need to be considered. The vast unconstrained exploration problem is simplified to the more tractable task of selecting the optimal location to place the survey. However, assessing every possible survey placement exhaustively is still a large problem.

To further reduce the size of the problem, the survey utility is evaluated at a small set of locations and a functional representation is learnt using a Gaussian process (GP). The functional representation allows the survey utility to be tractably approximated over the entire domain of interest by querying the GP utility model. This process is not complicated by obstacles. In natural marine environments it is possible for vehicles to maintain a survey template in the horizontal plane by traversing over obstacles.

Before the planning method can be applied, a utility function which captures the statistical objectives of the survey must be specified. Many utility functions can be used to assign a value to the quality of a survey. Utility functions found in the literature are discussed in the following section.

5.2 Utility Functions

A utility function allows potential survey placements to be compared and ranked in order of preference. This utility function should reflect the objectives of the survey and can be used to guide exploration. In this chapter, the goal is to create an accurate habitat map. Several utility functions found in the literature are summarised in this section.

5.2.1 Random Sampling

A naive strategy for fulfilling modelling objectives is to randomly select data for analysis. This strategy is also known as passive learning and is often used as a benchmark for active learning strategies. Active learning is considered worthwhile if the active learning strategy provides superior performance to random sampling for all or most of the active learning cycles.

5.2.2 Uncertainty Sampling

A probabilistic heuristic for determining which data points are informative is choosing to analyse the data which produces the most uncertain predictions. This framework for selecting data is known as uncertainty sampling [51]. The intuition behind uncertainty sampling is that a model can learn the most by analysing data where it knows the least.

Several measures of uncertainty are found in the active learning literature. These methods are popular and simple to implement in models which produce probabilistic output. Three measures, surveyed in [84], are summarised in this section.

Least confident sampling is a strategy which uses only the most probable class labels to calculate a measure of uncertainty:

$$\mathbf{X}^* = \arg\max_{\mathbf{X}_*} \left[1 - p(\hat{\mathbf{y}} \mid \mathbf{X}_*)\right] \tag{5.1}$$

where $\hat{\mathbf{y}}$ are the most probable class labels predicted by the model and are given by:

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{X}_*) \,.$$

Data points with confident predictions indicate the model is able to differentiate the most likely label against all other labels. Conversely data points with predictions which are not confident, suggest the model is not able to strongly distinguish the most likely class from all others. Intuitively, gathering data in these regions should help the model separate the confused classes.

Minimum margin sampling uses both the most and second most probable class labels as a proxy for uncertainty:

$$\mathbf{X}^* = \arg\min_{\mathbf{X}_*} \left[p(\hat{\mathbf{y}}_1 \mid \mathbf{X}_*) - p(\hat{\mathbf{y}}_2 \mid \mathbf{X}_*) \right] \tag{5.2}$$

where $\hat{\mathbf{y}}_1$ and $\hat{\mathbf{y}}_2$ are the most and second most probable class labels predicted by the model respectively. Large margins imply the model can distinguish between the two most likely classes. Small margins occur when the model has trouble discriminating between the two most likely classes. By querying small margins, the active learning agent is introducing data which should help to separate the two most likely classes.

Maximum entropy sampling is an information theoretic approach to measuring uncertainty and is given by the Shannon entropy of the predictions:

$$\mathbf{X}^* = \arg\max_{\mathbf{X}_*} \left[-\sum_{i=1}^{C} p(\mathbf{y}_i = 1 \mid \mathbf{X}_*) \log p(\mathbf{y}_i = 1 \mid \mathbf{X}_*) \right] . \tag{5.3}$$

Shannon entropy differs from the previous two methods in that it uses the information contained in all C classes to quantify the uncertainty of a data point. Minimum (zero) entropy occurs in regions where the model can assign data to one class with full confidence. As the probability of class membership converges to an uninformed state, the entropy grows. Maximum entropy occurs when the individual class probabilities

are equal. In these regions, the input data is equally well explained by all classes. Collecting data in these regions should help disambiguate between classes.

5.2.3 Bayesian Experimental Design

Bayesian experimental design [16, 53] is particularly suited to information gathering tasks. Under the Bayesian paradigm, prior knowledge and subsequent observations can be fused incrementally into a model. This elegant property allows Bayesian methods to propagate estimates and their uncertainty as data is collected.

The outcome of an experiment can only be observed once the experiment has been performed. Given this uncertainty, a planning agent must select an experiment or action based on its *expected* utility. If the outcomes of the experiment were known in advance, calculating the utility of the experiment would be trivial. By considering all possible future outcomes of an experiment, conditioned on the data that has already observed, $\mathcal{D}$, a planning agent can calculate the expected utility of performing the experiment. For a particular experiment, $\mathcal{E}$, and a general utility function, $U(\ldots)$, this expectation is given by

$$\mathbb{E}[U_{\mathcal{E}}] = \int p(\mathbf{y}_+ \mid \mathbf{X}_{\mathcal{E}}, \mathcal{D})\, U(\ldots)\, d\mathbf{y}_+, \tag{5.4}$$

where $\mathbf{y}_+$ is an unobserved future observation. This expectation captures the balance between risk and reward. A risky experiment which has high-utility outcomes that are unlikely to occur may be more rewarding than a conservative experiment where low-utility outcomes are very likely to occur.

The optimal experiment is simply the experiment which maximises the expected utility from the set of proposed experiments ξ,

$$\mathcal{E}^* = \arg\max_{\mathcal{E}\in\xi} \mathbb{E}[U_{\mathcal{E}}]\,. \tag{5.5}$$

In any experimental design application the utility function should reflect the objectives of the experiment. The Kullback-Leibler divergence (KLD) between the pos-

terior belief after conducting an experiment and the prior belief before conducting the experiment was proposed as a measure of information gain early in the statistics literature [53]. The KLD is used to measure the difference between the two distributions and can be viewed as the amount of extra information required to explain the updated posterior belief in terms of the prior belief. For the prior belief $p(\mathbf{X})$ over some random variable $\mathbf{X}$ and the updated posterior belief $p(\mathbf{X} \mid \mathbf{y}_+)$ given a future observation $\mathbf{y}_+$, the KLD is given by (5.6).

$$\begin{aligned} U(\ldots) &= KL\left(\, p(\mathbf{X} \mid \mathbf{y}_+) \parallel p(\mathbf{X}) \,\right) \\ &= \int p(\mathbf{X} \mid \mathbf{y}_+) \log \frac{p(\mathbf{X} \mid \mathbf{y}_+)}{p(\mathbf{X})}\, d\mathbf{X} \end{aligned} \tag{5.6}$$

Since the outcome of the experiment is unknown, only the *expected* utility can be quantified. By simplifying the notation in (5.4) and combining with (5.6) the expected utility can be described in information theoretic terms.

$$\begin{aligned} \mathbb{E}[U_{\mathcal{E}}] &= \int p(\mathbf{y}_+ \mid \mathbf{X}_{\mathcal{E}}, \mathcal{D})\, KL\left(\, p(\mathbf{X} \mid \mathbf{y}_+) \parallel p(\mathbf{X}) \,\right)\, d\mathbf{y}_+ \\ &= \mathbb{E}_{\mathbf{y}_+}[KL\left(\, p(\mathbf{X} \mid \mathbf{y}_+) \parallel p(\mathbf{X}) \,\right)] \\ &= I\left[\mathbf{X}; \mathbf{y}_+\right] \end{aligned} \tag{5.7}$$

The result given by (5.7) shows that by combining the objective function with a utility function based on information gain, the optimal experiment is the one which maximises the mutual information between $p(\mathbf{X})$ and $p(\mathbf{X} \mid \mathbf{y}_+)$. Further insight into this objective function can be provided by expanding (5.7).

$$\begin{aligned} \mathbb{E}[U_{\mathcal{E}}] &= \int p(\mathbf{y}_+ \mid \mathbf{X}_{\mathcal{E}}, \mathcal{D})\, KL\left(\, p(\mathbf{X} \mid \mathbf{y}_+) \parallel p(\mathbf{X}) \,\right)\, d\mathbf{y}_+ \\ &= \int p(\mathbf{y}_+ \mid \mathbf{X}_{\mathcal{E}}, \mathcal{D}) \left[\int p(\mathbf{X} \mid \mathbf{y}_+) \log \frac{p(\mathbf{X} \mid \mathbf{y}_+)}{p(\mathbf{X})}\, d\mathbf{X} \right] d\mathbf{y}_+ \\ &= \int p(\mathbf{y}_+ \mid \mathbf{X}_{\mathcal{E}}, \mathcal{D}) \left[\int p(\mathbf{X} \mid \mathbf{y}_+) \left[\log p(\mathbf{X} \mid \mathbf{y}_+) - \log p(\mathbf{X})\right]\, d\mathbf{X} \right] d\mathbf{y}_+ \\ &= \int p(\mathbf{y}_+ \mid \mathbf{X}_{\mathcal{E}}, \mathcal{D}) \left[- \int p(\mathbf{X} \mid \mathbf{y}_+) \log p(\mathbf{X})\, d\mathbf{X} - \mathrm{H}[\mathbf{X} \mid \mathbf{y}_+] \right] d\mathbf{y}_+ \end{aligned}$$

$$
\begin{aligned}
&= -\int\int p(\mathbf{X} \mid \mathbf{y}_+)\, p(\mathbf{y}_+) \log p(\mathbf{X})\, d\mathbf{X} d\mathbf{y}_+ - \mathbb{E}_{\mathbf{y}_+}[\mathrm{H}[\mathbf{X} \mid \mathbf{y}_+]] \\
&= -\int p(\mathbf{X}) \log p(\mathbf{X})\, d\mathbf{X} - \mathbb{E}_{\mathbf{y}_+}[\mathrm{H}[\mathbf{X} \mid \mathbf{y}_+]] \\
&= \mathrm{H}[\mathbf{X}] - \mathbb{E}_{\mathbf{y}_+}[\mathrm{H}[\mathbf{X} \mid \mathbf{y}_+]] \qquad (5.8)
\end{aligned}
$$

The intuition behind (5.8) is that the mutual information represents the expected reduction in the uncertainty about $\mathbf{X}$ as a consequence of making an observation $\mathbf{y}_+$. The expected reduction in uncertainty is measured using entropy, $\mathrm{H}[\cdot]$. Experiments which result in a large expected change in entropy are likely to introduce more information than experiments with a small expected change.

Choosing which distribution to use for $p(\mathbf{X} \mid \mathbf{y}_+)$ and $p(\mathbf{X})$ depends on the objectives of the experiment. In physical applications, the task is often to learn the parameters of a model [35, 54]. Given the task of inferring model parameters from noisy and incomplete observations, an appropriate utility function is one which quantifies the expected gain in information about the model parameters, after performing an experiment,

$$U(\ldots) = KL\left(\, p(\theta \mid \mathbf{X}_{\mathcal{E}}, \mathbf{y}_+, \mathcal{D}) \;\|\, p(\theta \mid \mathcal{D})\,\right).$$

Under certain conditions this utility function can be simplified down to a form of maximum entropy sampling [54, 83].

In other applications, the quality of predictions can be more important than estimating the parameters of a model. An appropriate utility function in these applications is one which quantifies the expected gain in information about the predictive distribution after performing an experiment [16],

$$U(\ldots) = KL\left(\, p(\mathbf{y}_* \mid \mathbf{X}_{\mathcal{E}}, \mathbf{y}_+, \mathbf{X}_*, \mathcal{D}) \;\|\, p(\mathbf{y}_* \mid \mathbf{X}_*, \mathcal{D})\,\right) \qquad (5.9)$$

where $\mathbf{y}_*$ is the predictive distribution at the test locations $\mathbf{X}_*$. This objective function is used in [45] in a regression problem with a Gaussian process. The task is to choose a near optimal set of observations locations $\mathcal{E}$, out of a finite subset ξ of possible

locations such, that the regression model contains the least generalisation error. Using Equation (5.8), the objective function is given as,

$$I\left[\mathbf{X}_{\xi\setminus\mathcal{E}}, \mathbf{X}_{\mathcal{E}}\right] = \mathrm{H}\left[\mathbf{X}_{\xi\setminus\mathcal{E}}\right] - \mathrm{H}\left[\mathbf{X}_{\xi\setminus\mathcal{E}} \mid \mathbf{X}_{\mathcal{E}}\right]$$

where the mutual information measures the difference in entropy between the measured locations, $\mathbf{X}_{\mathcal{E}}$, and the remaining space, $\mathbf{X}_{\xi\setminus\mathcal{E}}$.

Habitat mapping is a classification problem, not a regression problem. A similar utility function to Equation (5.9) is given in [78] and is applied to marine habitat mapping with a Gaussian process,

$$\mathbb{E}[U_{\mathcal{E}}] = \mathrm{H}[\mathbf{y}_* \mid \mathbf{X}_*, \mathcal{D}] - \mathbb{E}_{\mathbf{y}_+}[\mathrm{H}[\mathbf{y}_* \mid \mathbf{X}_{\mathcal{E}}, \mathbf{y}_+, \mathbf{X}_*, \mathcal{D}]]\,. \tag{5.10}$$

Since the first term in (5.10) does not depend on the experiment $\mathbf{X}_{\mathcal{E}}$, it is constant for all experiments and can be ignored. Maximising the mutual information can then be performed by minimising the entropy of the predictive distribution

$$\begin{aligned}\mathbb{E}[U_{\mathcal{E}}] &= -\mathbb{E}_{\mathbf{y}_+}[\mathrm{H}[\mathbf{y}_* \mid \mathbf{X}_{\mathcal{E}}, \mathbf{y}_+, \mathbf{X}_*, \mathcal{D}]] \\ &= -\int p(\mathbf{y}_+ \mid \mathbf{X}_{\mathcal{E}}, \mathcal{D})\, \mathrm{H}[\mathbf{y}_* \mid \mathbf{X}_{\mathcal{E}}, \mathbf{y}_+, \mathbf{X}_*, \mathcal{D}]\; d\mathbf{y}_+ \end{aligned} \tag{5.11}$$

where

$$\begin{aligned}\mathrm{H}[\mathbf{y}_* \mid \mathbf{X}_{\mathcal{E}}, \mathbf{y}_+, \mathbf{X}_*, \mathcal{D}] = -\sum_{i=1}^{N}\sum_{j=1}^{C} & p(\mathbf{y}_{*ij} \mid \mathbf{X}_{\mathcal{E}}, \mathbf{y}_+, \mathbf{X}_{*i}, \mathcal{D}) \\ & \log p(\mathbf{y}_{*ij} \mid \mathbf{X}_{\mathcal{E}}, \mathbf{y}_+, \mathbf{X}_{*i}, \mathcal{D})\,.\end{aligned} \tag{5.12}$$

The predictive entropy (5.12) is often only calculated at unlabelled locations of the data set but it can be calculated at any relevant portion of the input space. Variants of this objective function have been used widely in the active learning community [28, 29, 66, 85].

5.2.4 Minimising Predictive Variance

In general, performing Bayesian experimental design requires optimising an objective function with analytically intractable integrals. For certain combinations of models and objective functions, closed-form solutions can be derived. In the majority of cases these combinations do not exist and Bayesian experimental design requires expensive numerical sampling. A non-Bayesian approach to experimental design has been to minimise the predictive variance.

Optimal experimental design in linear models has largely been concerned with optimising statistical properties of a model's Fisher information matrix [2]. The Fisher information matrix is a function of a model's explanatory variables and can be used to minimise the predictive variance of the model. The predictive variance and Fisher information of the model share an inverse relationship. Maximising the Fisher information corresponds to minimising the predictive variance.

Several target functions are commonly used to maximise the Fisher information. These target functions are known as alphabetic design criteria. Popular design criteria are A-optimality and D-optimality. A-optimal designs minimise the trace of the inverse Fisher information matrix. This corresponds to minimising the average variance of the predictions. D-optimal designs minimise the determinant of the inverse Fisher information matrix. This corresponds to minimising the expected posterior entropy over the model parameters.

Many of the alphabetic optimality criteria have Bayesian analogues [16]. Bayesian alphabetic optimality has the same objective - to reduce the predictive variance of a model. The key difference between Bayesian alphabetic optimality and classic alphabetic optimality is that the classical methods assume the current model is correct. Bayesian methods are robust to biases caused by an incorrect model as they account for uncertainty about the current model [23]. Closed-form solutions for optimising these quantities have been derived for various models including linear models [16] and neural networks [56]. The Bayesian alphabetic optimality criteria can be derived for Gaussian processes [45].

5.3 Planning without Observations

All of the utility functions described in Section 5.2 rely on evaluating properties of a model that already exists. When no observations of the environment exist, it is not possible to build and exploit a model. Although no in-situ samples of the environment exist, the structure of bathymetry features in the environment can be exploited to design surveys which cover as much of the bathymetry feature space as possible.

Recall that habitat maps are created by correlating features extracted from the digital elevation model (DEM) with classified in-situ observations. Since the independent variable is the bathymetry feature space, exploration should be optimised in this space. The bathymetry feature space can be compactly represented by modelling the density of the features using a Gaussian mixture model (GMM).

A candidate survey which collects a representative set of observations from the environment will produce a density of observations similar to that of the entire environment. The density of features observed by a survey can also be compactly modelled using a GMM.

Similarity between the density of features in the environment and the density of features to be visited by a candidate survey can be measured using the KLD. The KLD is given by (5.13),

$$KL\left(\, p \,\|\, q \,\right) = \int p(\mathbf{x}) \log \left(\frac{p(\mathbf{x})}{q(\mathbf{x})} \right) d\mathbf{x}, \tag{5.13}$$

where $p(\mathbf{x})$ is the density of the bathymetry and $q(\mathbf{x})$ is the density of a survey. A survey with a low KLD will cover a more representative set of bathymetry features in the environment than a survey with a higher KLD. The goal for the remainder of this chapter is to provide a framework for selecting a survey placement so as to minimise the KLD between the density of features in the environment and the density of features to be visited.

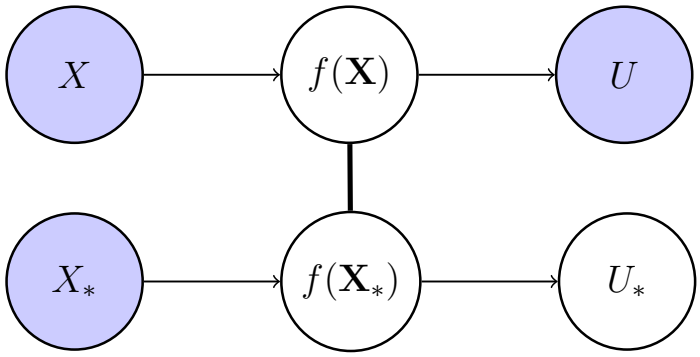


Figure 5.2 – Graphical model of a Gaussian process modelling the hidden survey utility function. Observed and unobserved variables are shown as shaded and unshaded nodes, respectively. The bold edge between the hidden utility function variables, $f(\cdot)$, indicates that these latent variables reside in a fully connected Gaussian field. A description of the node variables is provided in the main text.

Evaluating the utility of every possible survey placement is an exhaustive and intractable task in large-scale environments. To avoid relying on an inefficient brute-force approach, the survey utility can be estimated using a small number of training surveys. In this chapter a GP [75] is used to model the underlying utility function for a specific survey template.

A graphical model of the GP utility model is shown in Figure 5.2. All possible survey placements are represented by the node $\mathbf{X}_*$. The survey training locations, or training inputs, are represented by the node $\mathbf{X}$ and are a small subset of locations drawn from $\mathbf{X}_*$. The survey utility at the training locations, or training targets, are represented by the node $\mathbf{U}$ and are calculated explicitly. The *predicted* survey utility for all possible survey placements is represented by the node $\mathbf{U}_*$.

Designing a representation for the input space, $\mathbf{X}$ and $\mathbf{X}_*$, is an important consideration. A complex representation of survey placements could be expressed in the bathymetry feature space. Although the batch of bathymetry features traversed by a candidate survey describes the survey exactly, learning a model in this high-dimensional space is challenging. A low-dimensional summary of the bathymetry traversed by a survey placement would be required. However, it must be possible to calculate millions of these 'survey features' in a tractable amount of time.

In this chapter a survey placement is represented by the Cartesian co-ordinates of its origin which is trivial to calculate for the millions of possible survey placements. Using this representation, the GP is essentially performing a spatial interpolation of the training data. It is worth noting that by modelling the utility function with a GP, it would also be possible to learn how the survey utility varies with parameters which control the shape of a survey template. Such parameters might include the orientation, length, width and track spacing of a survey template.

To ensure the predicted KLD metric will be positive, regression is performed in a log-KLD space. This is facilitated by transforming the training data into a log space and subtracting the mean to ensure the log-KLD training data has a zero mean. To transform the predicted log-KLD values back to a linear-KLD space, the reverse process is used.

The GP utility model can be trained by minimising the negative log marginal likelihood of the GP with respect to the hyperparameters of the covariance function. In this chapter a squared exponential covariance function, Equation (2.9), is optimised using conjugate gradient descent.

Once the GP has been trained, the survey utility can be estimated for all possible survey placements by querying the utility model. The prohibitive cost of performing experimental design at all possible survey locations has been replaced with the tractable requirement of calculating data to support training and inference in the utility model. The optimal survey placement for the first dive is given by selecting the location with the lowest predicted KLD.

5.3.1 Relation to Previous Work

In [20], separate GPs are used to model the value function and state-action space in dynamic programming. Modelling the control problem in this way allows dynamic programming to be performed in a continuous state and action space and provides a mechanism for learning the system dynamics in reinforcement learning problems. A similar approach was published in [81]. The exploration method proposed in this

chapter is inspired by the concept of using a GP to model a utility function. The approaches differ in what the GP is modelling. Rather than applying the concept to dynamic programming, this chapter applies the concept to experimental design. In this context the motivation is not to operate in a continuous space but to replace the costly process of experimental design with relatively cheap utility model queries. Departing from the dynamic programming framework also frees the approach proposed in this chapter from specified initial and goal configurations.

A similar application, planning informative surveys in marine environments, is detailed in [78]. The planning framework is designed to rank a list of candidate surveys where the design and location of all surveys in the list is the responsibility of platform operators. The primary advantage of the method proposed in this chapter over [78] is the ability to calculate the survey utility for a survey template in arbitrary locations. Although operators must still design a survey template, they are freed from the responsibility of having to specify a survey location. Whilst the method proposed in [78] can naturally rank heterogeneous survey templates, it is possible to attain a similar result with the proposed method by running a planning cycle for each individual survey template.

5.4 Results

In this section KLD as a planning heuristic is evaluated. The method is evaluated using the data set from Section 4.4. In this chapter, the bathymetry feature space is described using depth, log rugosity and log slope with rugosity and slope calculated at scales of 2 ($8m$), 4 ($14.4m$), 8 ($27.2m$) and 16 ($52.8m$) (see Figure 4.7, Figure 4.8). The bathymetric feature vector is standardised by subtracting the mean from each dimension and dividing by the standard deviation. This data is used to both demonstrate and validate the proposed exploration method. The task is to explore the environment efficiently using only this information.

Firstly, a survey template is designed. In this demonstration the survey template shown in Figure 5.1 is used. To train the GP utility model, 4933 survey placements are distributed evenly throughout the environment. This sampling approach is used for its simplicity. More sophisticated approaches might sample uniformly in the bathymetry feature space.

The density of the environment and each potential survey placement is modelled by fitting a GMM to the data. The number of Gaussians in the mixture, the mixing coefficient and the mean and variance of each Gaussian are estimated using an unsupervised variational Dirichlet process (VDP) [46]. Since no analytic solution for the KLD between two GMMs exists, Monte Carlo sampling is used [32] to evaluate Equation (5.13). The KLD for each of the training placements are shown in Figure 5.3.

The predicted KLD for all possible placements of the survey template is shown in Figure 5.4. Each of the 29,361,249 pixels represents the predicted utility of a survey placement which allows at least half of the survey template to traverse valid bathymetry features.

The predicted survey utility closely matches the training data as shown in Figure 5.4. Regions with a high KLD are coloured yellow and red. At these locations the survey template is traversing a set of bathymetry features which lacks diversity. For instance, surveys which occur in the most Easterly region of the DEM traverse flat bathymetry which is likely to contain only sand. Regions shown in blue are survey placements which result in a low KLD. These placements typically occur in shallow regions, near the coast line, where they are likely to traverse relatively more complex and diverse regions of the bathymetry feature space. The predominance of low-variance predictions, shown in Figure 5.5, indicates that predictions are confident and that the training set is sufficient to describe the space of possible survey placements.

The optimal survey placement, as predicted by the survey utility model, is shown in Figure 5.4 and scores a survey utility of 6.4341 nats. The utility model has selected a placement where the survey template traverses a broad range of depths. The terrain traversed by the survey template also varies from rugged and complex to flat and lacking diversity.

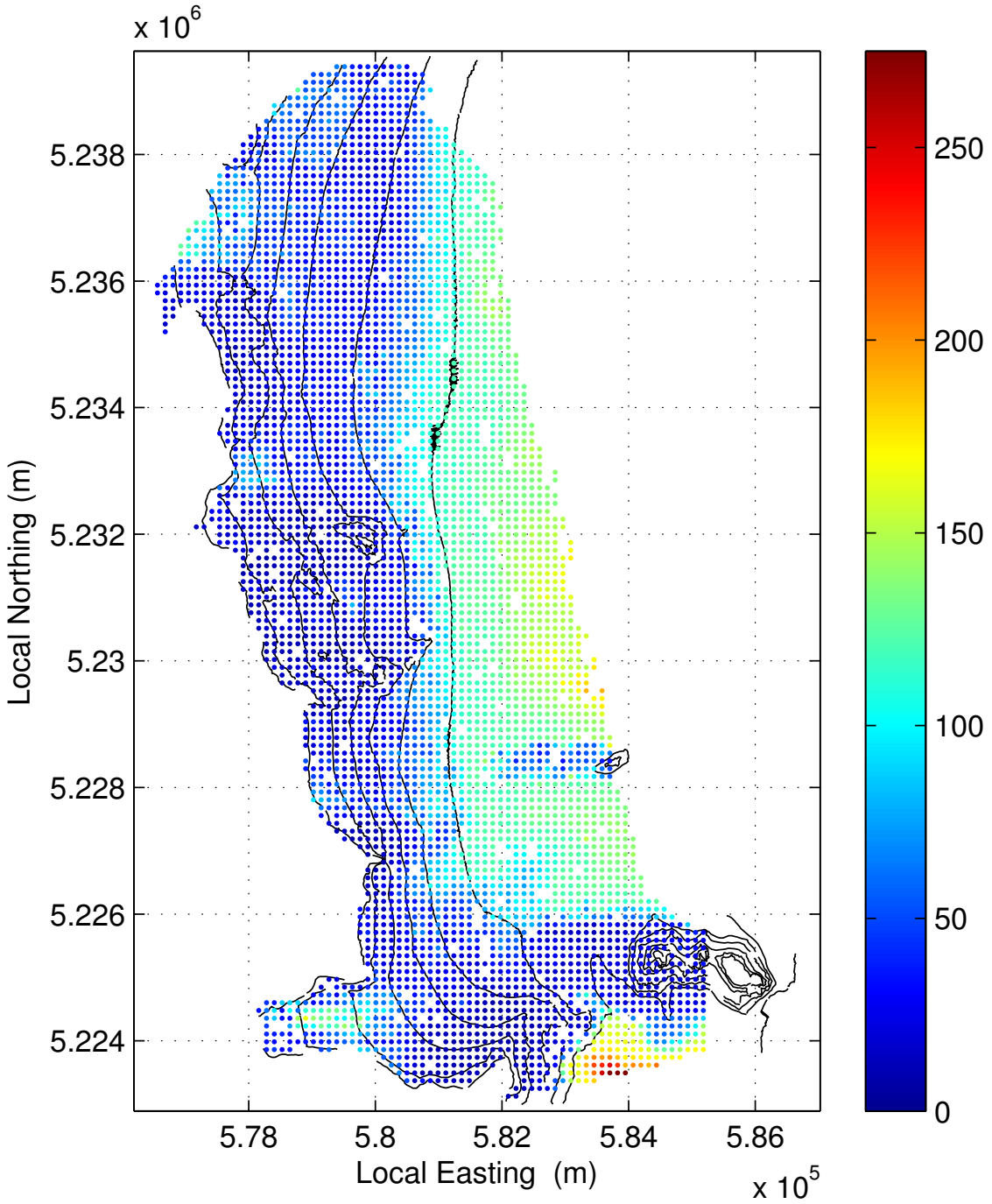


Figure 5.3 – Training targets for the Gaussian process (GP) utility model. The Kullback-Leibler divergence (KLD) for the survey template was calculated at the training locations. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.

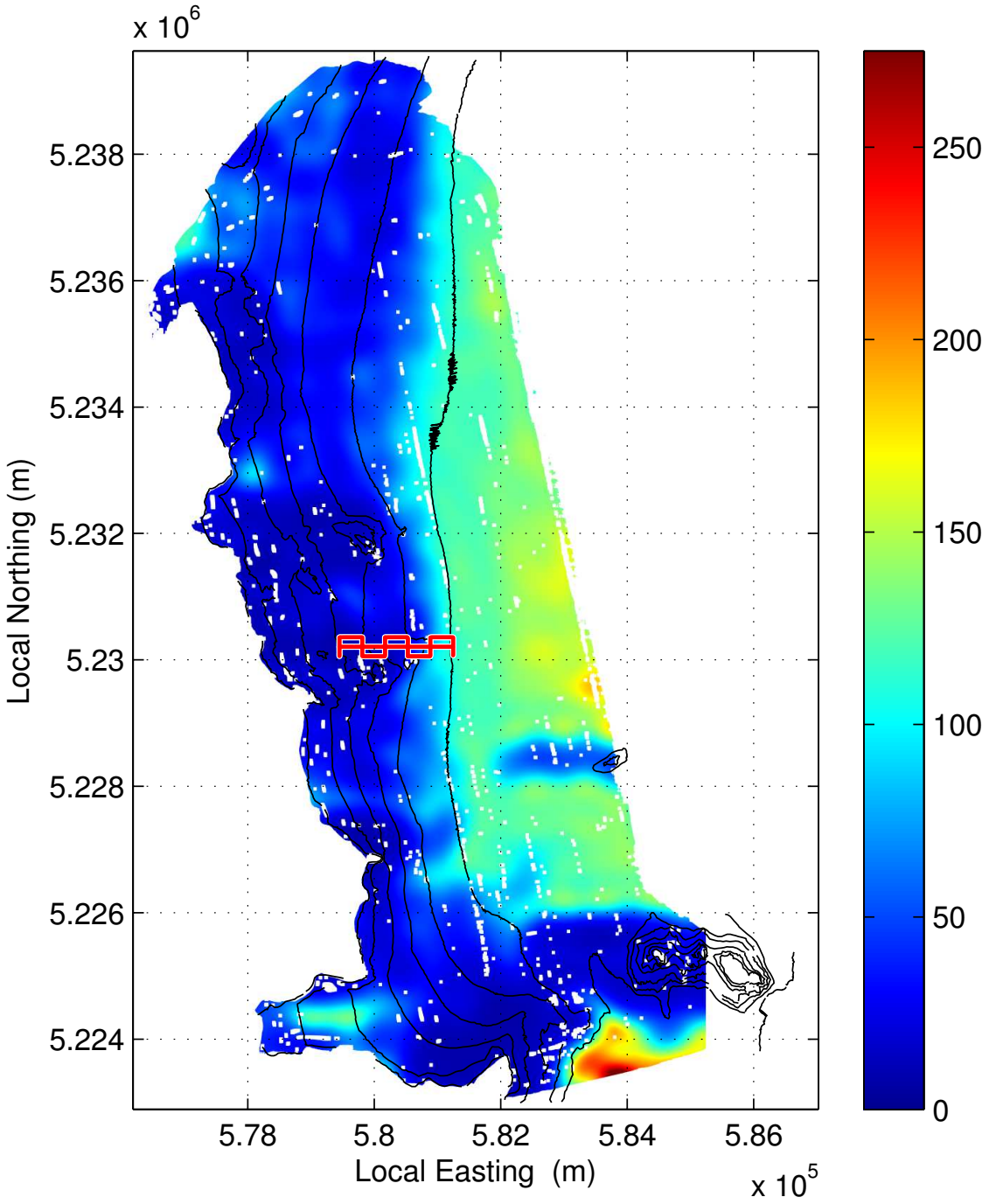


Figure 5.4 – Predicted survey utility over the entire search space. The predicted survey utility approximates the Kullback-Leibler divergence (KLD) given the survey template. The optimal survey placement as predicted by the model is shown in red. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.

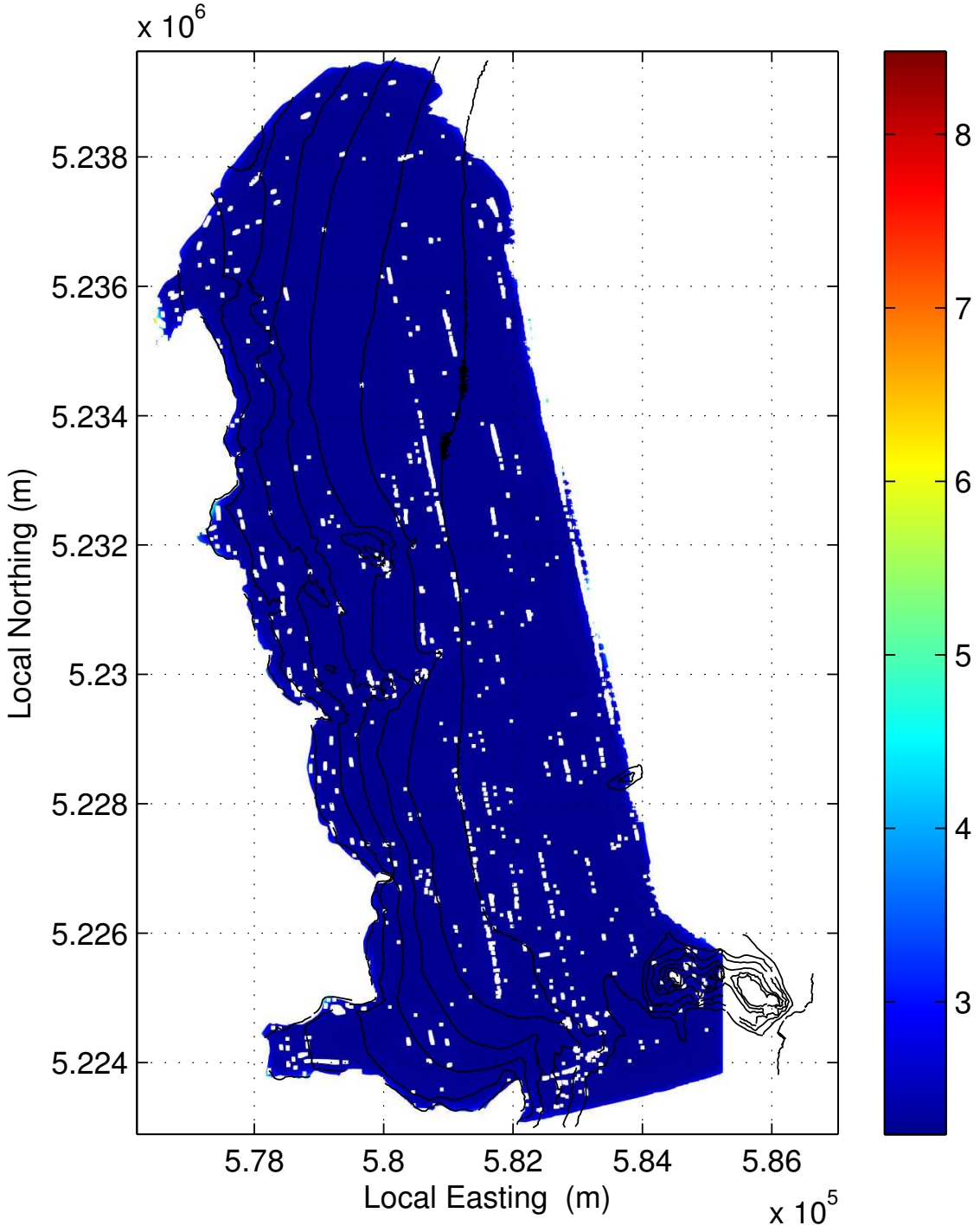


Figure 5.5 – Variance of the survey utility predictions. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.

The planning process took a total of 7.94 hours. It took 2.91 hours to calculate the survey utility for all 4933 training placements. The remaining time was spent calculating data to support the survey utility model, training the model and performing inference. A brute-force search, calculating the KLD explicitly for all 29,361,249 possible survey placements, would take 1.98 years.

To compare the performance of GP regression to simpler methods, 1000 unique random survey locations were drawn from the environment for the survey template shown in Figure 5.1. The predicted survey utility at these locations was evaluated using GP interpolation, linear interpolation and nearest neighbour search. The training data shown in Figure 5.3 was used for interpolation. The mean squared error (MSE), calculated between the true and estimated survey utility, for each method is shown in Table 5.1.

	GP	Linear Interpolation	Nearest Neighbour
MSE	77.786	63.970	96.636

Table 5.1 – Mean squared error (MSE) of interpolation methods.

The comparison shows that interpolation provides better performance than simply choosing the closest training point via a nearest neighbour search. Although GP interpolation is outperformed by linear spatial interpolation, it is preferred for the purposes of this chapter as it provides a measure of confidence and the ability to perform interpolation in more complex, nonlinear dimensions.

In 2008 several surveys were performed in the Tasman peninsula. These surveys are shown in Figure 4.11. To validate the proposed exploration method, the predicted survey utility at the location where each survey was actually performed is calculated. The intention is not to calculate where each survey *should* have been placed but rather to quantify how well the utility model can approximate the actual utility of the surveys which were conducted.

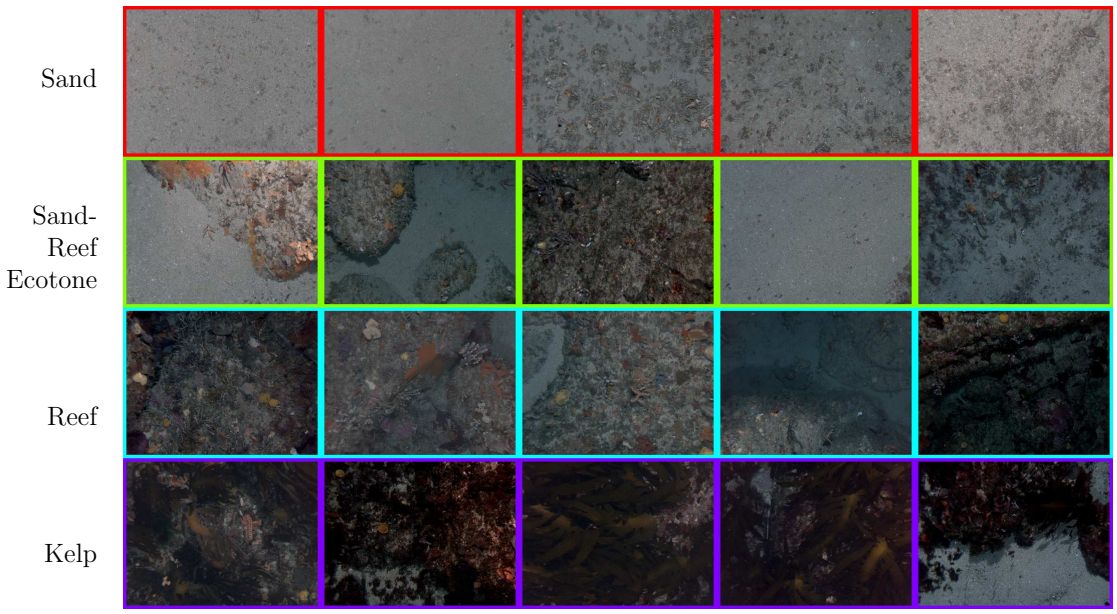


Figure 5.6 – Habitat clusters created by a VDP and verified by a human. The mosaics for clusters one through four are represented by the rows from top to bottom and are created by randomly sampling the habitat clusters.

The suitability of KLD as an exploration heuristic is evaluated by examining the accuracy of the habitat maps produced by each survey. Habitat classes are determined by categorising the in-situ imagery collected by the AUV into discrete categories. To reduce the labour requirements of classifying the data, the AUV images are clustered using visual features and a VDP model as detailed in [91].

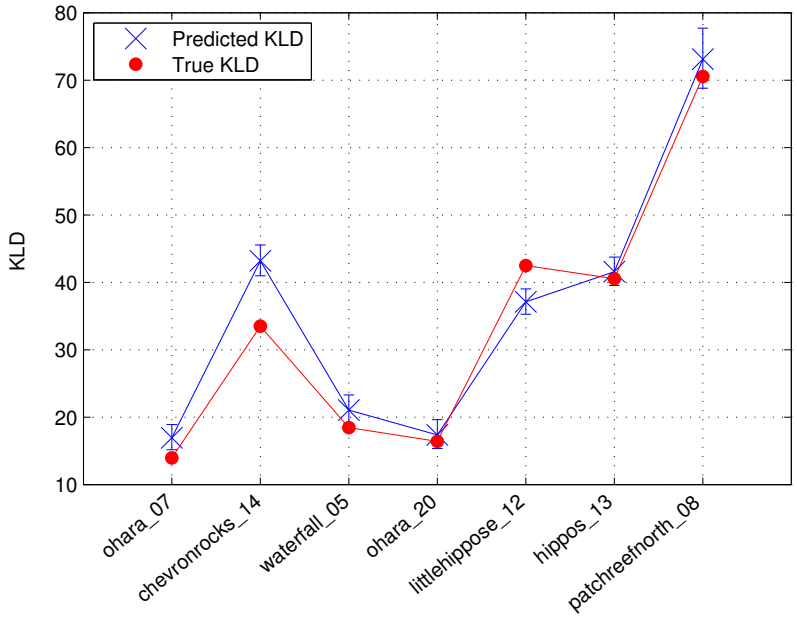
The survey data collected by all surveys is clustered in a single batch. This ensures each survey has access to a globally consistent set of labels and allows the habitat maps produced by individual surveys to be compared. The same clusters, shown in Figure 4.14 are used. Compared to Figure 4.15, the *Silt* and *Screw-Shell Rubble* classes have been merged into a single *Sand* class. This was done as these classes are location specific and are only observed by the *waterfall* and *ohara* dives (see Figure 4.17). To provide a set of labels common to all dive sites, *Silt* and *Screw-Shell Rubble* are considered to be *Sand*. The resulting habitat clusters are shown in Figure 5.6.

A habitat model trained on all of the available data provides a benchmark for habitat map accuracy. If the surveys are pooled together a total of 11,997 data points are available for training. In this chapter, a probabilistic targets least squares classifier (PTLSC) [4, 75] is used to create benthic habitat models. Due to the cubic complexity of the GP-based model, only 6000 randomly selected observations are used to train the model. The re-substitution accuracy for all of the data is calculated using *all* observed data points for interpolation and *all* of the data for testing. The accuracy given by this method is 88.21% and is provided as an optimistic, best-case reference since the test data is also used in the GP kernels for interpolation. Accuracy is lost on the sand-reef ecotone cluster which is misclassified as either sand or reef. This confusion is unsurprising as sand-reef ecotone is a mixture of both sand and reef.

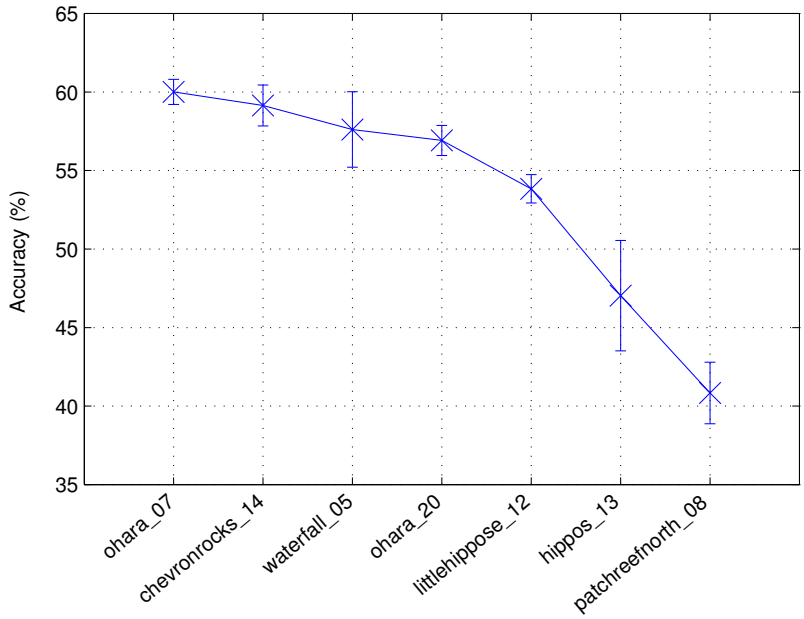
To estimate how well the data observed by a survey generalises to the entire environment, a habitat model is created using 90% of the data selected randomly. The remaining 10% is set aside for testing. This is done for each survey. The habitat models are validated on the test data from *all* surveys pooled together. Scoring the accuracy in this way ensures no training data appears in the test set. It also allows the habitat models to be compared fairly by providing an independent data set for validation which is available to all models. This process is repeated for 10 different random folds and the results are averaged across the folds.

The predicted and true KLD for each survey is shown in Figure 5.7a. The surveys *waterfall_06*, *littlehippoN_11*, *ChevronRockN_10* and *hippoN_09* are omitted from this analysis as they traverse holes in the bathymetry. The mean and variance of the validation accuracy for each survey is shown in Figure 5.7b. Both figures in Figure 5.7 share the same survey order to facilitate comparison. The surveys are ordered by validation accuracy. Although the predicted KLD closely models the true KLD, it is clear that an exact relationship between habitat map accuracy and survey utility does not exist.

All surveys with a low KLD perform well. These surveys tend to be larger, providing them with more information to correctly model decision boundaries. However, not all surveys with a high KLD perform badly. For instance *chevronrockS_14* has the fourth-



(a) Predicted KLD



(b) Habitat map accuracy

Figure 5.7 – Validation of the proposed exploration method. **(a)** Predicted and true Kullback-Leibler divergence (KLD) for each survey. The true KLD is shown in red. The predicted KLD and two standard deviation variance, supplied by the Gaussian process (GP) utility model, is shown in blue. The asymmetric variance bounds are caused by converting from the log-KLD space back to the linear-KLD space. **(b)** Mean and variance of the habitat map accuracy produced by each survey for 10 validation folds.

worst true KLD yet it achieves the second-highest accuracy. A survey with a high KLD can still collect observations which accurately identify the decision boundaries. This indicates that a habitat model's performance does not strongly depend on the density of observations. The conclusion drawn from these results is that minimising the survey KLD is a reasonable heuristic for recommending surveys which generate accurate habitat models. The method favours larger surveys placed in diverse areas of the bathymetry. This conclusion comes with the caveat that it is difficult to make strong statements about the performance of surveys with a high KLD.

The habitat map produced using all of the available data from the survey with the lowest KLD, *ohara_07*, is shown in Figure 5.8. This survey also produces the most accurate habitat map. Figure 4.20 and Figure 5.11b show that the predicted environment is predominantly sand with kelp common in 30-50m depths and reef common in 30-70m depths.

The habitat map created in Section 4.5.2 used data from all of the surveys. The habitat map using all available data predicts less reef and more sand classes (Figure 4.22) than the habitat map predicted using only *ohara_07* data (Figure 5.11). The habitat map created using only *ohara_07* data predicts a large volume of reef in the 80-90m depth band. This prediction is made with a low confidence and contributes to a bias in reef predictions. Both habitat maps predict the same amount of *Kelp*.

The mean variance calculated across the habitat latent functions is shown in Figure 5.9. The figure shows that the habitat model has learnt a strong correlation with bathymetry features that occur close to those observed by *ohara_07*. These features are typical in the 60-70m depth band. In distant regions of the feature space such as depths beyond 75m, the maximum depth of the survey, there is no data to support inference and the predicted variance is high.

The habitat map created in Section 4.5.2 has more data to perform inference. As a result, the variance of the habitat map in Section 4.5.2 is lower than the habitat map presented in this section. This can be observed by comparing Figure 4.21 and Figure 5.9. This illustrates the benefit of having more data available for training.

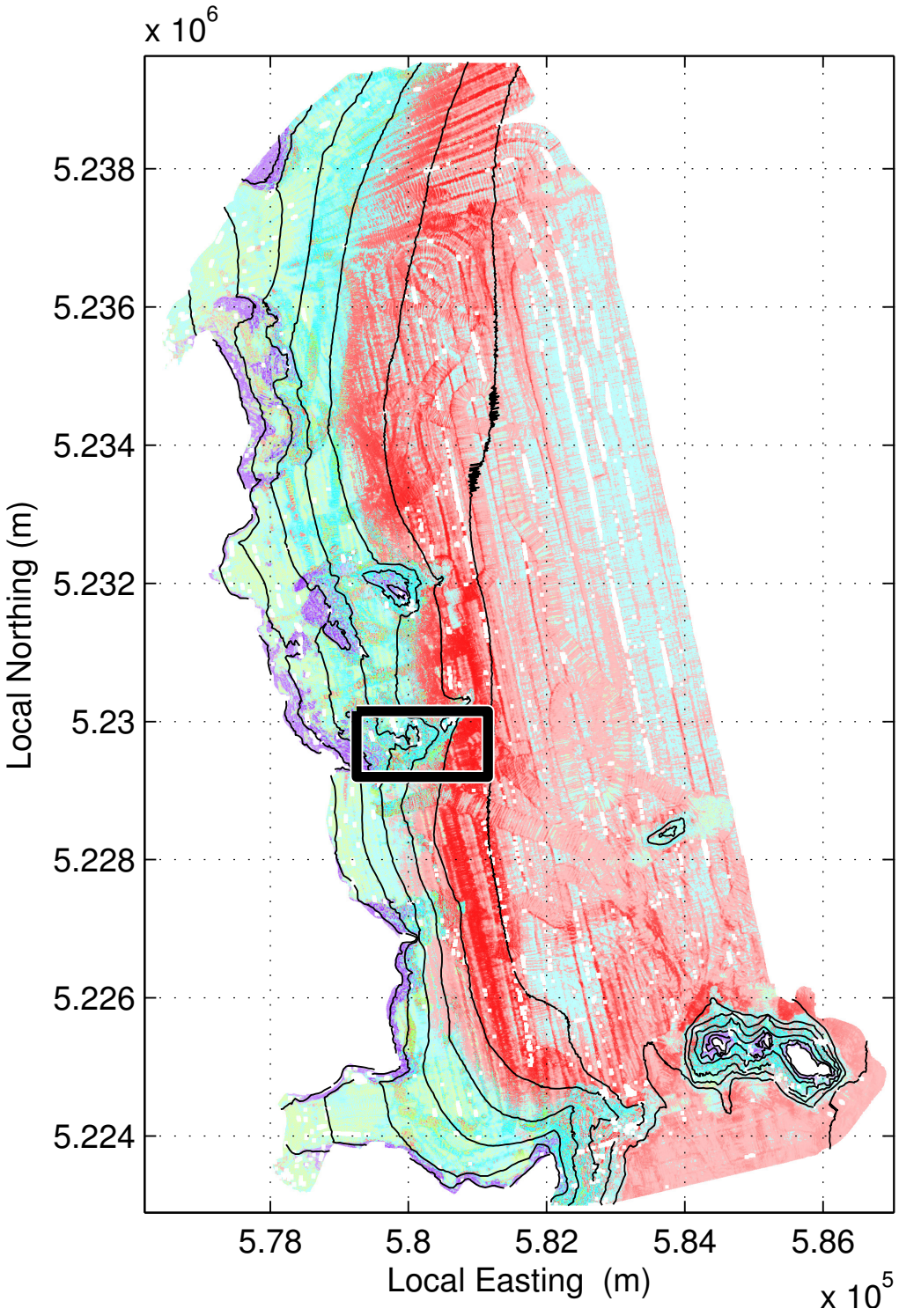


Figure 5.8 – Habitat map created from *ohara_07*. The bathymetry contours are shown as black lines. The intensity plot shows the most likely habitat where the intensity of the colour is proportional to the probability of the most likely habitat. As the predictions become less certain, the colour fades to white. Colours correspond to the habitat classes shown in Figure 5.6. A detailed section of the map, denoted by the black box, is shown in Figure 5.10. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.

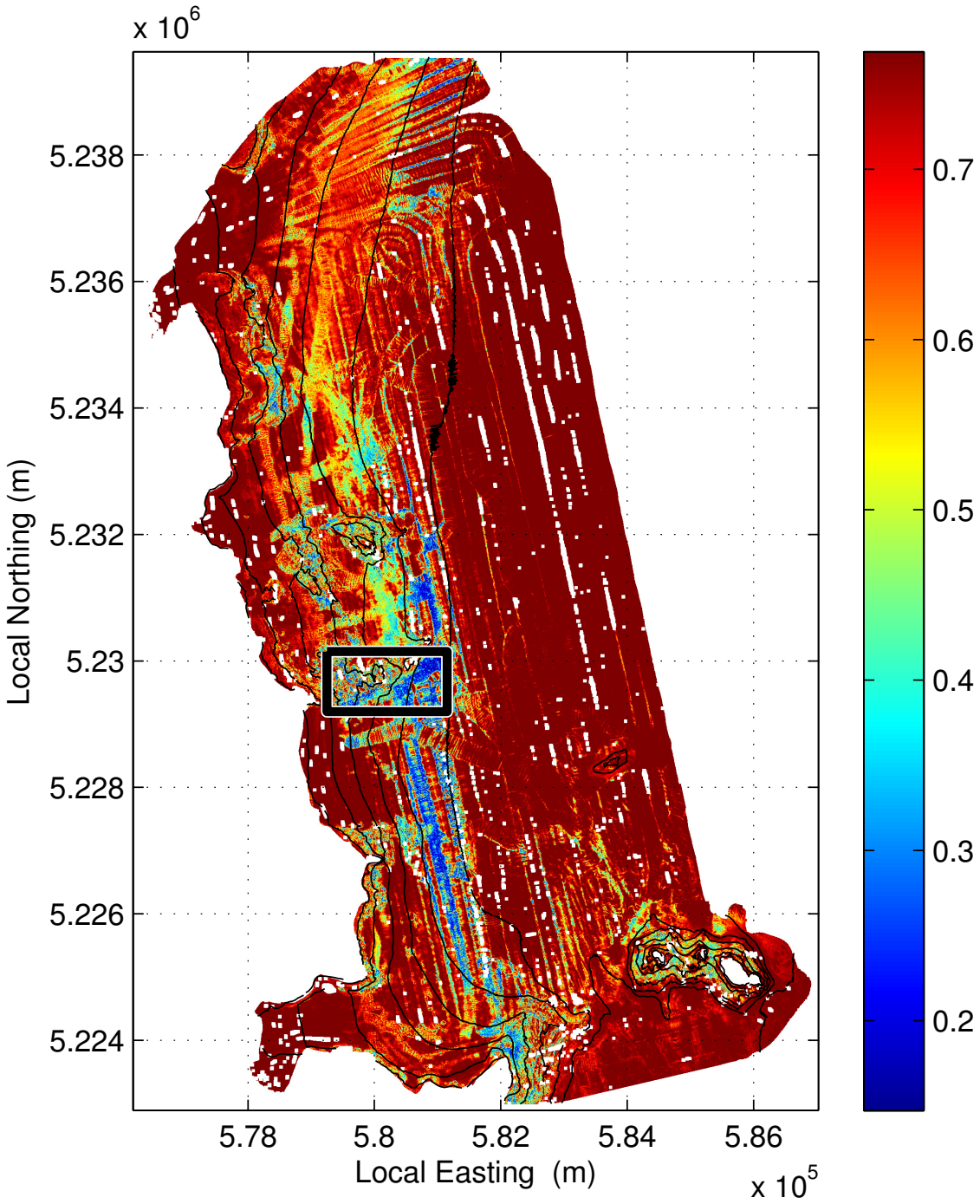
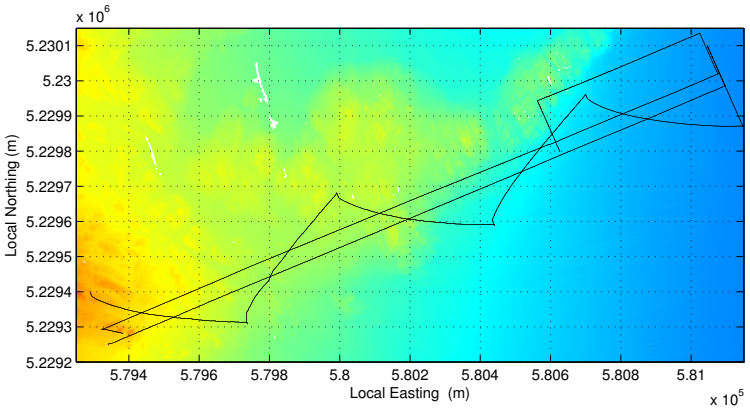
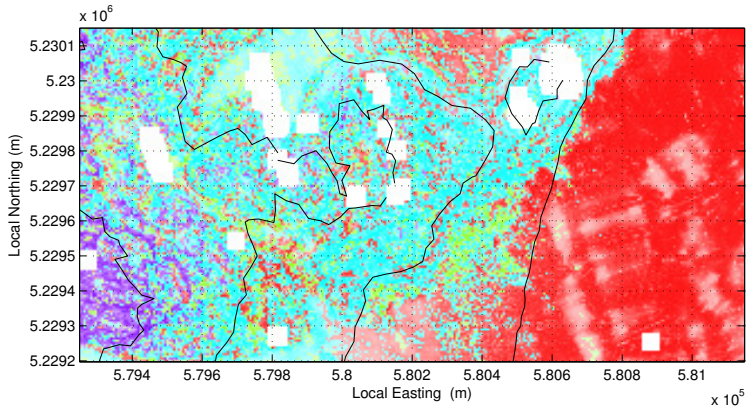


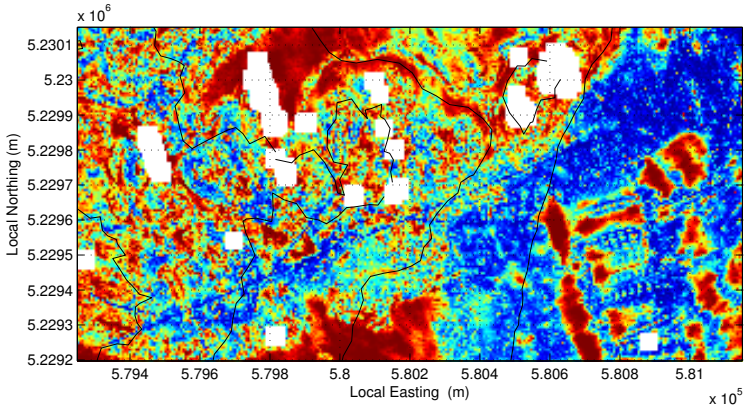
Figure 5.9 – Variance of habitat map created from *ohara_07*. The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. A detailed section of the map, denoted by the black box, is shown in Figure 5.10. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.



(a) Detail of survey region

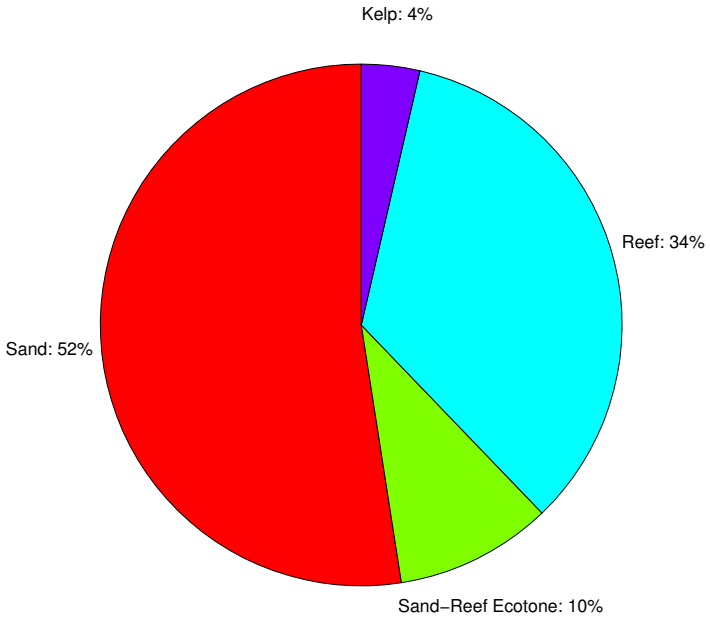


(b) Detail of predicted probability

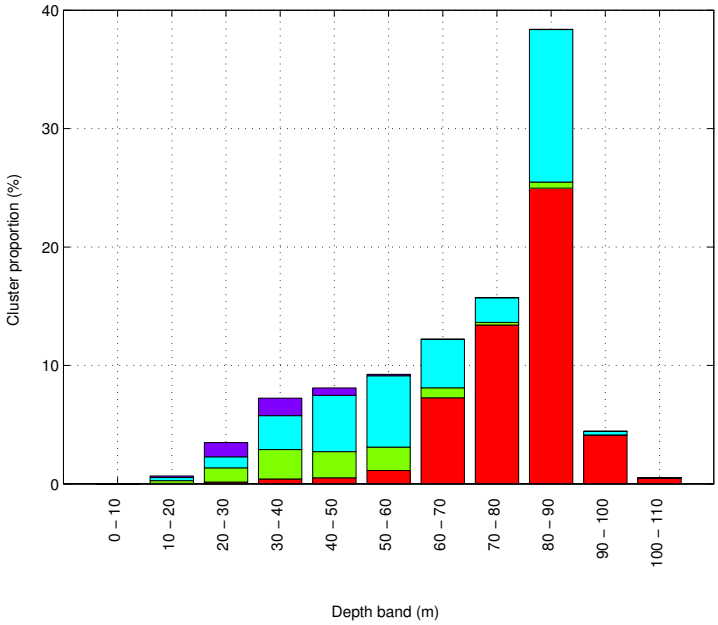


(c) Detail of mean predicted variance

Figure 5.10 – Detail of habitat map created from *ohara_07*. **(a)** Survey region showing the autonomous underwater vehicle (AUV) trajectory. **(b)** Predicted probability of habitat clusters. **(c)** Mean variance of the habitat Gaussian process (GP) latent functions. Local coordinates are expressed in zone 55G of the universal transverse Mercator coordinate system.



(a) Predicted proportion of the environment occupied by each habitat



(b) Habitat composition at 10 metre depth intervals

Figure 5.11 – Habitat composition of southeastern Tasmania using only data from *ohara_07*. **(a)** Predicted proportion of habitats in southeastern Tasmania. **(b)** Habitat composition at 10 metre depth intervals. Each bar in the graph represents a specific depth band. Each bar is composed of habitats found in the depth band as a proportion of the total environment.

5.5 Summary

This chapter presented a method for planning informative surveys in large-scale, unexplored environments where a low-resolution representation of the environment is available. The method focuses on where to place a known survey template. Platform operators are freed from the responsibility of specifying an initial and goal state. Instead they are required to design a survey trajectory. Given this trajectory, the proposed exploration method is able to learn the underlying survey utility and recommend an optimal survey location.

Many utility functions are found in the literature to guide active learning or experimental design. A short review of these measures is provided. All of the methods described in the review rely on evaluating properties of a model that already exists. For exploration problems where no model exists a method for using the KLD as a utility function is proposed. This novel usage of the KLD is applied to the problem of selecting an informative AUV survey location in a large, unobserved marine environment. The objective is to gather data which will create an accurate habitat map.

The results showed that the proposed exploration method is able to produce a functional representation of the survey utility which closely approximates the true survey utility in regions of interest. The surveys recommended by the proposed exploration method result in habitat maps which are accurate over a broad region of the environment.

Chapter 6

Planning with Prior Observations

In the previous chapter a general planning framework is proposed. The framework is applied to the problem of nominating a survey location in a large unobserved marine environment. This is achieved by using the Kullback-Leibler divergence (KLD) as a novel planning heuristic. Using the KLD rewards the agent for traversing a collection of bathymetry features which closely match the density of bathymetry features in the environment. Although this heuristic promotes coverage of the bathymetry feature space, it does not consider the relationship between the bathymetry and the observed habitat classes.

As observations are gathered from the environment and converted into a model it is possible to exploit properties of the model to guide further exploration. Closing the loop between decision making, observations and modelling allows a planning algorithm to directly optimise a statistical property of the environmental model.

In this chapter, the planning framework proposed in Section 5.3 is extended to planning scenarios where a model of the environment already exists. Section Section 6.2 discusses related work. In Section 6.1 a utility function for planning surveys with prior observations is recommended. Section 6.3 describes a marine data set collected from Sydney, Australia. The planning framework is tested in Section 6.4 where an autonomous underwater vehicle (AUV) was deployed to collect validation data. Section 6.5 concludes the chapter.

6.1 Utility Function

Several commonly used utility functions found in the fields of experimental design and active learning are reviewed in Section 5.2. All of the utility functions apply to classification models and are applicable to bathymetric habitat mapping. Several of the utility functions reviewed in Section 5.2 are discussed in greater detail in this section.

Krause et al., 2008 [45] studied the effectiveness of using the mutual information criteria in spatial modelling problems involving Gaussian processes (GPs). The research is demonstrated on a sensor placement problem where the objective is to place sensors to optimally model temperature and precipitation. For sensor network applications, Krause et al. attribute the mutual information criteria to [15]. As discussed in Section 5.2.3, an information theoretic approach to experimental design was proposed early in the statistics literature [53]. By applying these principles to experimental design, the mutual information criteria can be derived as a measure of information gain.

Singh et al., 2009 [88] use the mutual information criteria to plan paths for multiple robots. The objective is to model the chemical properties of a lake for the purpose of monitoring algal blooms. Both publications address experimental design in GP regression problems. They also assume fixed observation locations where the challenge is to select the subset of locations which maximises the amount of information gathered from the environment. The difference between the two approaches is that the planning method proposed in [88] is designed to consider path costs during planning.

Maximising mutual information, I, aims to reduce the predicted variance at all unobserved locations, ξ. In the sensor placement problem specified by Krause et al., the goal is to find a subset of observation locations, $\mathcal{E}$, that maximally reduces the entropy, $\mathrm{H}[\cdot]$, over the rest of the space $\xi \setminus \mathcal{E}$. Mathematically this is given by Equation (6.1).

$$I\left[\mathbf{X}_{\xi\setminus\mathcal{E}}, \mathbf{X}_{\mathcal{E}}\right] = \mathrm{H}\left[\mathbf{X}_{\xi\setminus\mathcal{E}}\right] - \mathrm{H}\left[\mathbf{X}_{\xi\setminus\mathcal{E}} \;\middle|\; \mathbf{X}_{\mathcal{E}}\right] \tag{6.1}$$

If a GP is used to model the observation locations, the predictions take the form of a multivariate Gaussian. The entropy of a multivariate Gaussian is given by

$$\mathrm{H}\left[\mathbf{X}\right] = \frac{1}{2}\left[\ln\left|\mathbf{\Sigma}\right| + D\left(\ln\left(2\pi\right) + 1\right)\right], \tag{6.2}$$

where $|\mathbf{\Sigma}|$ is the determinant of the covariance matrix. Since a GP is being used to model the observations, the covariance matrix, $\mathbf{\Sigma}$, is given by the predictive covariance of a GP. This equation is given in Section 2.2 by Equation (2.8) and is repeated here for convenience.

$$\mathbb{V}[\mathbf{f}_*] = \mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) - \mathbf{K}(\mathbf{X}_*, \mathbf{X})\left[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_n^2\mathbf{I}\right]^{-1}\mathbf{K}(\mathbf{X}, \mathbf{X}_*)$$

An important property of GP inference is that the posterior variance is independent of the observed targets $\mathbf{y}$. It is only a function of the observation locations $\mathbf{X}$ and the query locations $\mathbf{X}_*$. This provides a closed-form solution to Equations (6.1) and (6.2) for arbitrary unobserved query locations. The effect of selecting the observation locations $\mathbf{X}_\mathcal{E}$ on the entropy of $\mathbf{X}_{\xi\setminus\mathcal{E}}$ can be evaluated without actually observing the target values at $\mathbf{X}_\mathcal{E}$. By operating with conjugate distributions, the GP framework avoids the requirement to solve the cumbersome integrals given in Equation (5.8) with numerical sampling. This makes Bayesian experimental design very efficient in GP regression.

Benthic habitat mapping is a classification problem, not a regression problem. Statistical utility functions applied to optimal decision making with GP classifiers have also been proposed in the literature. Kapoor et al. [41] propose using the posterior mean and the posterior variance to select unobserved data where classification is uncertain. Unlike mutual information, the method proposed in [41] does not consider the impact a new observation will have on the remaining unobserved data. Instead each unobserved instance of data is ranked independently and the most favourable instance is selected for analysis.

Multiclass probabilistic classifiers return their predictions as multinomial distributions. Each multinomial prediction is a vector where the elements contain the probability that the test data belongs to a specific class. The entropy of a multinomial distribution is given by,

$$\mathrm{H}\left[\mathbf{x}_*\right] = -\sum_{C=1}^{K} p(C \mid \mathbf{x}_*) \log p(C \mid \mathbf{x}_*)\,,$$

where K is the number of classes and $p(C \mid \mathbf{x}_*)$ is the probability of observing class C at location $\mathbf{x}_*$. Rigby et al., 2010 [78] use this form of entropy to calculate mutual information. The calculation is generalised to multiple observations by calculating the mean entropy over a set of query locations.

To simplify notation, the input locations $\xi \setminus \mathcal{E}$ will be expressed as $\mathbf{X}_*$. Rigby et al. calculate the mutual information across all locations in the environment which define the habitat map. The conditional entropy for the habitat map updated with data from a candidate survey is expressed as

$$\mathrm{H}[\mathbf{X}_* \mid \mathbf{X}_\mathcal{E}] = -\sum_{i=1}^{N}\sum_{C=1}^{K} p(C_{i*} \mid \mathbf{X}_{i+}, \mathbf{y}_{i+}, \theta) \log p(C_{i*} \mid \mathbf{X}_{i+}, \mathbf{y}_{i+}, \theta)\,, \tag{6.3}$$

where N are the number of query locations, $\mathbf{X}_+$ is given by $\mathbf{X} \cup \mathbf{X}_\mathcal{E}$ and $\mathbf{y}_+$ is given by $\mathbf{y} \cup \mathbf{y}_\mathcal{E}$. All of these variables are known except for $\mathbf{y}_\mathcal{E}$. Rigby et al. note that this formulation of the posterior entropy *does* depend on observed target values.

Unlike the approach adopted by Krause et al., Equation (6.3) depends on the predictive mean of the GP, via Equation (2.11). The predictive mean is given by Equation (2.7) and is repeated here using the variables from Equation (6.3)

$$\mathbb{E}[\mathbf{f}_*] = \mathbf{K}(\mathbf{X}_*, \mathbf{X}_+)\left[\mathbf{K}(\mathbf{X}_+, \mathbf{X}_+) + \sigma_n^2\mathbf{I}\right]^{-1}\mathbf{y}_+$$

Since $\mathbf{y}_+$ depends on the unobserved values contained in $\mathbf{y}_\mathcal{E}$, to evaluate Equation (6.3), Rigby et al. use an expensive Monte Carlo sampling algorithm to simulate values of $\mathbf{y}_\mathcal{E}$.

In this chapter, models are evaluated by the confidence of their predictions. High-quality models are models which produce confident predictions. The utility function nominated for exploration must appropriately reflect this standard. The mutual information criteria given in Equation (6.1) satisfies this objective.

Although Rigby et al. are correct in identifying that the mutual information of the predicted probabilities of a GP classifier depend on future observed values, they overlook the role of the GP latent function in classification. The mean of the GP latent function is squashed through a sigmoid function to produce predictive probabilities. One way of interpreting the latent function predictive variance is that it specifies a distribution over the predicted probabilities.

In both regression and classification, the predicted variance represents confidence in the mean prediction. When the model is confident about its hypothesis, the predicted variance is low and when the model lacks confidence, the predicted variance is high. The volume of this hypothesis space can be measured by its entropy using Equation (6.2). The goal is to make the hypothesis space as small as possible. The mutual information criteria encourages collecting observations which achieve this goal.

An alternative way of interpreting the predicted variance of the latent function is that it represents the level of correlation within the data. Highly correlated data will have a low-variance and poorly correlated data will have a high-variance. The degree to which data points are correlated is determined by the hyperparameters of the covariance function. In the squared exponential covariance function, Equation (2.9), the length scale parameter provides a measure of how much separation is needed in the input space for two points to become uncorrelated. Under this interpretation, training a GP can be considered a feature-learning exercise where the aim is to model the degree to which observations are correlated. In bathymetric habitat mapping, the input space is comprised of bathymetric features. The mutual information criteria encourages visiting unobserved locations in the bathymetric feature space that are highly correlated with large portions of the remaining space.

In large, continuous environments there is an infinite number of observable locations. Rather than considering ξ to be a finite list of unobserved locations, as in [45], in the general case it will be considered to be the set of all possible surveys. This infinite space covers every conceivable placement of every possible survey design. A single survey selected from ξ is denoted $\mathcal{E}$. The goal is to find the survey location, $\mathcal{E}$, that maximally reduces the entropy of the latent function over the entire environment, $\mathbf{X}_\omega$. The optimal survey, $\mathcal{E}^*$ is then given by Equation (6.4).

$$\mathcal{E}^* = \arg\max_{\mathcal{E}\in\xi} \mathrm{H}\left[\mathbf{X}_\omega\right] - \mathrm{H}[\mathbf{X}_\omega \mid \mathbf{X}_\mathcal{E}] \tag{6.4}$$

The assumptions and planning framework proposed in Chapter 5 makes operating in ξ tractable. In digital elevation models (DEMs) the environment, $\mathbf{X}_\omega$ is naturally limited to the number of pixels in the raster. Although finite, the size of a DEM is likely to make multiple evaluations of $\mathrm{H}[\mathbf{X}_\omega \mid \mathbf{X}_\mathcal{E}]$ inefficient. To approximate $\mathbf{X}\omega$, a tractable subset of evaluation locations is used. The number of support points used to approximate uncertainty in the environment, $\mathbf{X}\hat{\omega}$, can be selected based on the available computing resources. Their distribution in the environment can be selected according to the scientific objectives of a survey.

The amount of uncertainty in the environment, $\mathrm{H}\left[\mathbf{X}_\omega\right]$, is a constant term which does not vary as a function of the nominated survey, $\mathcal{E}$. By approximating the uncertainty of the future habitat map with support points and ignoring the constant entropy term, the mutual information criteria can be rewritten as Equation (6.5).

$$\mathcal{E}^* = \arg\min_{\mathcal{E}\in\xi} \mathrm{H}[\mathbf{X}_{\hat{\omega}} \mid \mathbf{X}_\mathcal{E}] \tag{6.5}$$

This objective function minimises the uncertainty of the habitat map for a given survey location. For GP models, this is equivalent to the Bayesian D-optimal design criteria, discussed in Section 5.2.4, where uncertainty is measured using the entropy of a multivariate Gaussian.

A further simplification to Equation (6.4) can be made by adopting the A-optimal criteria. Rather than measuring uncertainty using Equation (6.2), uncertainty is measured using the trace, or sum of variances, of the posterior covariance. For multiple classes the A-optimal objective function is given by Equation (6.6).

$$\mathcal{E}^* = \underset{\mathcal{E}\in\xi}{\arg\min} \left(\frac{1}{KN} \sum_{k=1}^{K} \sum_{n=1}^{N} \mathbb{V}[\mathbf{f}_{\hat{\omega}_n} \mid \mathbf{X}_{\hat{\omega}_n} \mathbf{X}, \mathbf{X}_{\mathcal{E}}, \theta_k] \right) \tag{6.6}$$

In Equation (6.6), the latent function variance is calculated at all N environment support points in $\mathbf{X}_{\hat{\omega}}$ for each of the K classes. Rather than calculating the total predicted variance, the mean predicted variance is used as it is independent of N. This objective function estimates the future mean variance of the environment given locations in the bathymetry feature space that have already been observed, $\mathbf{X}$, and locations in the bathymetry feature space recommended by a candidate survey, $\mathbf{X}_{\mathcal{E}}$. The goal is to select the survey which produces the smallest future mean variance of the environment.

While the D-optimal utility function is mathematically more rigorous, its practical use is limited by its sensitivity to changes in the eigenvalues. In [87] the authors show that the determinant of a covariance matrix tends towards zero as a single eigenvalue approaches zero. This is problematic and can lead to situations where the uncertainty of a covariance matrix, as measured by the D-optimal utility function, may be dominated by a few small eigenvalues which prevent other dimensions from properly contributing to the measurement of uncertainty [87].

The A-optimal utility function offers two advantages. Unlike the determinant operation used in the D-optimal utility function, the trace operation used in the A-optimal utility function is robust to changes in the eigenvalues of the covariance matrix. Since the A-optimal utility function calculates the "mean" uncertainty across all states in the model, every state is weighted equally. The second advantage of using the A-optimal utility function over the D-optimal utility function is that it avoids the $\mathcal{O}(N^3)$ determinant operation in Equation (6.2). For these reasons the A-optimal utility function is used to measure the uncertainty of a habitat map.

6.2 Related Work

Section 5.2.3 shows how a utility function can be designed for information gathering tasks by following information theoretic principles. Section 6.1 applies this theory to GP models and establishes a utility function, (Equation (6.6)), which favours actions which are expected to maximally reduce uncertainty in the GP model. The utility function is based on the work of [45], [88] and [78]. Similar approaches to exploration can be found in the literature.

Binney et al., 2013 [7] propose a solution to planning informative paths in situations where the objective function is submodular and the observations are allowed to depend on the time. The work is demonstrated using an autonomous surface vehicle where the objective is to measure wireless signal strength on a lake. The work generalises the recursive greedy algorithm, which is also used in [88], to solve problems with time-varying objective functions. Like the planning algorithm presented in Section 5.1, the algorithm presented by Binney et al. is nonadaptive and generates entire paths prior to deployment.

Binney et al. have also modified the recursive greedy algorithm to include the value of information gained from observations accumulated when traversing between way-points. Similarly, the algorithm presented in Section 5.1 offers the ability to include the value of information collected at an arbitrarily fine resolution along the length of the entire vehicle trajectory. Unlike the algorithm proposed in [7], the algorithm presented in Section 5.1, is not restricted to operate on a graph.

Hollinger et al., 2013 [33] propose a method for planning a sequence of AUV view points to properly model the uncertainty of a three-dimensional ship hull reconstruction. The most salient difference between the algorithm proposed in Section 5.1 and the method proposed in [33] is that the later provides adaptive strategies which allow the vehicle to change its plan as new information becomes available. The method proposed in [33] is only able to make observations at the nominated view points. This restriction is reasonable for inspection tasks but hampers environmental monitoring where it is natural to gather observations of the seafloor between way-points.

To plan a vehicle trajectory, the sampling-based roadmap method used in [24] is used to nominate view points. View points which are expected to minimise the uncertainty of the three-dimensional model of the ship hull are then greedily selected. Once a sufficient number of view points have been selected or the expected variance drops below an acceptable level, the lowest cost tour between all points is generated.

Both [7] and [33] plan informative paths using a GP model and score information using variance reduction, which is equivalent to the A-optimal utility function. The main difference between the objective function given by Equation (6.6) and the objective function used in [7, 33] is that Equation (6.6) produces an approximation to the total variance by using environmental support points. The applications specified in [7] and [33] do not operate in large environments. Due to a lower number of query locations, both methods can afford to calculate the variance at all areas in the environment. The applications specified in [7] and [33] are also regression problems. Habitat mapping is a classification. To calculate the variance of a multiclass classifier, Equation (6.6) averages across all the latent functions in the classifier.

Thompson et al., 2011 [94] propose an information-driven approach to planning informative paths, on kilometre scales, for planetary exploration. A GP is used to model the mapping from remotely sensed multispectral data to basalt abundance. The objective is to reduce the uncertainty of this model subject to path constraints. This application is very similar to the benthic habitat mapping problem. Both problems require the agent to learn the correlation between low-resolution, remotely sensed data and fine-scale, in-situ observations.

Thompson et al. use an objective function similar to Equation (6.5). Unlike Equation (6.6), uncertainty is calculated for the entire environment. Rather than quantifying uncertainty using the total variance of the predictions, the entropy of the predictions is used instead. This is equivalent to D-optimal design. Observation locations are selected using a modified version of the recursive greedy algorithm. The modified algorithm ensures the vehicle reaches a goal location and encourages the vehicle to allocate its resources evenly. In contrast to the method proposed in Section 5.1 the value of observations gathered between way-points is not considered.

6.3 Bate Bay Data Set

6.3.1 Bathymetry

Ship-borne multibeam echo-sounder (MBES) bathymetry was collected from Bate Bay, about 25km south of Sydney, Australia by the N.S.W Office of Environment and Heritage (OEH). The survey region is shown in Figures 6.1 and 6.1. The bathymetry is rasterised to a pixel size of 2.0 metres and is 6.298km tall and 5.150km wide. There are 8,114,400 valid data points in the raster which cover an area of approximately 32.46km^2.

Rugosity and slope for the Bate Bay data set were calculated at scales of 2 (10m), 4 (18m), 8 (34m) and 16 (66m) and are shown in Figures 6.3 and 6.4. Log rugosity and log slope are used to increase 'contrast' in the data. The multibeam data collected in Bate Bay is a preliminary data product and has not been optimised.

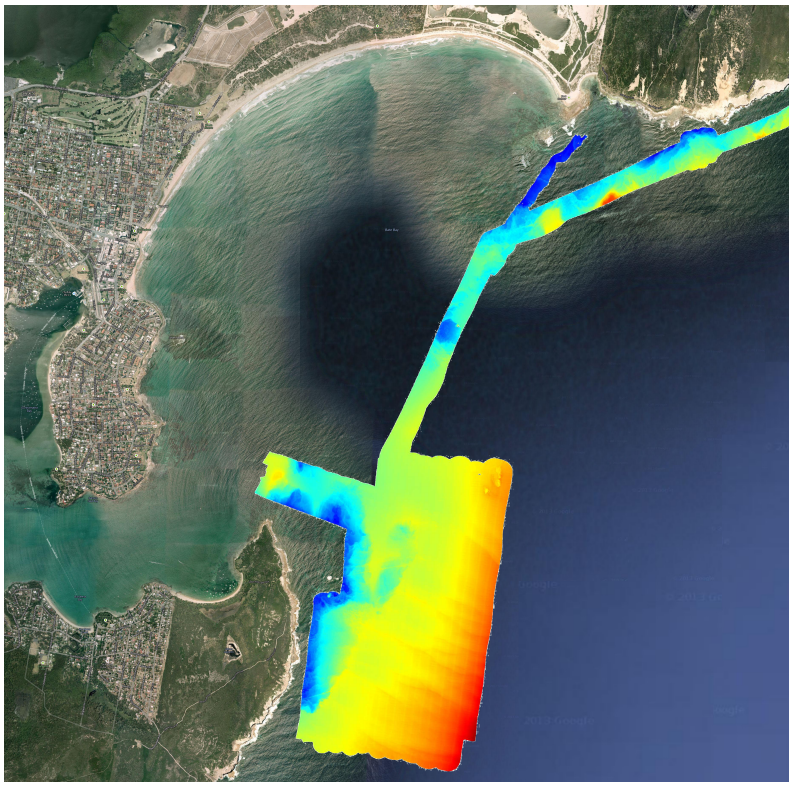


Figure 6.1 – Multibeam bathymetry of Bate Bay survey region layered on top of satellite imagery obtained from Google Earth.

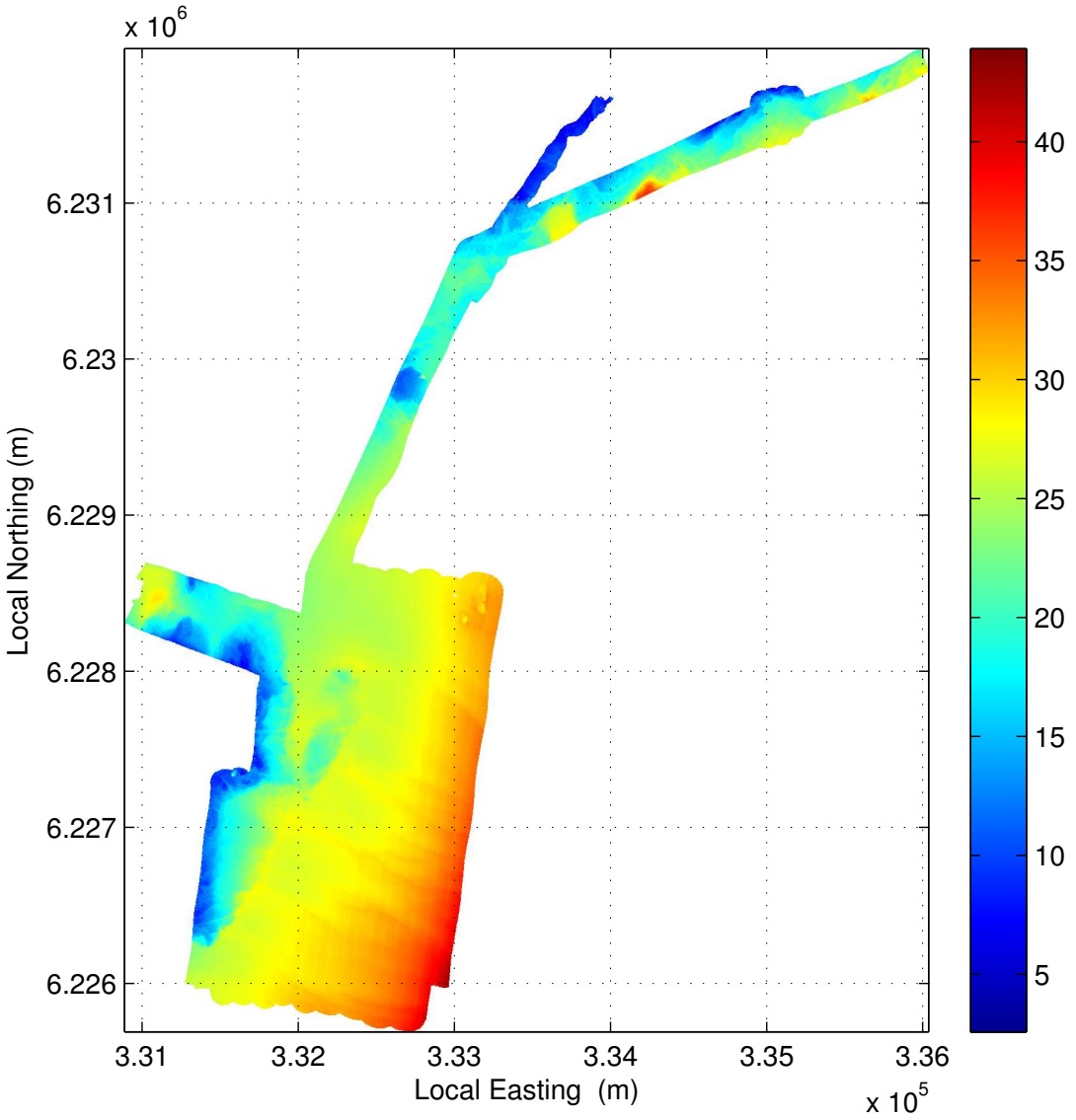
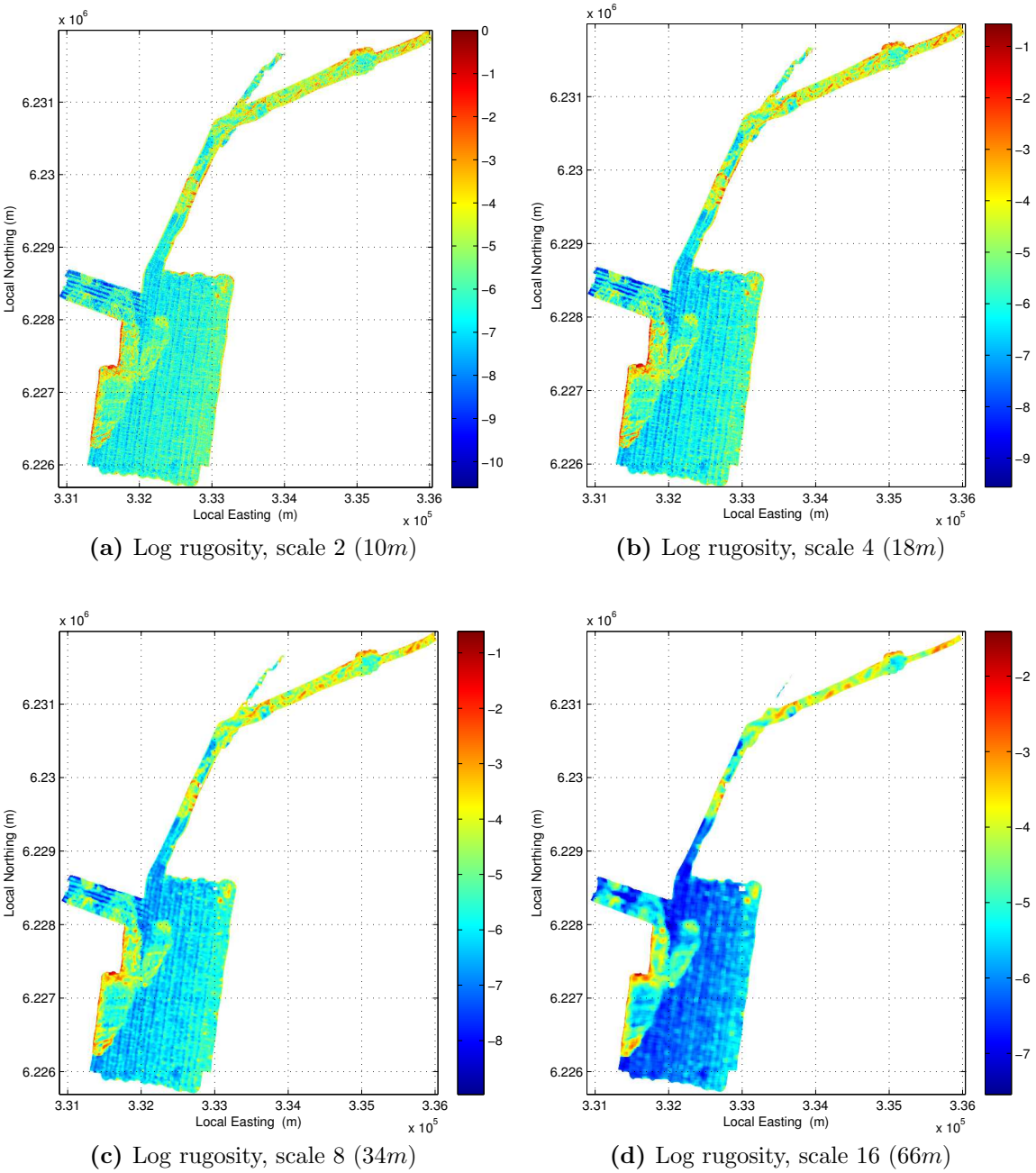


Figure 6.2 – Multibeam bathymetry of Port Hacking survey region. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.



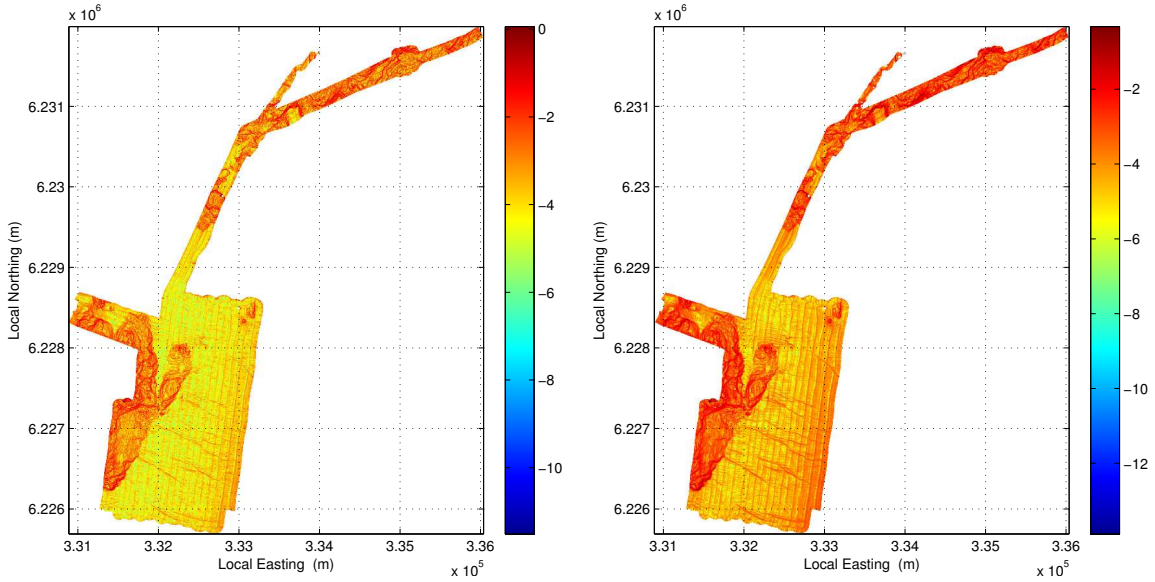
(a) Log rugosity, scale 2 ($10m$)

(b) Log rugosity, scale 4 ($18m$)

(c) Log rugosity, scale 8 ($34m$)

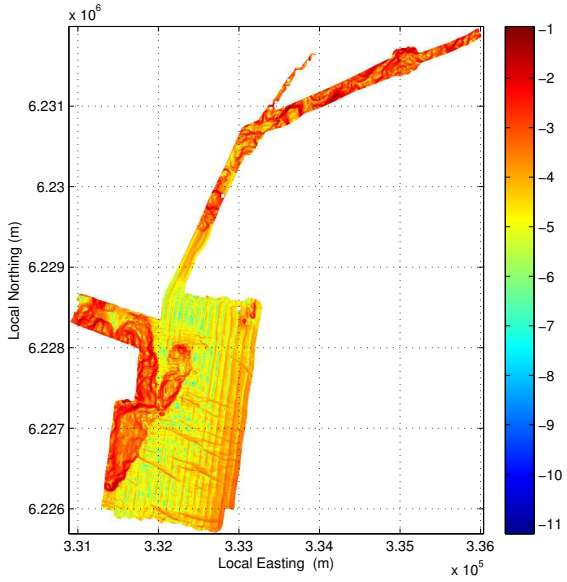
(d) Log rugosity, scale 16 ($66m$)

Figure 6.3 – Log rugosity for Bate Bay survey area calculated at four scales. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

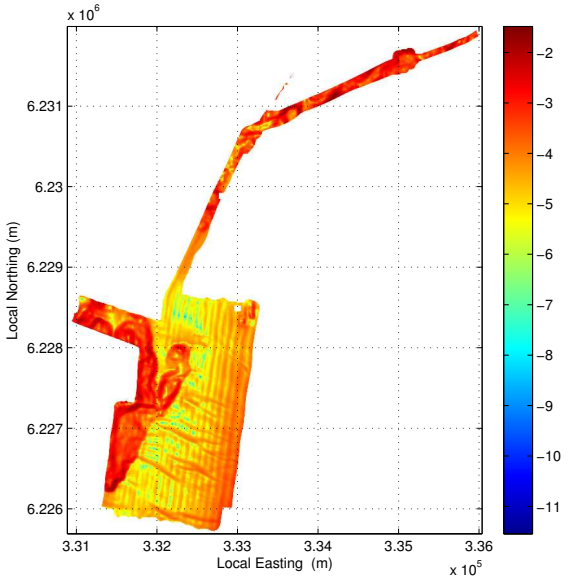


(a) Log slope, scale 2 ($10m$)

(b) Log slope, scale 4 ($18m$)



(c) Log slope, scale 8 ($34m$)



(d) Log slope, scale 16 ($66m$)

Figure 6.4 – Log slope for Bate Bay survey area calculated at four scales. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

6.3.2 Surveys

An optical survey of the Bate Bay bathymetry was conducted using *Sirius* (see Section 4.4.2) in November 2012 and is shown in Figure 6.5. *Sirius* was programmed to perform the survey at an altitude of $2m$ and at a speed of $0.5m/s$. Summary statistics for the survey conducted in Bate Bay are shown in Tables 6.1 and 6.2.

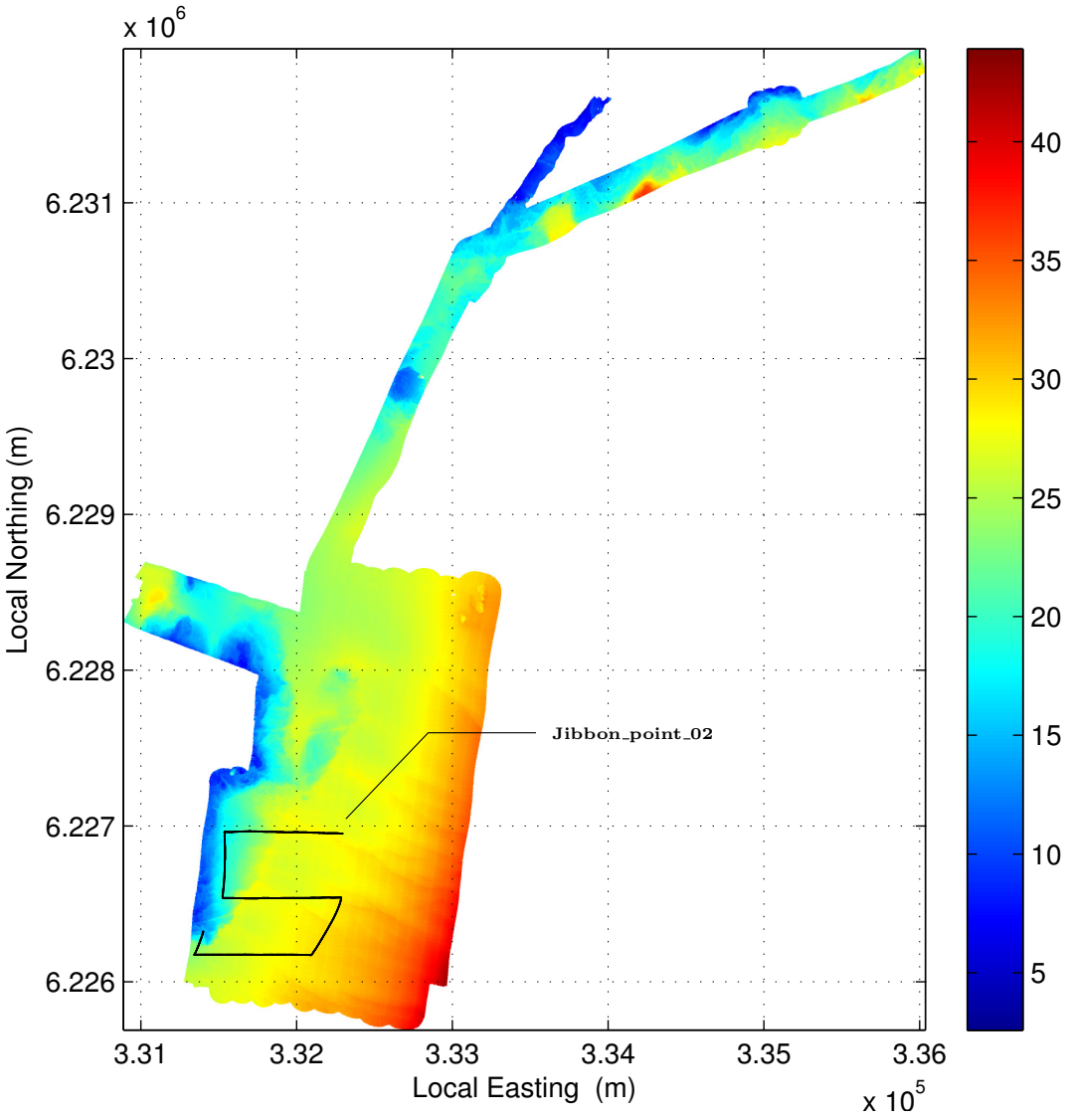


Figure 6.5 – Survey conducted by *Sirius* in Bate Bay. The survey is labelled with its survey name. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

Survey Name	Duration (h)	Distance (km)	Images Captured
Jibbon_point_02	1.88	3.37	6,736

Table 6.1 – Summary statistics of *Sirius* survey in Bate Bay.

Survey Name	Depth (m)			Altitude mean (m)
	min	mean	max	
Jibbon_point_02	9.78	22.68	27.34	2.01

Table 6.2 – Depth statistics of *Sirius* survey in Bate Bay.

6.3.3 Categorising the In-situ Imagery

The AUV image feature vector contains 22 dimensions [91] including image-based features such as colour, texture and three-dimensional stereo-image features such as rugosity and slope [25]. Compared to Section 4.4.3, the smallest stereo-imagery rugosity scale failed to be processed and was omitted from the feature vector.

To avoid poor illumination in the imagery, images below an altitude of 0.2m and above an altitude of 3m are removed. For the AUV data shown in Figure 6.5, approximately 5,115 images remain when high and low altitude images are removed. The AUV image feature vector is standardised by subtracting the mean from each dimension and dividing by the standard deviation.

The variational Dirichlet process (VDP) clustering model detected 11 clusters in the AUV imagery. Example images randomly sampled from each of the 'habitat' clusters are shown in Figure 6.6. Three habitat categories remain after the clusters shown in Figure 6.6 have been reviewed and consolidated. The consolidated habitat categories can now be given semantic labels and considered habitat classes as shown in Figure 6.7. The spatial contiguity of the habitat classes is shown in Figure 6.8.

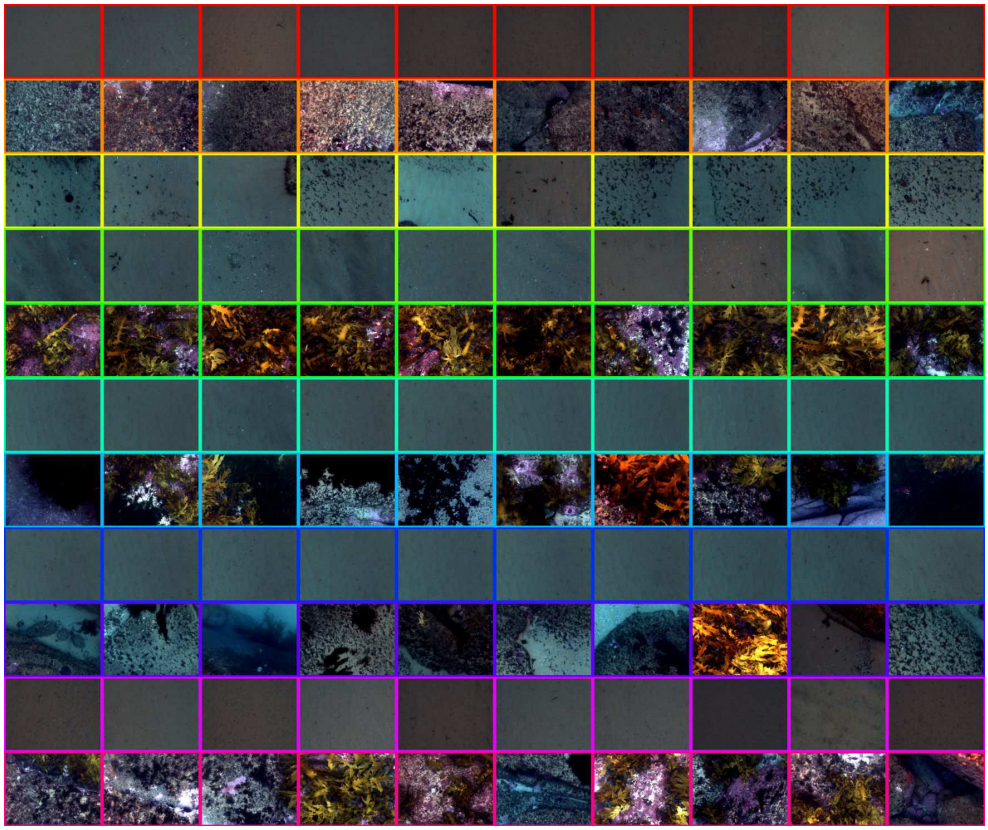


Figure 6.6 – Example images drawn randomly from the habitat clusters. Prior to human verification and merging, the VDP clusters are not associated with semantic labels.

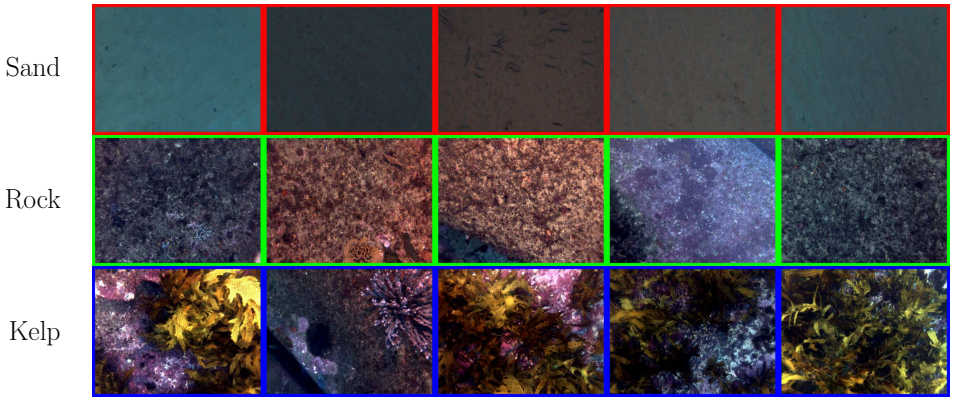


Figure 6.7 – Example images drawn randomly from the habitat classes. Once the VDP clusters have been verified and merged by a human expert, they can be assigned a semantic label and considered classes.

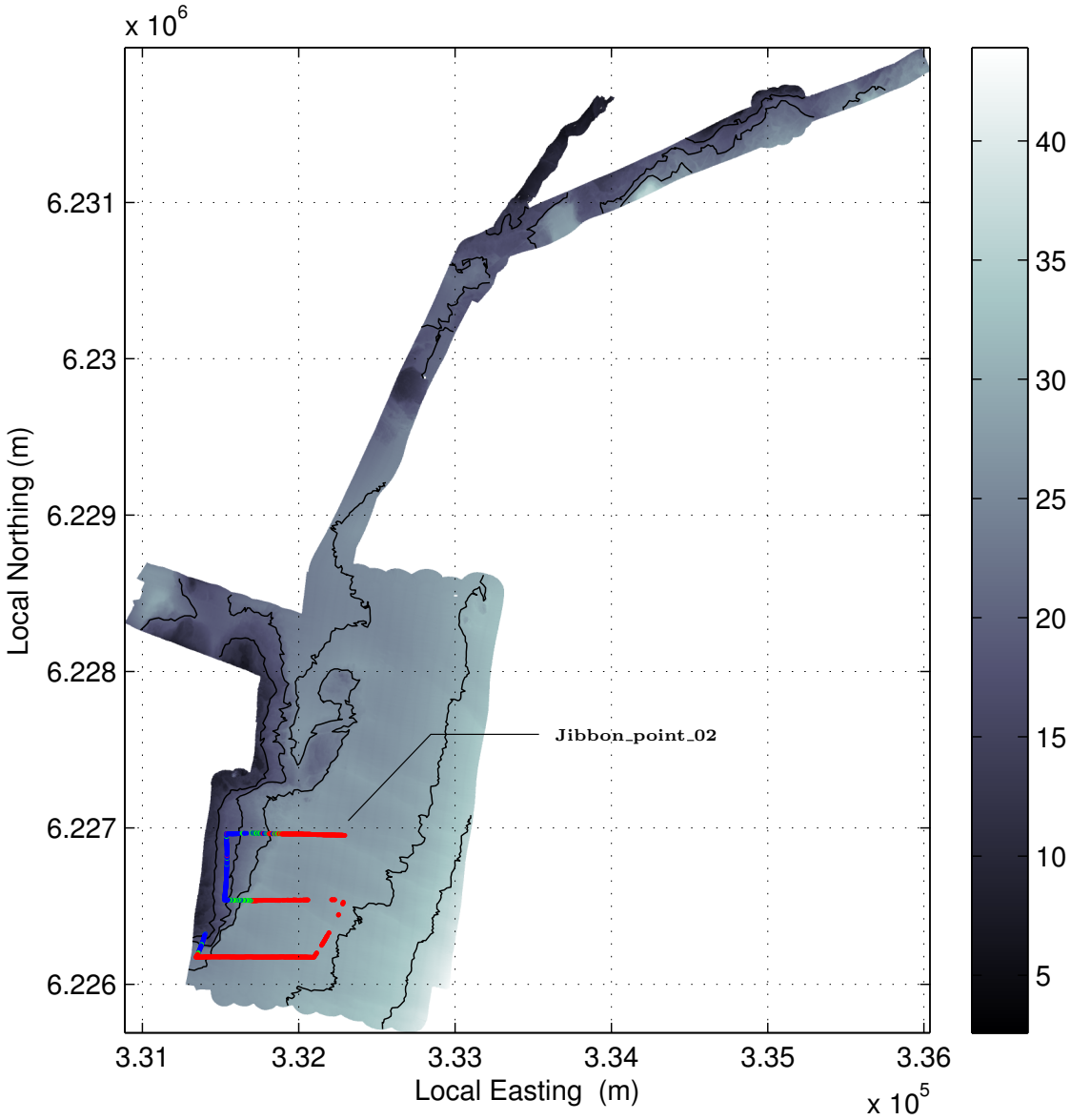


Figure 6.8 – Classified AUV surveys conducted in Bate Bay. The intensity plot and black lines represent the bathymetry and depth contours. Each AUV pose is shown as a coloured dot where the colours indicate a specific habitat class. The colours for each habitat class are shown in Figure 6.7. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

6.3.4 Habitat Map

The data presented in Section 6.3.1 and Section 6.3.3 was converted into a habitat map using a probabilistic least squares classifier (PLSC). The distribution of predicted habitats is shown in Figure 6.9 and the predicted variance is shown in Figure 6.10. The intensity of the colours in the habitat map and the prevalence of low-variance predictions indicates that the model is producing confident predictions.

Figure 6.11 shows the composition of the habitats in Bate Bay. The environment is predominantly *Sand* and has extensive fields of *Kelp*. *Rock* and *Kelp* share a common spatial distribution. This coexistence suggests that *Kelp* may prefer inhabiting *Rock* over *Sand*. More than 60% of the environment occurs between 20-30m depths where all habitats can be found. The large proportion of *Rock* estimated at depths between 30-40m, in Figure 6.11b is likely to be incorrect. In the southeast portion of the habitat map, *Rock* is predicted with a high variance (Figure 4.21). Intuition suggests that this region is more likely to be *Sand*. The high-variance in this regions signals a lack of confidence in the predictions and suggest that this bathymetry is far from observed values.

The confusion matrix for the habitat model, shown in Figure 6.12, confirms that the PLSC is modelling its training data accurately. There is some confusion where 12% of the *Rock* classes were incorrectly classified as *Kelp*. This level of confusion is acceptable as *Kelp* shares as similar distribution to *Rock* in Bate Bay and shares similar bathymetry as a result. A small amount of confusion also exists between *Kelp* and *Rock* and also between *Rock* and *Sand*. Low confidence predictions at the interface between *Rock* and *Kelp* (Figure 6.9) suggests that this confusion could be attributed to the transition between classes.

The advantage of learning a habitat map in a bathymetry feature space rather than a Cartesian space is clear in the Bate Bay habitat map. Only a small portion in the south of the environment has been observed yet confident predictions can be made up to 4km away. This is possible because habitats in the environment can be separated spatially but share a common bathymetry signature.

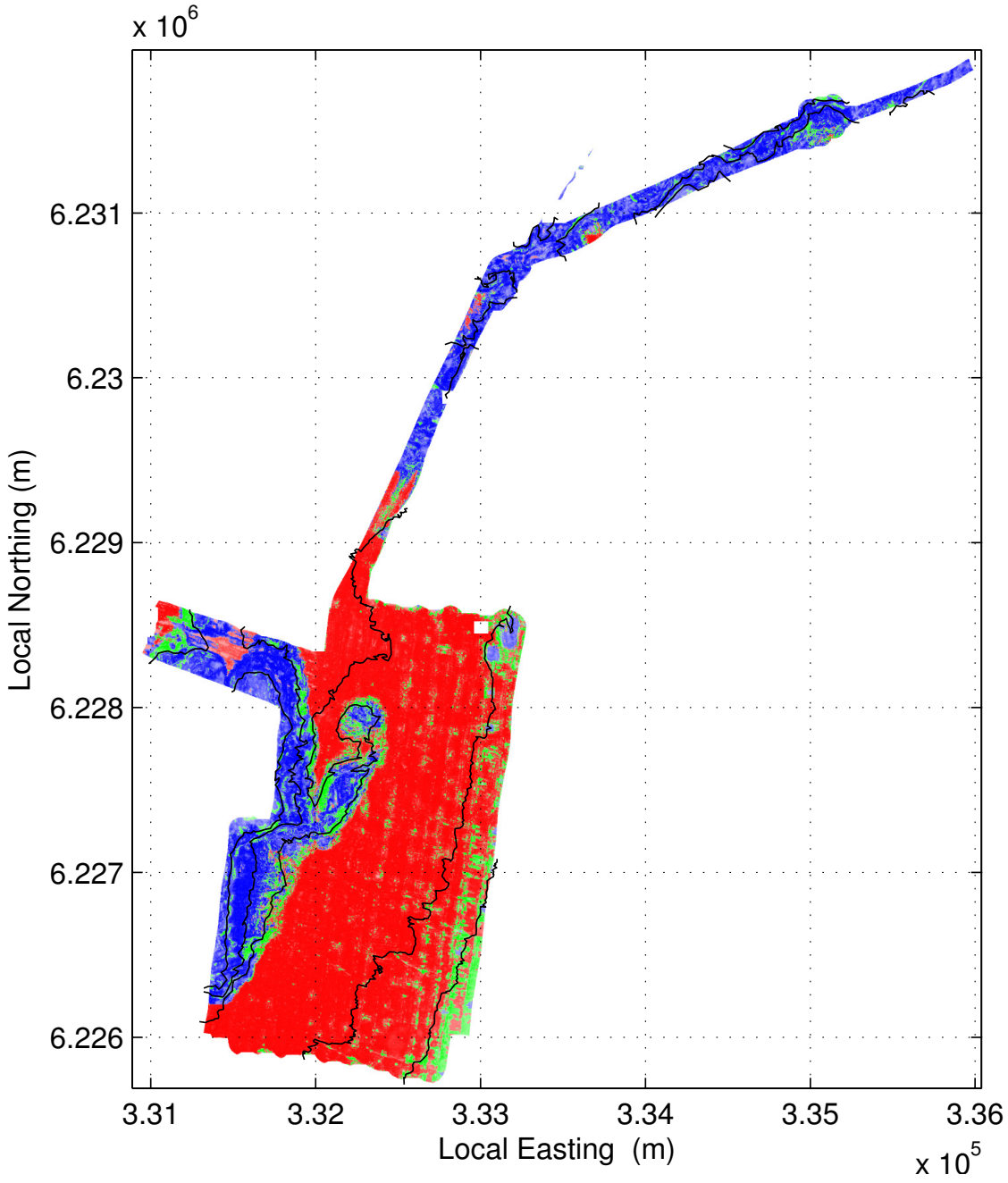


Figure 6.9 – Bate Bay habitat map generated by a probabilistic least squares classifier (PLSC). The bathymetry contours are shown as black lines. The intensity plot shows the most likely habitat where the intensity of the colour is proportional to the probability of the most likely habitat. As the predictions become less certain, the colour fades to white. Colours correspond to the habitat classes shown in Figure 6.7. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

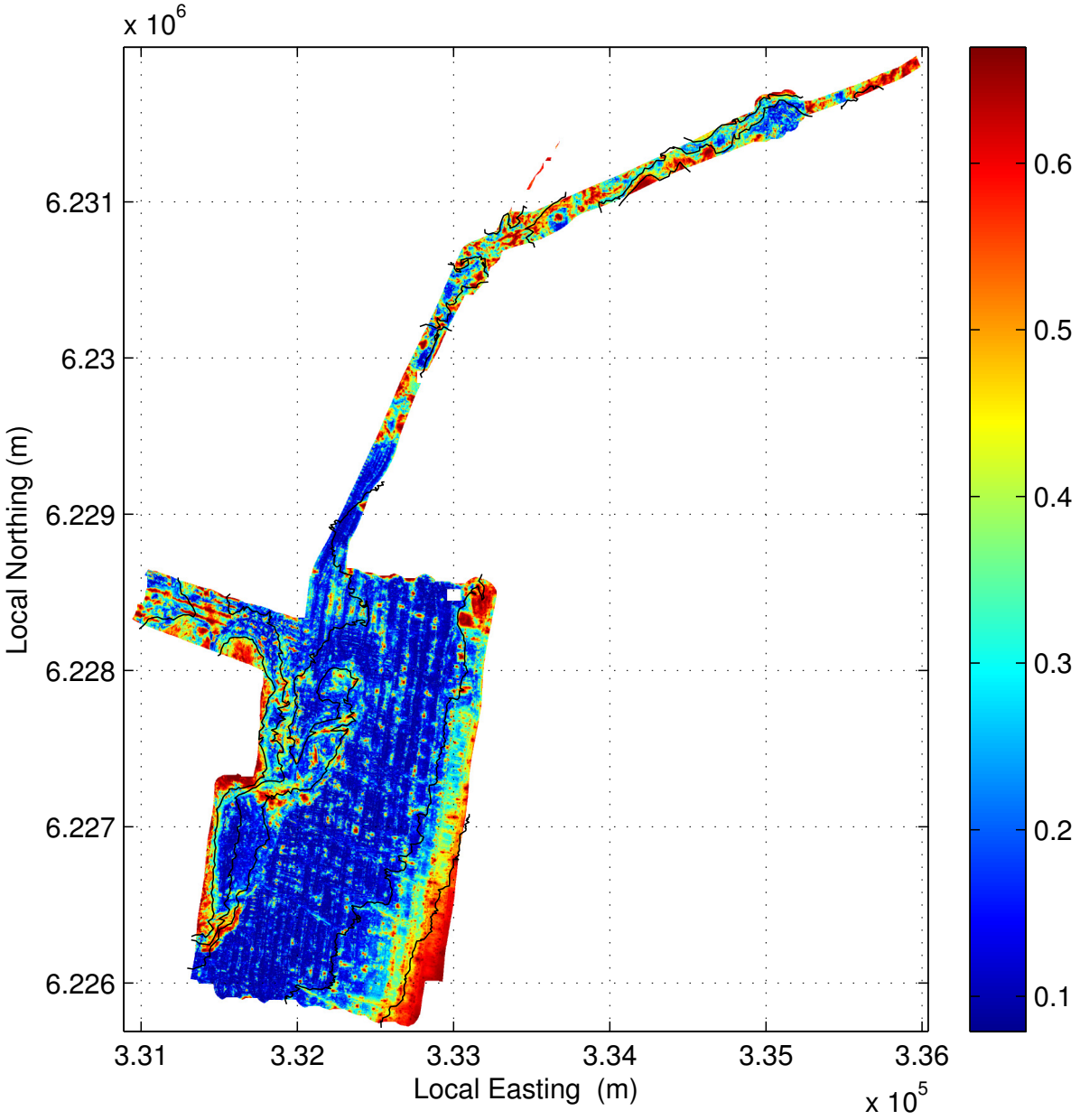
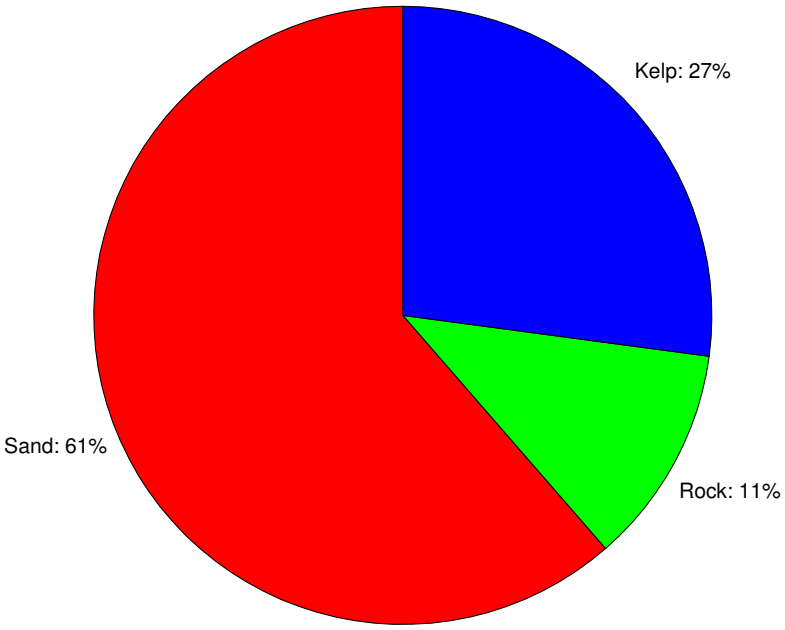
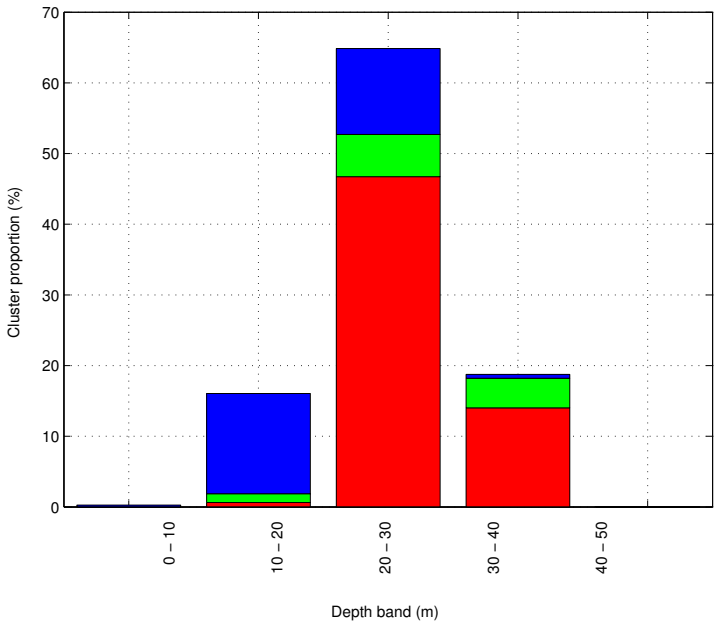


Figure 6.10 – Variance of Bate Bay habitat map generated by a probabilistic least squares classifier (PLSC). The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.



(a) Predicted proportion of the environment occupied by each habitat



(b) Habitat composition at 10 metre depth intervals

Figure 6.11 – Habitat composition of Bate Bay. **(a)** Predicted proportion of habitats in Bate Bay. **(b)** Habitat composition at 10 metre depth intervals. Each bar in the graph represents a specific depth band. Each bar is composed of habitats found in the depth band as a proportion of the total environment.

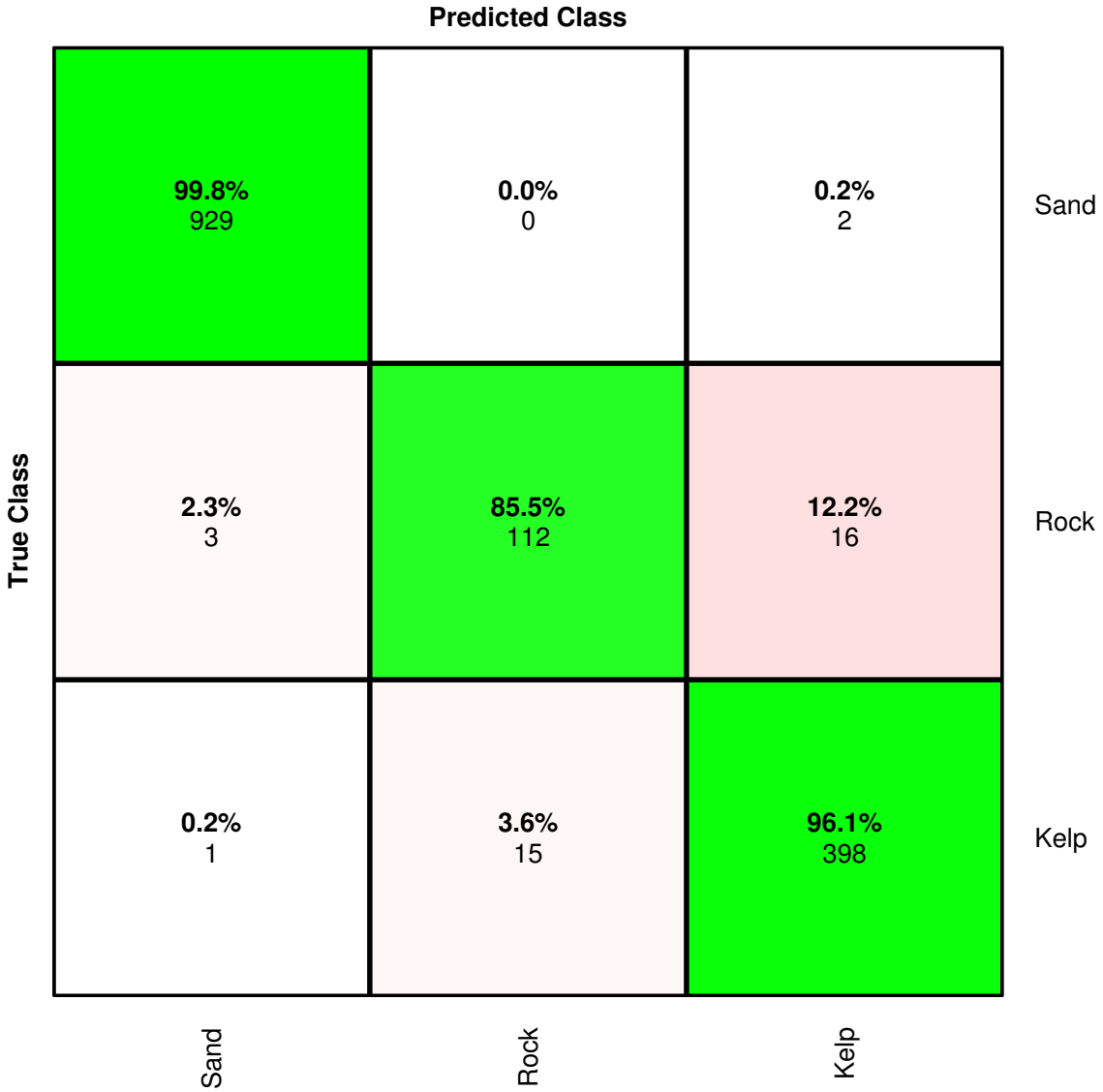


Figure 6.12 – Confusion matrix for the PLSC. The confusion matrix was calculated by using all of the data for training and all of the data for testing. Data re-substitution is used to provide an optimistic indication of model performance, not necessarily an indication of model generalisation.

6.4 Planning with Observations

In this section the planning framework proposed in Chapter 5 is used with the utility function proposed in Section 6.1. The planning framework is demonstrated on the Bate Bay data set described in Section 6.3.

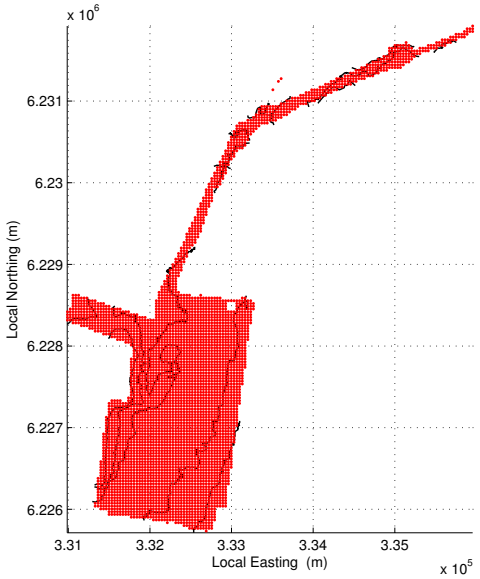
The planning process is almost identical to the process described in Section 5.3. Like Section 5.3 the process begins by specifying a survey template. In this demonstration a $500m$ West-East transect is used. This simple survey roughly follows the easterly gradient of the bathymetry. To train the GP utility model, 1,840 survey placements are distributed evenly throughout the environment.

Selecting environment support points is an important aspect of approximating Equation (6.4). The support points should be distributed wherever confident predictions are required. This might be uniformly across the environment or focused on a specific area. In this demonstration, 4,472 environment support points are distributed uniformly across the bathymetry as shown in Figure 6.13a.

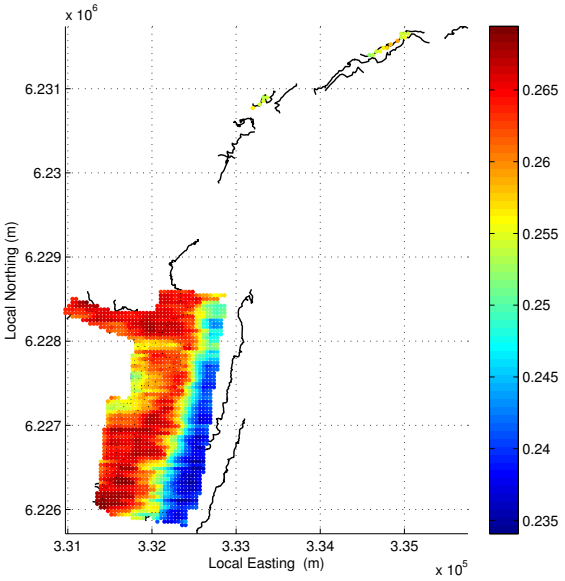
The predicted future mean variance for all 1,840 training survey placements is shown in Figure 6.13b. The training data is used to train a utility model which produces the prediction shown in Figure 6.13c. Each of the 814,213 pixels represents the predicted utility of a survey placement which allows at least 75% of the survey template to traverse valid bathymetry features. The predicted survey utility closely matches the training data, as shown in Figure 6.13.

Survey placements which produce a relatively high future variance are coloured yellow and red. For instance, surveys placements in the westerly region of the DEM traverse bathymetry similar to previous observations and are not expected to produce a large reduction in future variance. At these locations the survey template is traversing bathymetry features which are already highly correlated with observed values.

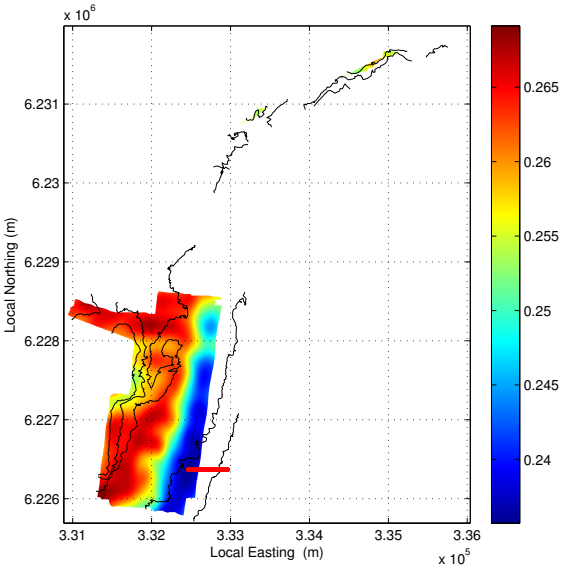
Survey placements which traverse confidently estimated bathymetry are not expected to produce a large reduction in future variance. Figure 6.10 shows that the main body of the bathymetry has already been predicted with a high degree of confidence. Sampling this area would not provide the model with any new information.



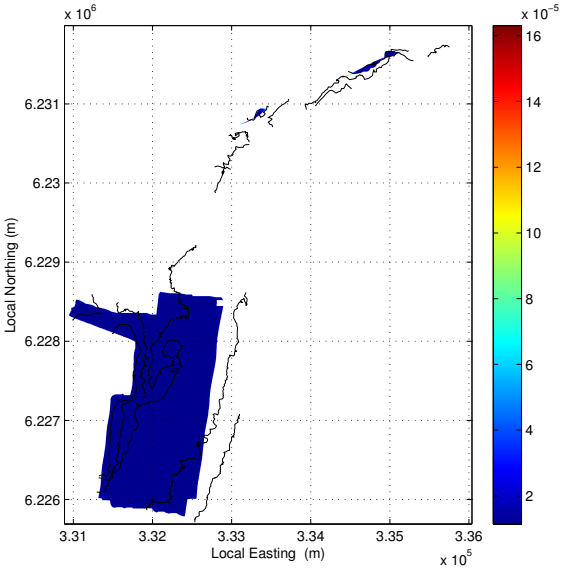
(a) Environment support points



(b) Training targets for the Gaussian process (GP) utility model



(c) Predicted survey utility



(d) Variance of the survey utility predictions

Figure 6.13 – Demonstration of planning with observations. **(a)** Environment support points used to approximate the mean variance of future habitat maps using Equation (6.6). **(b)** Training data used to train a Gaussian process (GP) utility model. Each dot represents a location where the mean future variance was calculated using the survey template. **(c)** Predicted survey utility using the GP utility model. The red line indicates the optimal survey location. **(d)** Variance of the survey utility predictions. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

Regions shown in blue are survey placements which result in a low future variance. These placements are occurring in the deepest and most Easterly region of the DEM where no observations have been made. The optimal survey placement, as predicted by the survey utility model, occurs in this region and is shown in Figure 6.13c. The optimal survey scores a mean future variance of 0.2357 and is expected to maximally reduce the variance of predictions made by the future habitat model.

Figure 6.13d shows that the survey utility has been predicted with a uniformly low-variance. This indicates that 1,840 training placements are sufficient to confidently describe the space of possible survey placements. Given the high confidence, it might be preferable to trade some accuracy in the predicted utility for faster planning times. This could be achieved by using less training placements.

The mean variance of the habitat model *before* performing the recommended survey is shown in Figure 6.14. This figure shows the actual variance of the habitat model given all of the observed data. The *expected* mean variance of the habitat model *after* performing the recommended survey is shown in Figure 6.15. Note that the proposed survey has not been performed. The depicted variance is a theoretical value.

Comparing Figure 6.14 and Figure 6.15 shows that performing the proposed survey is likely to cause a large reduction in variance. Although only a small number of observations would be gathered from the southeast, they are predicted to cause a large reduction in variance in the surrounding region.

The A-optimal objective function produces subtle behaviour. It does not simply reward gathering observations where the variance is the highest. Rather, it rewards gathering high-variance observations from regions of the environment where the data is highly correlated. A few observations in these locations will lead to a large decrease in the predicted variance across large portions of the environment. Conversely, observations gathered in high-variance areas of the environment where data is poorly correlated will lead to small local reductions in the predicted variance. This observation suggests that the region the recommended survey is traversing contains highly correlated data. Indeed the bathymetry features in this area are relatively uniform (Figures 6.3 and 6.4).

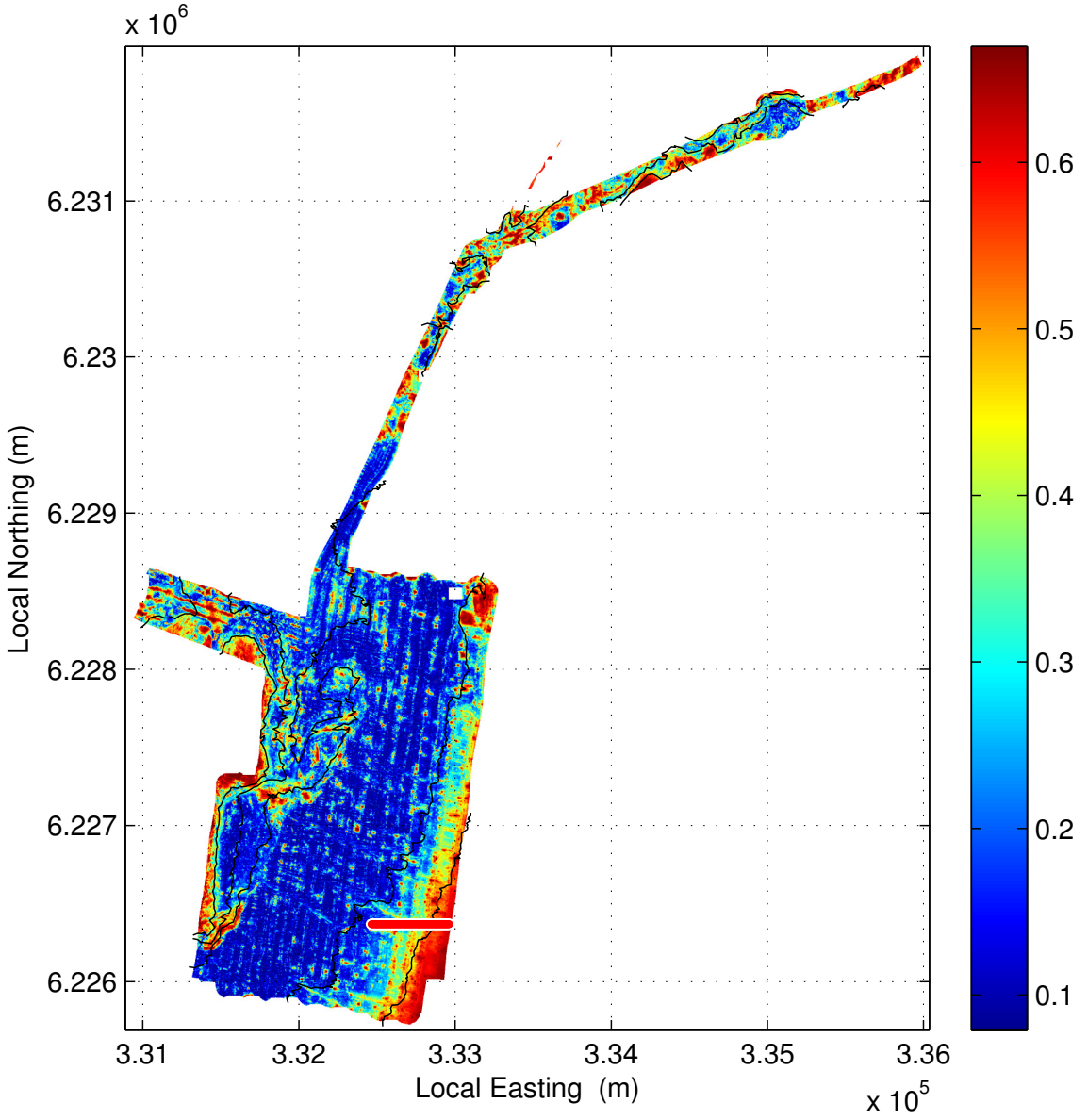


Figure 6.14 – Variance of the Bate Bay habitat map ***before*** performing the proposed $500m$ West-East survey survey (Figure 6.13c). This figure is identical to Figure 6.10. The mean variance is 0.2730. It is reproduced to facilitate comparison with Figure 6.15. The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. The red line depicts the proposed West-East survey. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

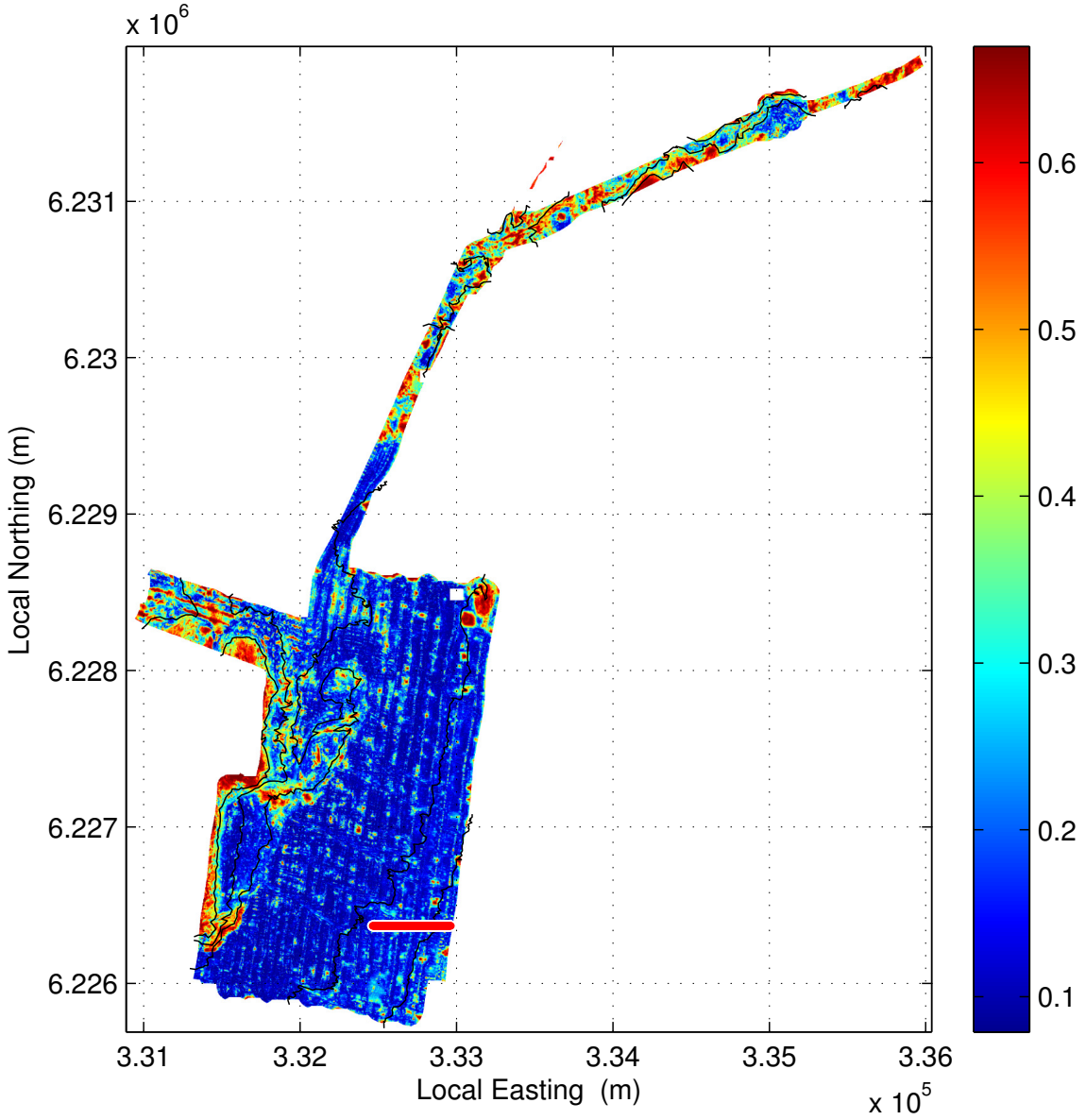


Figure 6.15 – Expected variance of the Bate Bay habitat map ***after*** performing the proposed $500m$ West-East survey (Figure 6.13c). The mean variance is 0.2372. The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. The red line depicts the proposed West-East survey. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

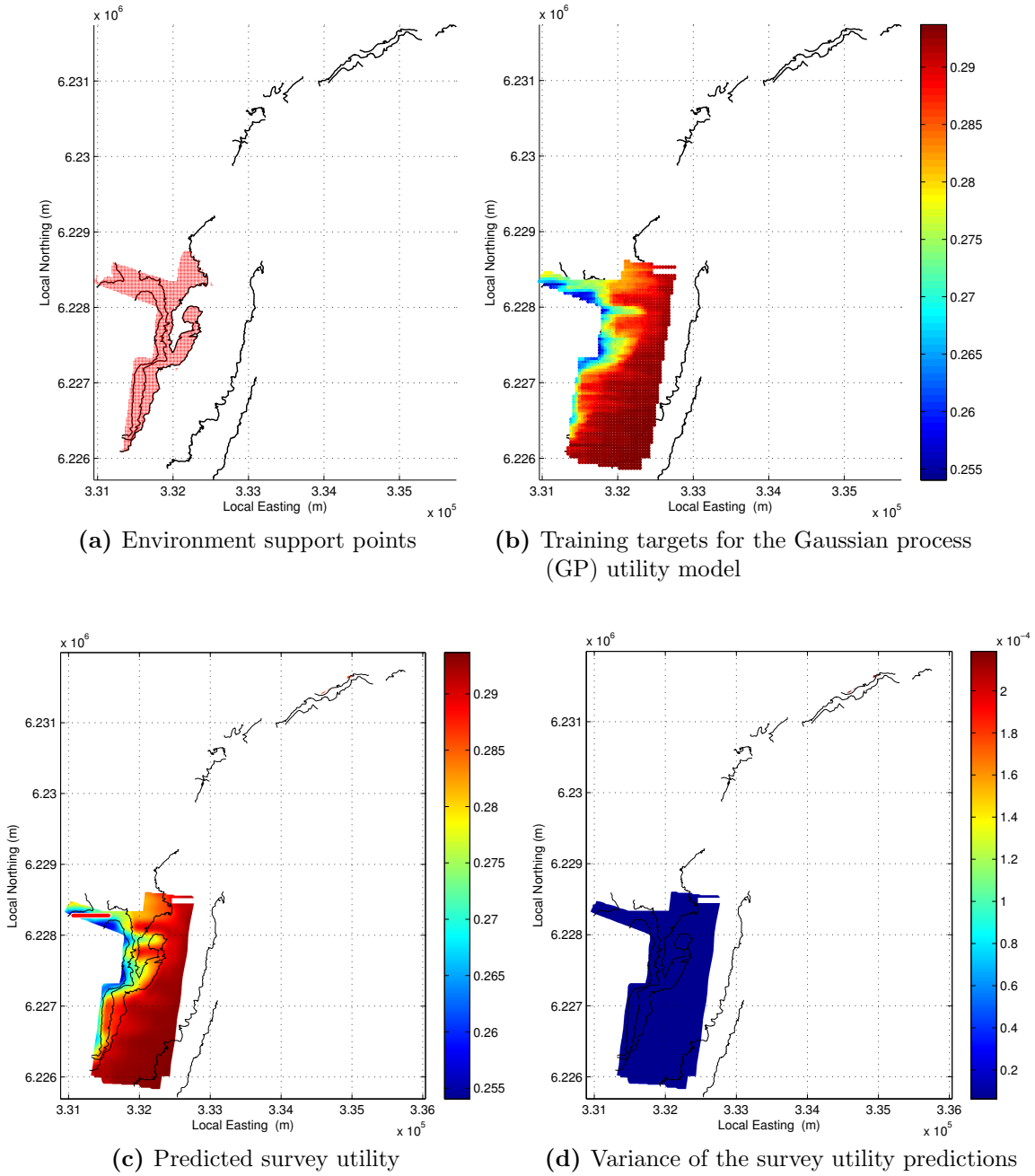
6.4.1 Directed Exploration

Survey placements recommended by the planning method are expected to maximally reduce the total variance of the future habitat map. This may create tension between the requirements of scientists and the behaviour produced by the A-optimal utility function. For example, the recommended survey placement shown in Figure 6.13, is likely to only gather observations of *Sand* which may have little science value.

The domain knowledge of scientists can be incorporated into the planning procedure by carefully choosing the locations of the environmental support points. This allows expert knowledge, or the imperatives of the research, to be expressed during planning. Suppose the objective of the mission was to collect informative observations of *Rock* and *Kelp*. This objective might be expressed by restricting the environmental support points to depths shallower than $25m$ in the main body of the bathymetry as shown in Figure 6.16a. During planning, the A-optimal utility function will not consider the effect a candidate survey will have on areas of the environment beyond these points.

Planning was performed using this strategy. Like the previous example, 1,840 training placements of the $500m$ West-East transect were used to train the survey utility function. The only difference is the distribution of the environment support points. This can be observed by comparing Figure 6.13a and Figure 6.16a. The survey utility function is shown in Figure 6.16c. The variance of the habitat map before the recommended survey is shown in Figure 6.17 and the expected variance after the recommended survey is shown in Figure 6.18.

The recommended survey has been placed in the most westerly region of the bathymetry. Unlike the recommended survey of the southeast (Figure 6.13c), the recommended survey in the west (Figure 6.16c) is only expected to cause a small decrease in the total variance of the habitat map. This is due to the composition of the environment. The terrain covered by the western survey is rugged and complex. Only a small local region around the survey is correlated with this variable terrain. As a result the western survey results in a smaller, localised change in variance (compare Figures 6.17 and 6.18) than the southeastern survey (compare Figures 6.14 and 6.15).



(a) Environment support points

(b) Training targets for the Gaussian process (GP) utility model

(c) Predicted survey utility

(d) Variance of the survey utility predictions

Figure 6.16 – Demonstration of directed exploration for West-East survey. **(a)** Environment support points used to approximate the mean variance of future habitat maps using Equation (6.6). The locations are restricted to the main body of the bathymetry at depths shallower than $25m$. **(b)** Training data used to train a Gaussian process (GP) utility model. Each dot represents a location where the mean future variance was calculated using the survey template. **(c)** Predicted survey utility using the GP utility model. The red line indicates the optimal survey location. **(d)** Variance of the survey utility predictions. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

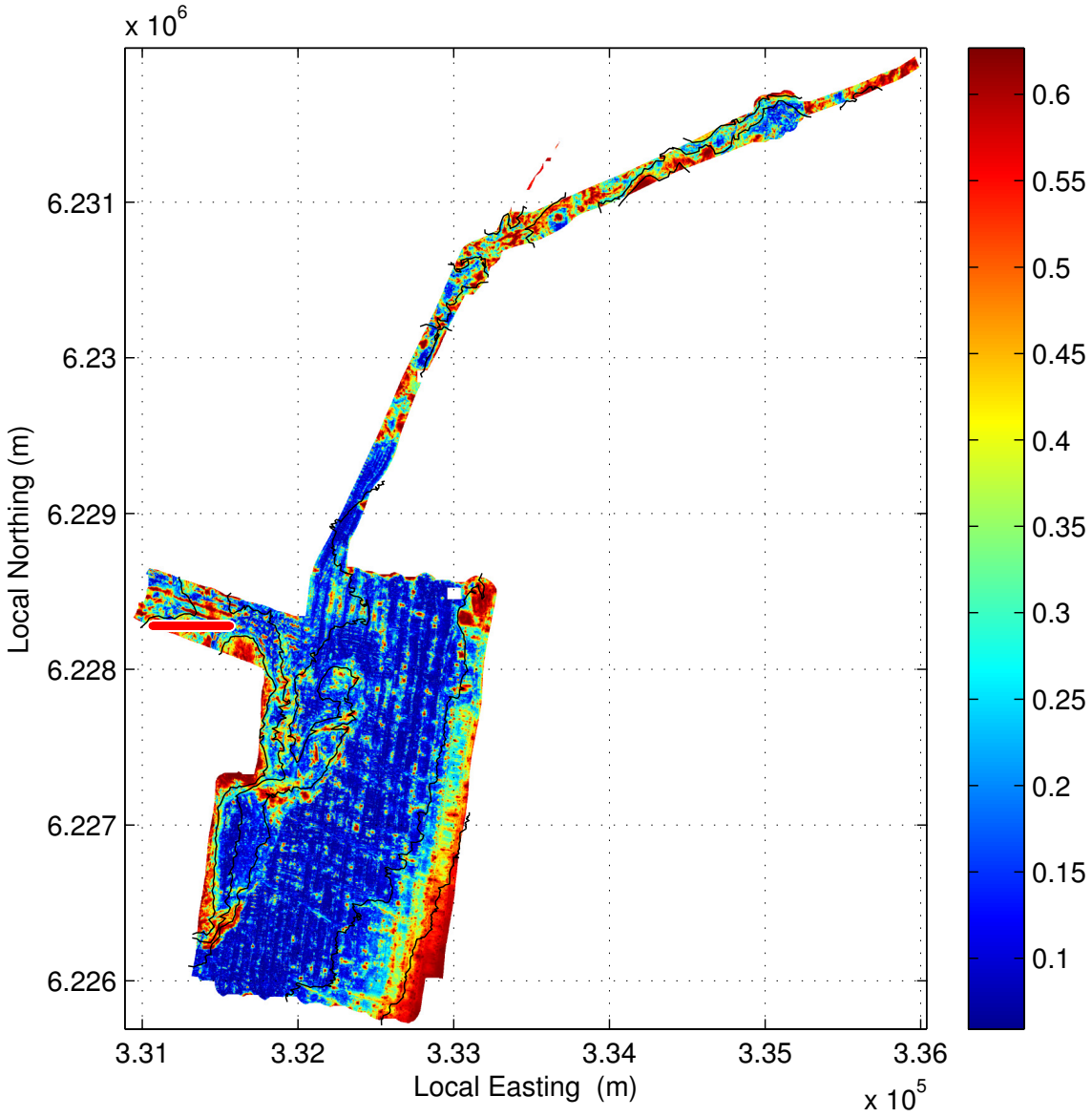


Figure 6.17 – Variance of the Bate Bay habitat map ***before*** performing the proposed $500m$ West-East survey (Figure 6.16c). This figure is identical to Figure 6.10. The mean variance is 0.2730. It is reproduced to facilitate comparison with Figure 6.18. The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. The red line depicts the proposed West-East survey. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

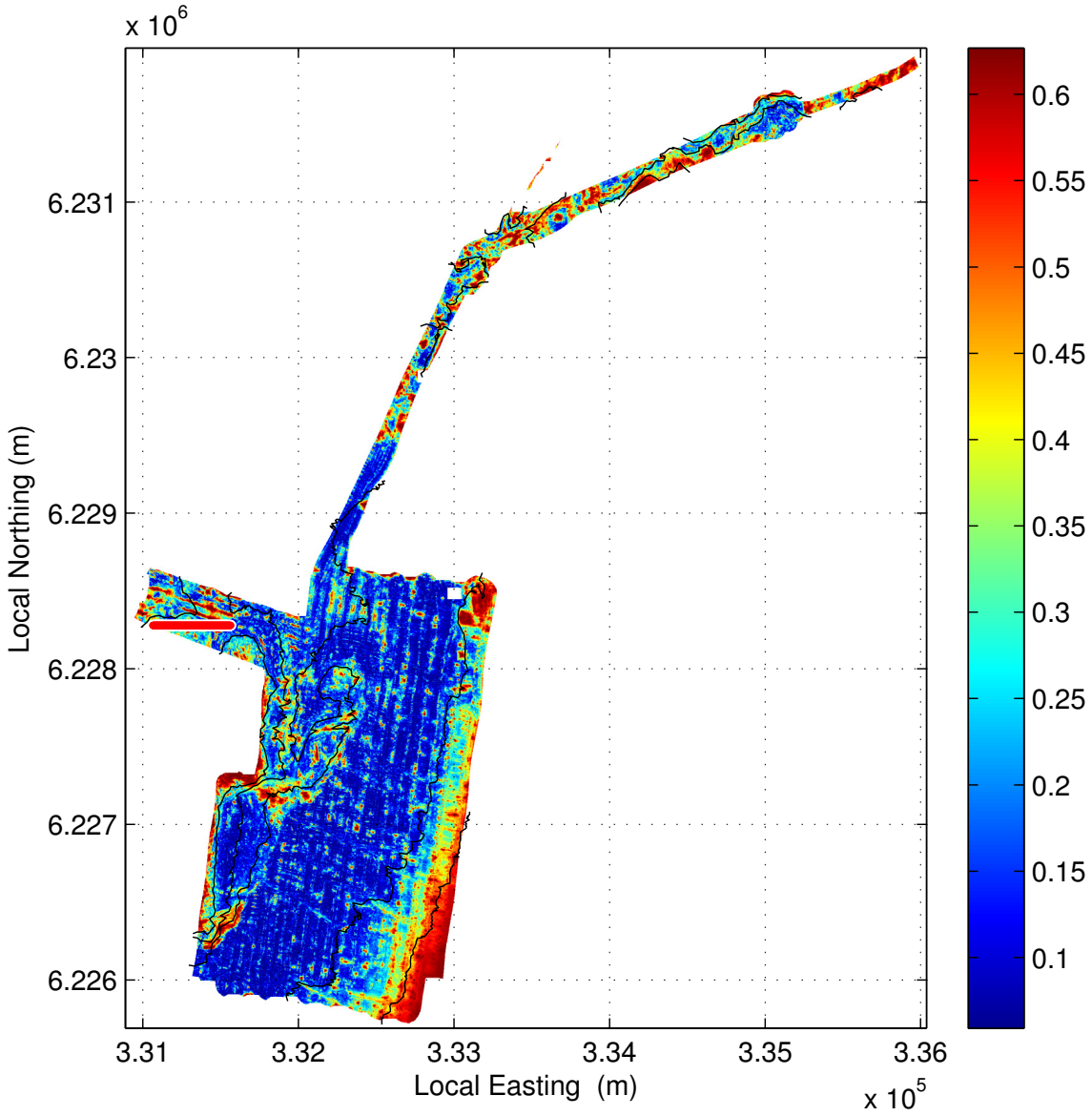


Figure 6.18 – Expected variance of the Bate Bay habitat map ***after*** performing the proposed $500m$ West-East survey (Figure 6.16c). The mean variance is 0.2589. The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. The red line depicts the proposed West-East survey. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

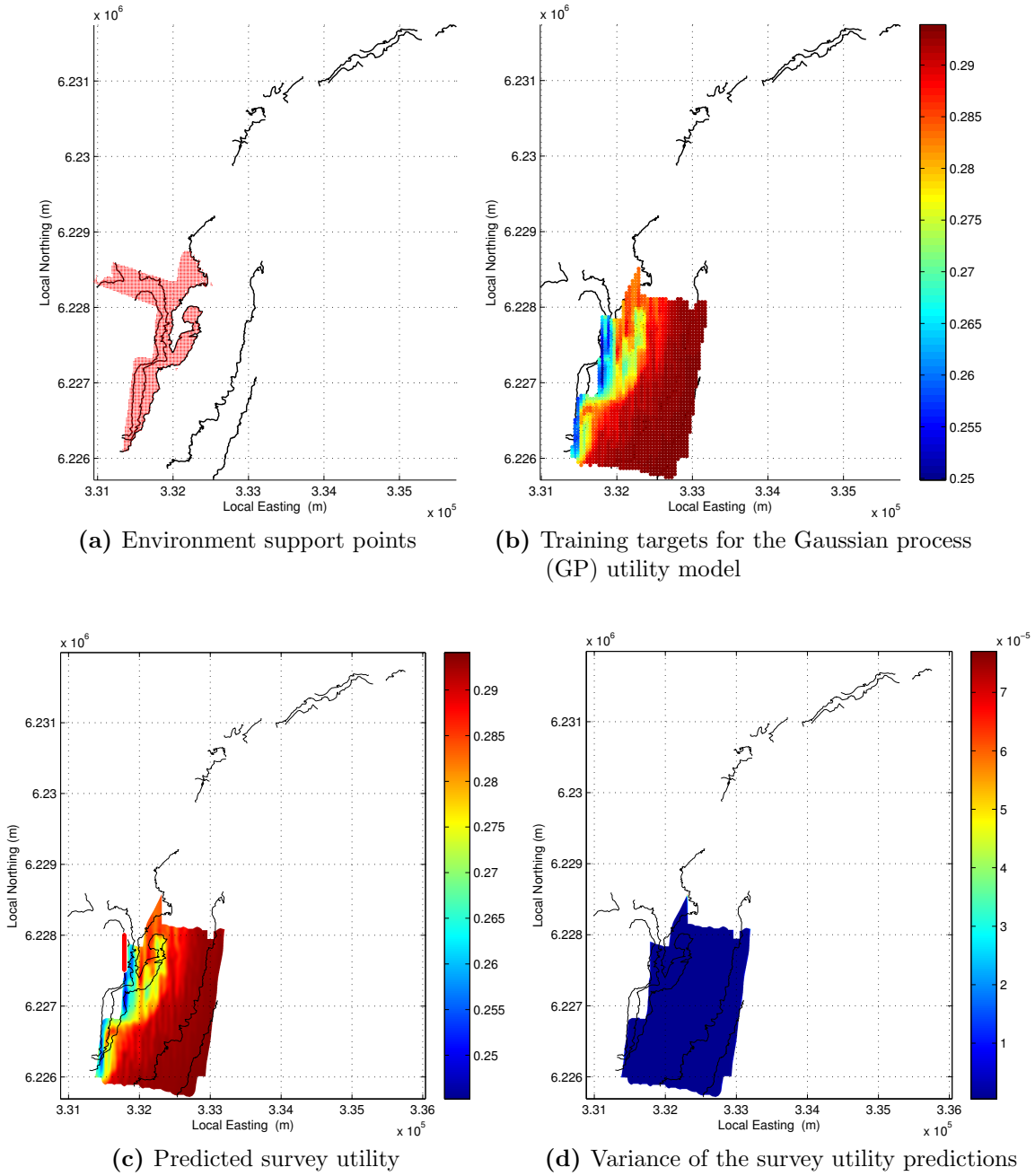
6.4.2 Experimental Results

Two additional dives of the Bate Bay survey area were performed in March 2013 including the proposed *West-East* $500m$ transect shown in Figure 6.16c. The second survey was a *South-North* $500m$ transect planned under similar circumstances to the *West-East* survey. The predicted survey utility and the recommended location of the *South-North* survey are shown in Figure 6.19. The expected mean variance of the habitat model *after* performing the *South-North* survey is shown in Figure 6.21.

The challenges of navigating in a dynamic ocean environment make it difficult to execute the planned surveys exactly. The survey trajectories that were actually realised are shown in Figure 6.22. The planning process was reperformed using these trajectories. The intention is not to plan where these marginally different surveys should have been placed. Rather, the objective is to quantitatively evaluate how well the planning method predicts the utility of the newly acquired data, independently of navigation error.

Quantitative measures of the survey trajectories realised during the 2013 deployments are compared in Table 6.3. The variance for each survey, predicted using the proposed planning method, is included in the second column. The variance predicted by the planning method and adjusted for navigation error is shown in the third column. The theoretical variance, prior to performing the survey, is included in the fourth column. This value is calculated by evaluating Equation (6.6) at *all* locations shallower than $25m$. The true variance is shown in the last column. The true variance is calculated by performing the recommended surveys, updating the habitat map and calculating the variance at *all* locations shallower than $25m$.

The second column of Table 6.3 shows the habitat map variance, in the shallow region, that is expected to occur as a consequence of performing the recommended surveys. The *South-North* survey is predicted to produce a lower habitat map variance and is the preferred survey. This result indicates that *South-North* is traversing data which generalises to more of the environment than *West-East*.



(a) Environment support points

(b) Training targets for the Gaussian process (GP) utility model

(c) Predicted survey utility

(d) Variance of the survey utility predictions

Figure 6.19 – Demonstration of directed exploration for South-North survey. **(a)** Environment support points used to approximate the mean variance of future habitat maps using Equation (6.6). The locations are restricted to the main body of the bathymetry at depths shallower than $25m$. **(b)** Training data used to train a Gaussian process (GP) utility model. Each dot represents a location where the mean future variance was calculated using the survey template. **(c)** Predicted survey utility using the GP utility model. The red line indicates the optimal survey location. **(d)** Variance of the survey utility predictions. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

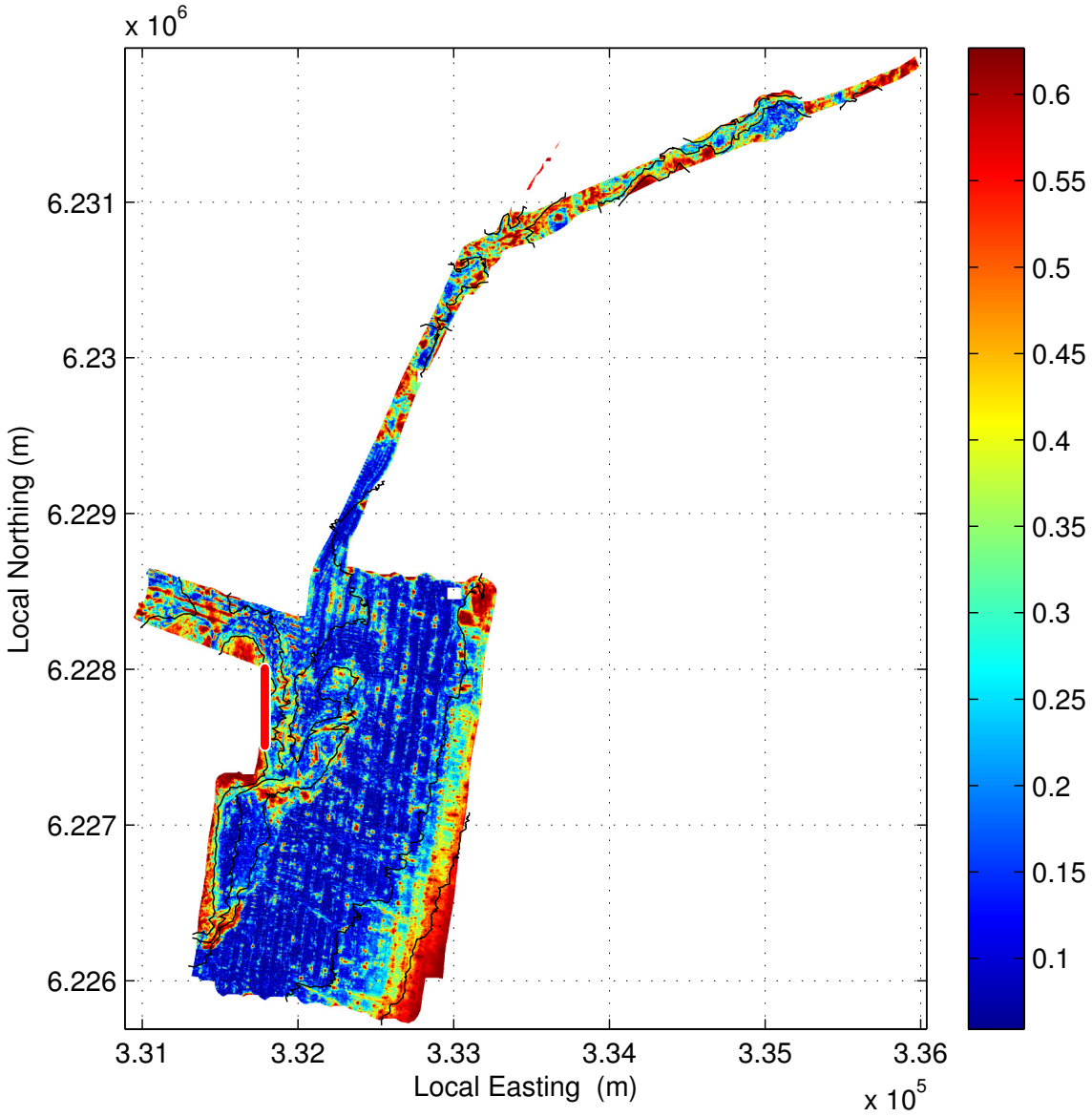


Figure 6.20 – Variance of the Bate Bay habitat map ***before*** performing the proposed $500m$ South-North survey (Figure 6.19c). This figure is identical to Figure 6.10. The mean variance is 0.2730. It is reproduced to facilitate comparison with Figure 6.21. The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. The red line depicts the proposed West-East survey. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

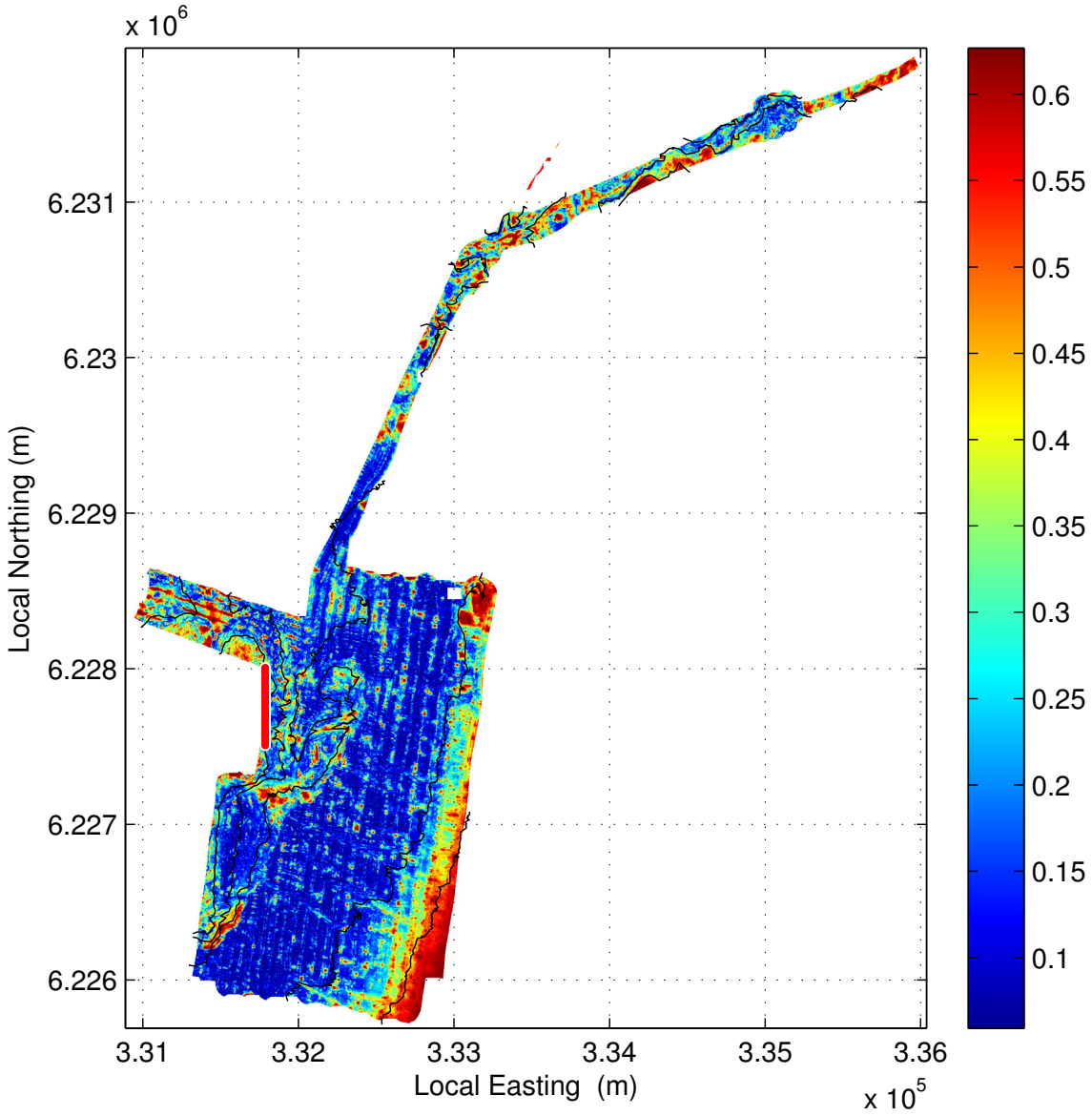
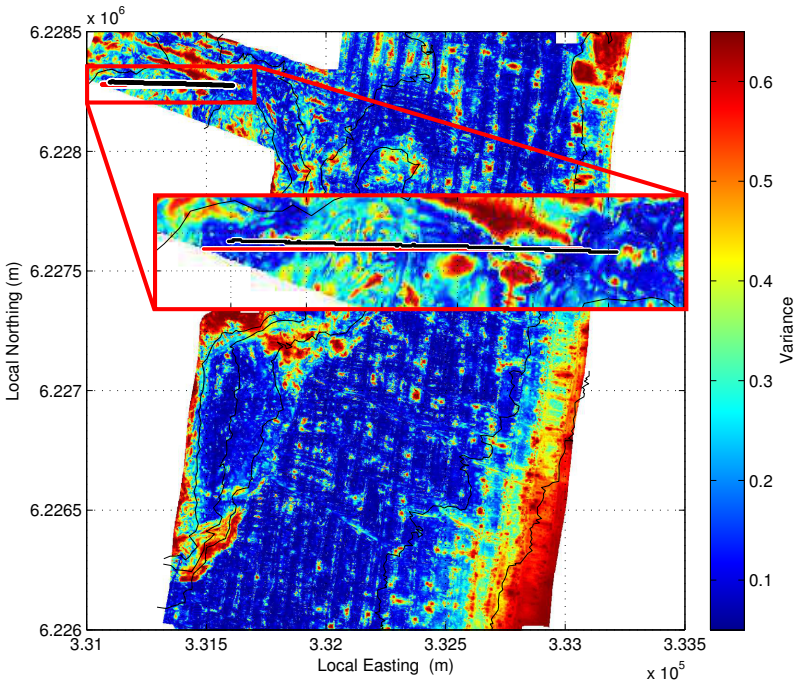
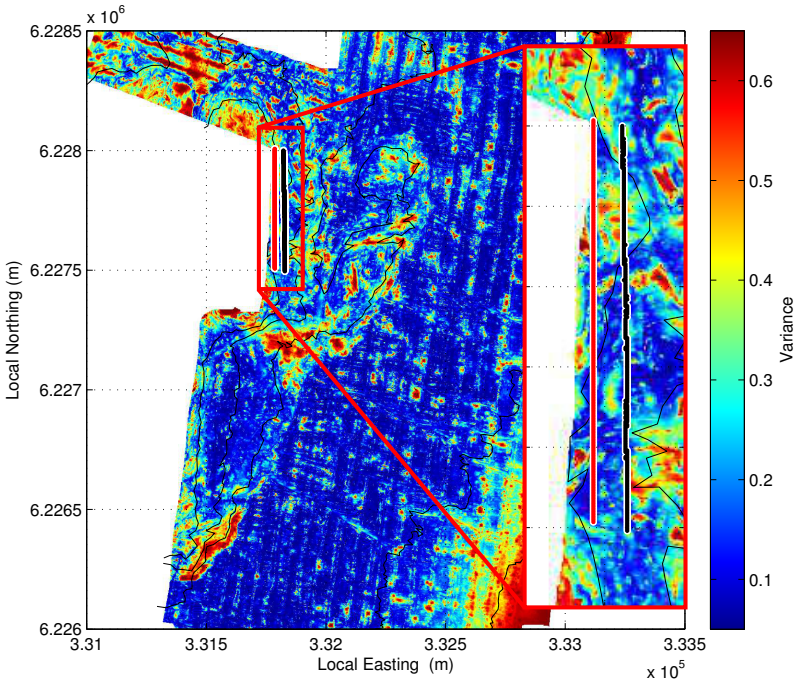


Figure 6.21 – Expected variance of the Bate Bay habitat map ***after*** performing the proposed $500m$ South-North survey (Figure 6.19c). The mean variance is 0.2589. The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. The red line depicts the proposed West-East survey. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.



(a) West-East survey



(b) South-North survey

Figure 6.22 – Detail of *West-East* and *South-North* surveys. **(a)** Detail of *West-East* survey. **(b)** Detail of *South-North* survey. The intensity plots show the expected variance *after* performing the proposed surveys. These intensity plots are identical to Figures 6.18 and 6.21. The red survey trajectories depict the *proposed West-East* (Figure 6.16c) and *South-North* (Figure 6.19c) surveys. The black survey trajectories depict the survey trajectories which were *actually* performed. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

Survey	Planned Variance	Predicted Variance	Theoretical Variance	True Variance
West-East	0.2541	0.2583	0.2601	0.3672
South-North	0.2452	0.2533	0.2488	0.2738

Table 6.3 – Comparison of variance values for recommended Bate Bay surveys. All variance values were calculated in the main body of the bathymetry at depths shallower than $25m$. The estimated variance is the variance predicted by the planning framework described in Section 6.4.

The third column of Table 6.3 shows the predicted variance of the survey trajectories which were actually performed. Replanning using the marginally different survey templates controls for the effects of navigation error. *Sirius* was deployed above *West-East* and programmed to transit to *South-North*. Due to good targeting, the predicted variance of the planned *West-East* survey closely matches the predicted variance of the actual *West-East* survey trajectory. During transit to the *South-North* survey, *Sirius* accumulated navigation error. *South-North* was executed approximately $30m$ east of the planned location. This larger targeting error causes the *South-North* survey trajectory which was actually performed to suffer a greater predicted loss of information than the *West-East* survey which was actually performed. Despite a loss in information due to navigation error, the *South-North* survey still provides a small advantage over the *West-East* survey and remains the preferred survey.

The fourth column of Table 6.3 shows the theoretical variance of the survey templates which have been adjusted for navigation error. The variance values contained in this column are produced by calculating the exact variance at *all* locations in the shallow region. Using this method in a planning context is equivalent to performing a brute force search of all possible survey placements. The utility is evaluated explicitly without approximation using environment support points or the GP utility function. This value is included to provide an indication of error in the planning process approximation. Again the *South-North* survey is calculated to produce the most informative survey.

Comparing column three to column four of Table 6.3 shows that there is approximation error between the estimated and theoretical values. This error does not prevent the surveys from being ordered correctly but it does inflate the predicted difference in utility between the two surveys. The proposed planning method finds a solution to the planning problem using significantly less data than exhaustive evaluation. While this makes the problem tractable, it introduces some approximation error into the solution. The approximation error in this experiment is tolerable as it does not cause the rankings of the surveys to deviate significantly from their theoretical rankings.

The last column of Table 6.3 contains the true variance of the future habitat maps. The true variance is calculated by updating the 2012 habitat model with the data that was collected from each survey and calculating the variance at *all* locations in the shallow region. The new imagery is pooled with the old data and clustered using the same method as Section 6.3.3. The new habitat classes are shown in Figure 6.23. The spatial distribution of the new classes is shown in Figure 6.8. The updated habitat maps are shown in Figures 6.25 and 6.26. The variance of the updated habitat maps are shown in Figures 6.27 and 6.28. Comparing the true variance to the other predicted quantities, in Table 6.3, shows that the theoretical values have estimated the rank of the surveys correctly. Again there is approximation error between the predicted and true values. This error does not prevent the surveys from being ordered correctly but it does inflate the predicted difference in utility between the two surveys.

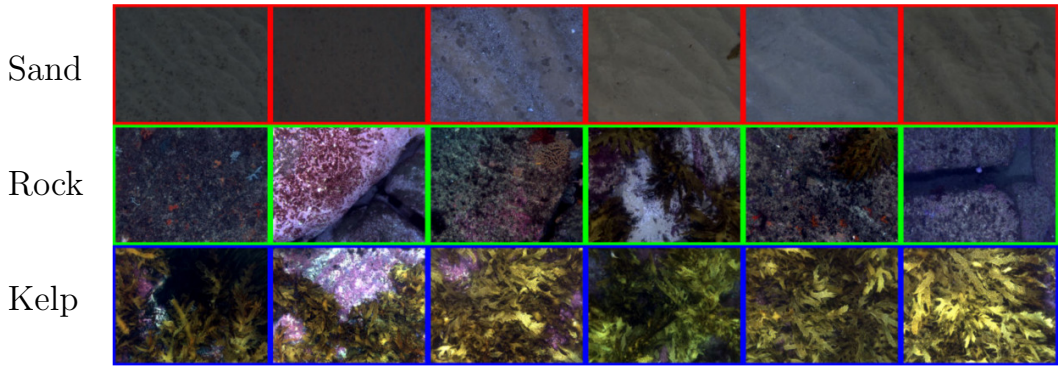


Figure 6.23 – New habitat classes after including imagery collected from the recommended *West-East* and *South-North* survey. The merged habitat clusters have retained the same semantic label.

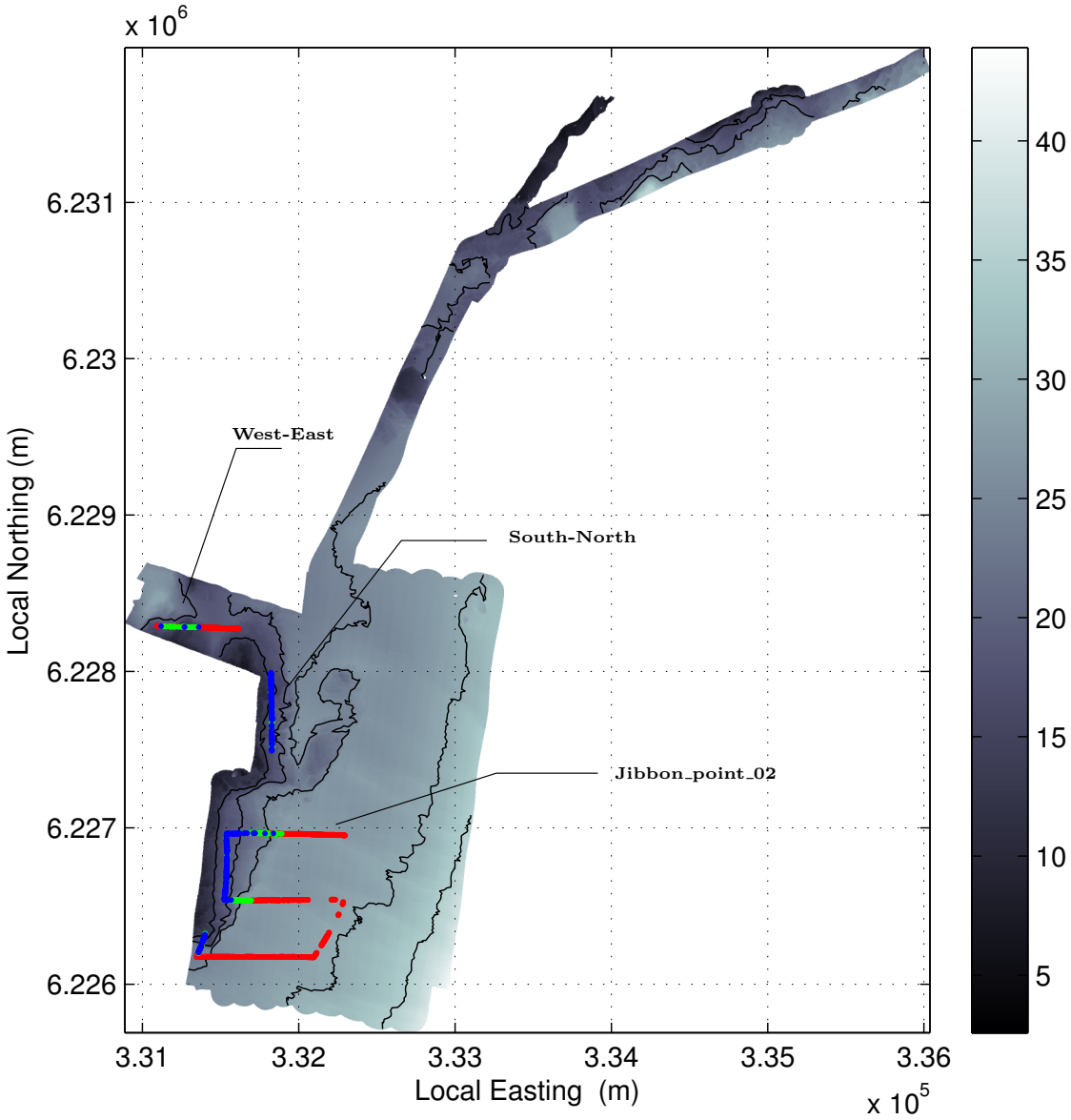


Figure 6.24 – Classified AUV surveys conducted in Bate Bay after performing the recommended *West-East* and *South-North* surveys. The intensity plot and black lines represent the bathymetry and depth contours. Each AUV pose is shown as a coloured dot where the colour indicates a specific habitat class. The colours for each habitat class are shown in Figure 6.23. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

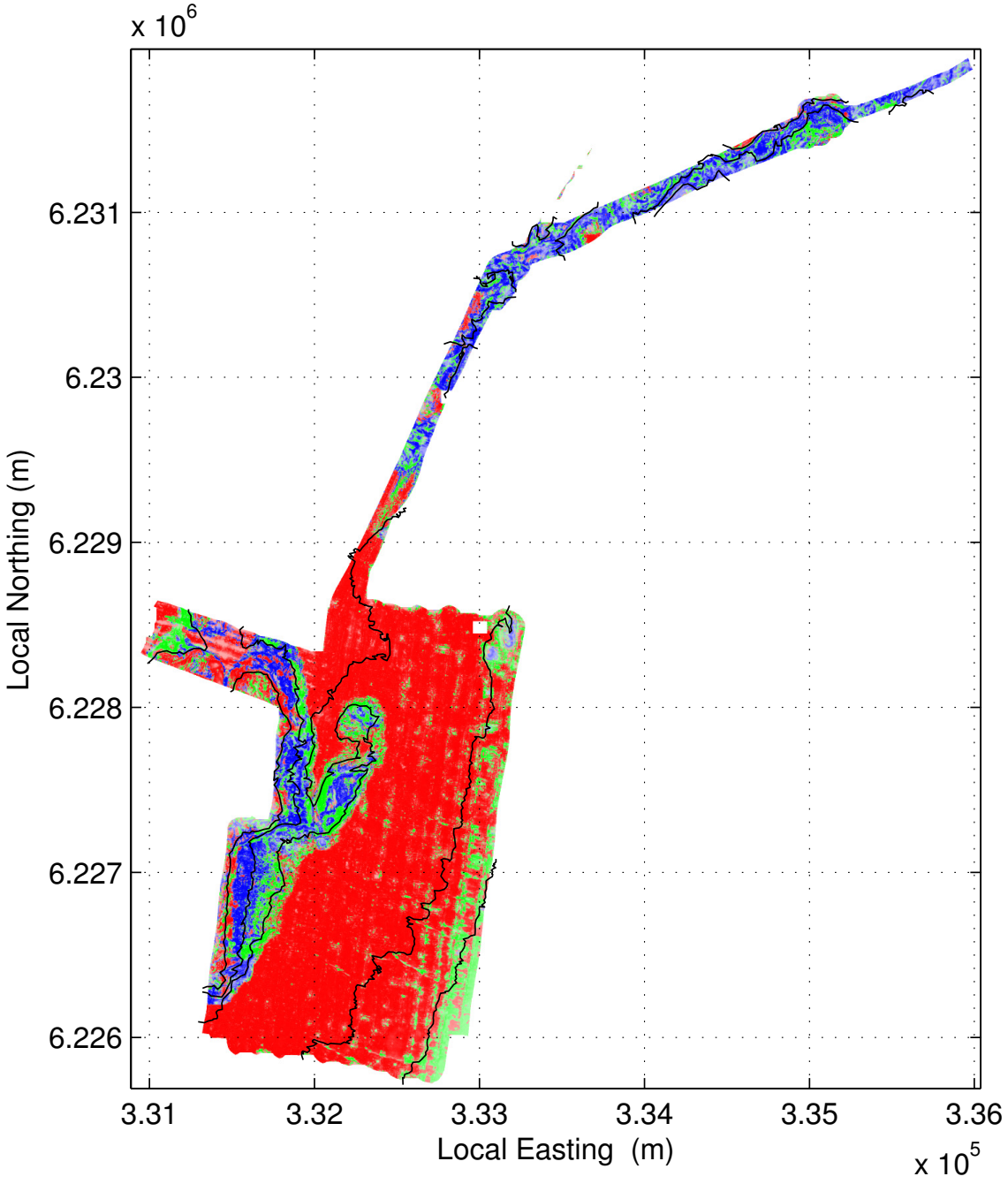


Figure 6.25 – Bate Bay habitat map updated with data observed by the West-East survey. The bathymetry contours are shown as black lines. The intensity plot shows the most likely habitat where the intensity of the colour is proportional to the probability of the most likely habitat. As the predictions become less certain, the colour fades to white. Colours correspond to the habitat classes shown in Figure 6.23. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

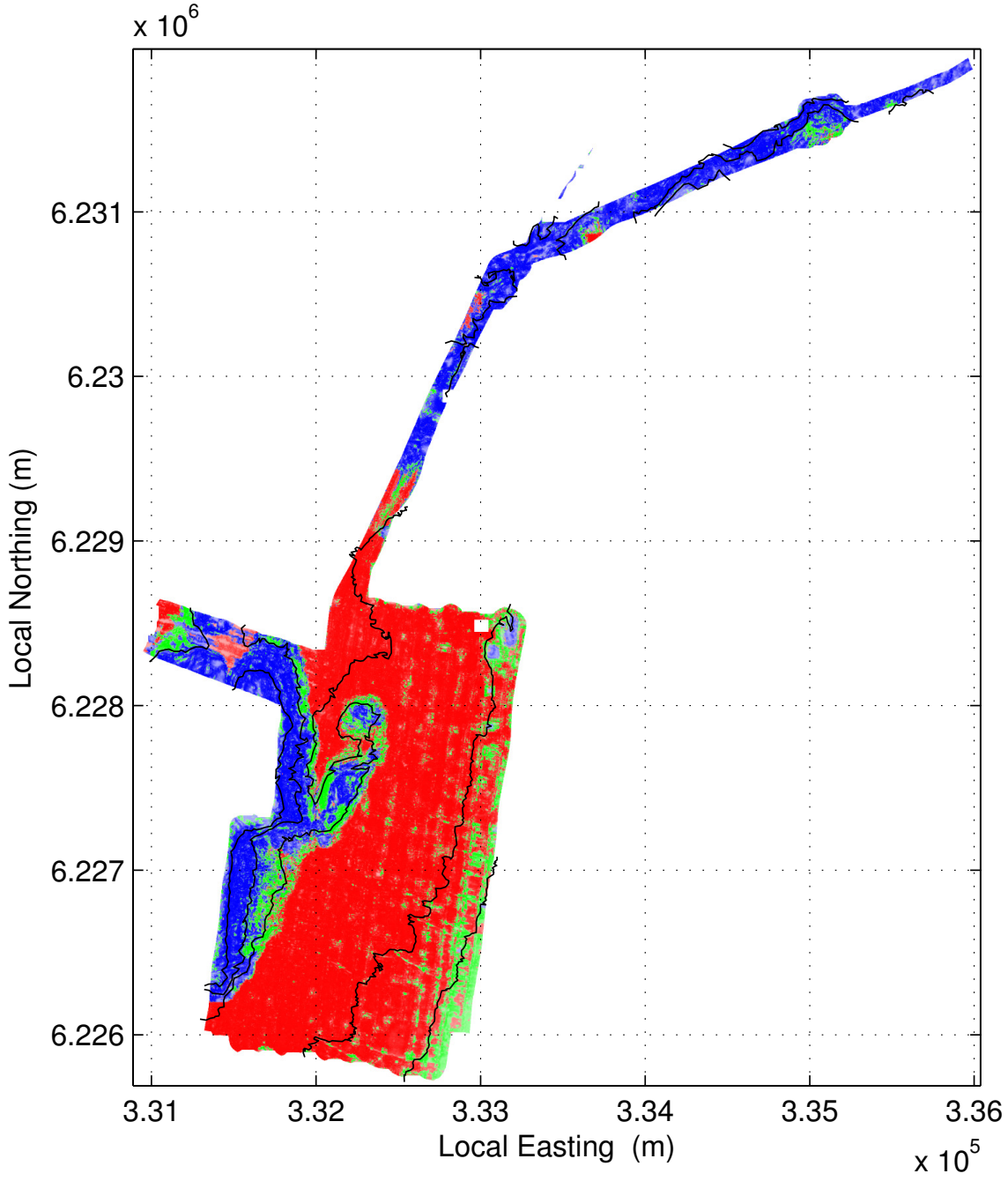


Figure 6.26 – Bate Bay habitat map updated with data observed by the South-North survey. The bathymetry contours are shown as black lines. The intensity plot shows the most likely habitat where the intensity of the colour is proportional to the probability of the most likely habitat. As the predictions become less certain, the colour fades to white. Colours correspond to the habitat classes shown in Figure 6.23. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

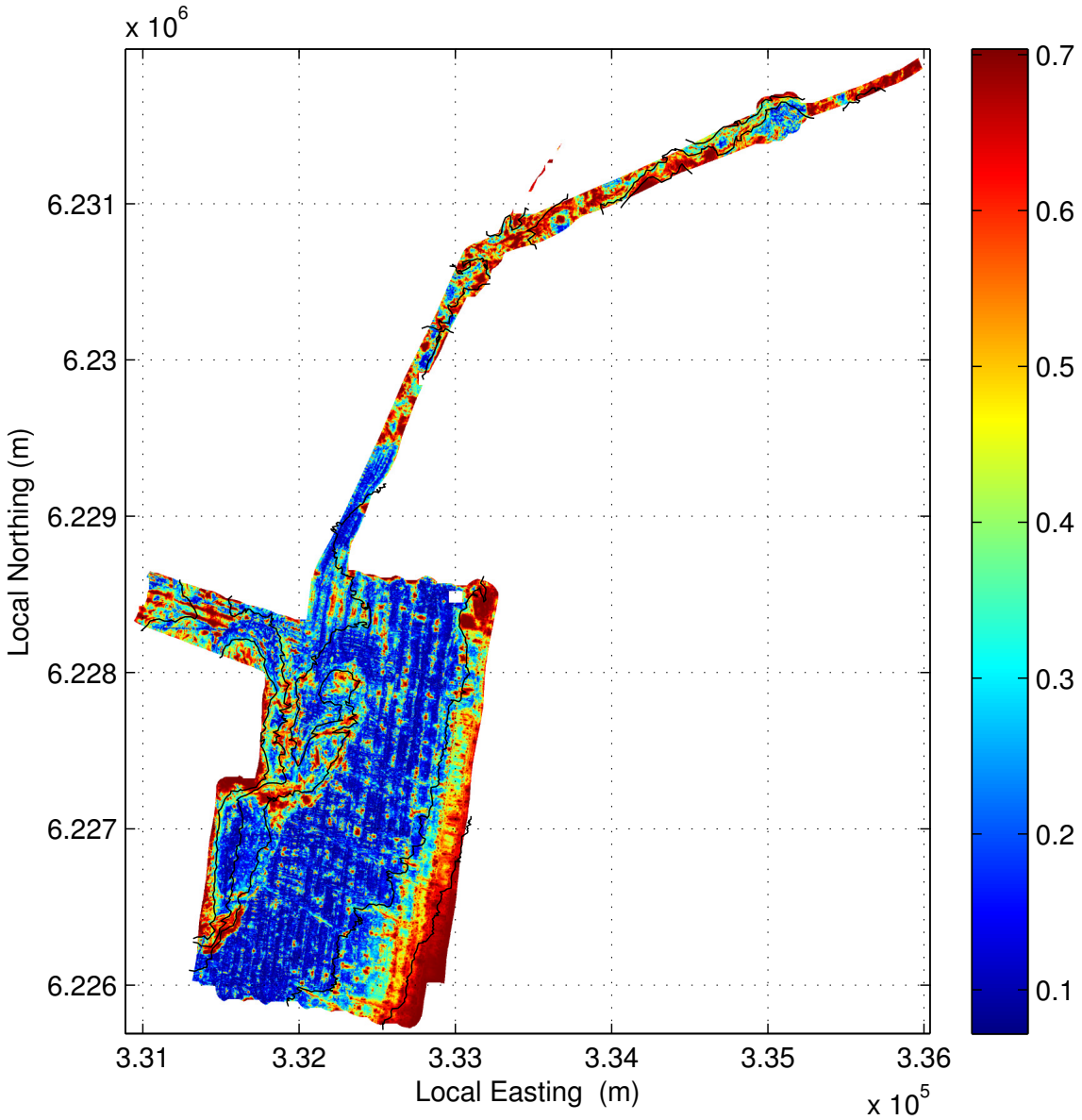


Figure 6.27 – Variance of Bate Bay habitat map updated with data observed by the West-East survey. The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

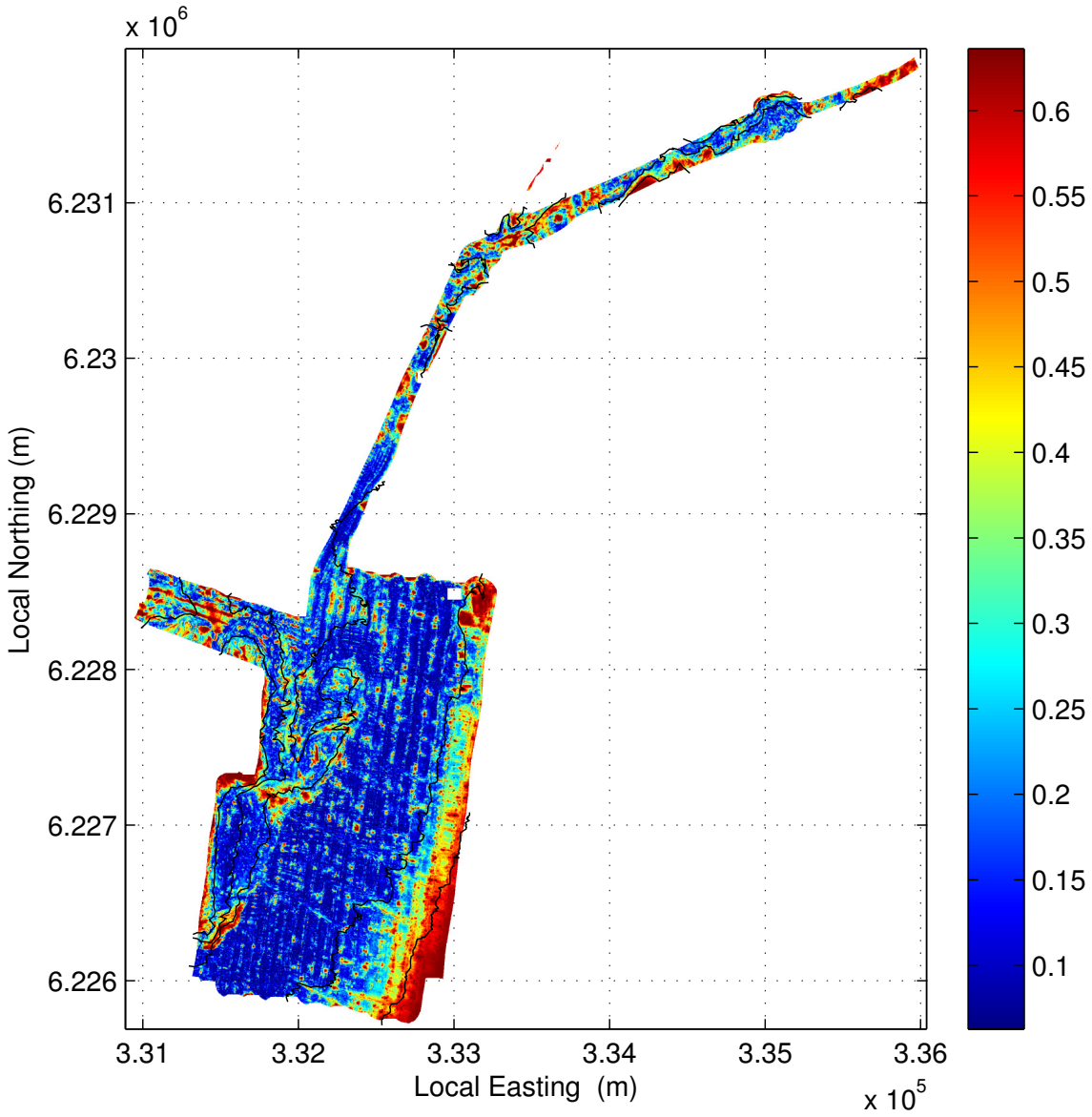


Figure 6.28 – Variance of Bate Bay habitat map updated with data observed by the South-North survey. The bathymetry contours are shown as black lines. The intensity plot shows the mean variance of the latent functions. Local coordinates are expressed in zone 56H of the universal transverse Mercator coordinate system.

The variance of the 2012 habitat model (Figure 6.10) in the shallow region is 0.2939. Interestingly, the variance of the *West-East* model has increased. This unexpected degradation of performance reveals a tacit assumption. The variance of the habitat map latent functions can be calculated at unobserved locations using Equation (2.8). Equation (2.8) is exact if the hyperparameters used in the GP are known to be true values. The utility function given in Equation (6.6) relies on Equation (2.8). If the hyperparameters of the GPs used in the habitat model do not change as a consequence of collecting new data, the utility equation is also exact. However, if the habitat model is retrained after data is acquired, the new hyperparameters may change to reflect a more ambiguous data set. In this scenario the habitat map may actually produce less certain predictions even though more data is available for training and inference.

After updating the habitat model with the observed data, the model parameters change to reflect more ambiguous data sets. Both surveys have been designed to target a confusing area of the environment. *West-East* observes rapid changes between *Rock*, *Kelp* and *Sand* causing the GP-based habitat map to learn shorter length scales and an increase in variance. *South-North* predominantly observes *Kelp* including some *Rock*, leading to less change in the length scales and a reduction in variance. Although both models are less certain than the predicted values, their ranking has been preserved and *South-North* is correctly identified as the more descriptive survey.

An increase in variance after updating the model parameters with the newly acquired data indicates that the covariance function used in the GP may not be flexible enough to model the data. In this thesis the squared exponential covariance function, Equation (2.9), is used. This covariance function assumes smoothness in the environment. This assumption is violated at the interface between classes such as *Rock* and *Sand*. The squared exponential function must choose parameters which make a compromise between modelling contiguous data and the step change between classes. Non-stationary covariance functions have an advantage as they can adapt to the data by varying their length scales locally. A covariance function which properly represents the shape of the space which the underlying phenomena resides in, is less likely to produce more uncertain habitat maps as new data is integrated into the model.

If previous models have been created with a sufficient amount of data, the model parameters are unlikely to change significantly with additional sampling and the rankings ought to remain accurate. If previous models have been created with a paucity of data, modelling error may generate unreliable rankings. The majority of predictions produced by the 2012 habitat model are low-variance. This indicates that there is sufficient data to model the environment and the rankings can be trusted.

6.5 Summary

This chapter extends the planning framework detailed in Chapter 5 to situations where prior observations of the environment exist. Surveys are ranked using an A-optimal utility function which rewards survey locations that reduce the expected variance of the habitat map over the entire environment. This objective function can be derived from information theoretic principles and has found practical use in the robotics literature.

The planning framework was used to plan two deployments. The results show that the proposed planning method can approximate the true A-optimal utility function well. In the surveys tested, the A-optimal utility function was able to correctly identify the survey which would produce the most confident habitat map.

The A-optimal utility function is theoretically correct when the hyperparameters of the GP are not changed as a result of including new data. This makes the A-optimal utility function most suitable for situations where the parameters of the habitat model are static or unlikely to change. For instance, models which have access to large amounts of prior data are unlikely to learn significantly different hyperparameters if candidate surveys only make up a small fraction of training data. In other scenarios the hyperparameters may change, causing the expected variance to deviate from theoretical values. The ranking of the surveys tested in this chapter were not affected by changing hyperparameters.

Chapter 7

Conclusion

This thesis attempts to address issues which arise when using robotic platforms to monitor natural environments. Two main obstacles are identified.

Developments in both hardware and software have extended the amount of time robotic platforms are able to operate in the environment. Improved sensor design and new sensing modalities offer a higher bandwidth of observations. The combination of increased endurance and higher fidelity sensors produces enormous data sets. Although these data sets are highly descriptive, the volume of data that must be processed is often too large for human experts to analyse exhaustively in a practical amount of time or in a cost-effective manner. This burden creates a bottleneck in the process of converting observations into scientifically relevant data.

Operational constraints such as finite battery capacities, data storage, computational resources and project budgets make it impractical to survey the environment exhaustively. A limitation on the number of surveys that can be performed creates a resource allocation problem. To use robotic platforms effectively, survey trajectories which maximise the amount of information gathered from the environment must be designed.

Solutions to these two problems are proposed in this thesis. The proposed solutions and contributions are summarised in Section 7.1. Section 7.2 concludes this thesis by raising possible avenues for future research.

7.1 Contributions

Three primary contributions are presented in this thesis: a new classifier designed to accept probabilistic training targets rather than discrete training targets; a semi-autonomous pipeline for creating benthic habitat maps; and an offline method for autonomously planning benthic surveys. These contributions are reviewed below.

7.1.1 Classification with Probabilistic Targets

A new Gaussian process (GP) based classifier, called the probabilistic targets least squares classifier (PTLSC), is proposed in Chapter 3. The PTLSC extends the probabilistic least squares classifier (PLSC) and is designed to accept probabilistic training data rather than discrete class assignments. This modification makes the new model more amenable to being used in conjunction with other machine learning algorithms. Results presented in Chapter 3 show that if a supervisory machine learning algorithm is able to produce probabilistic training data, the PTLSC will produce more accurate predictions than the PLSC. This makes the PTLSC an ideal candidate for participating in data processing pipelines.

7.1.2 Semi-Autonomous Habitat Mapping

The main application of this thesis is the creation of benthic habitat maps. Benthic habitat maps are created by correlating features extracted from a digital elevation model (DEM) collected using a ship-borne multibeam echo-sounder (MBES) system and in-situ observations collected by an autonomous underwater vehicle (AUV). However, the volume of data collected by AUVs is often too large for human experts to analyse exhaustively in a practical amount of time. Difficulty in classifying the AUV imagery into habitat classes typically prevents all of the AUV data from being included in habitat models. To help alleviate the burden placed on human experts, a semi-autonomous data processing pipeline for creating benthic habitat maps is proposed.

The proposed pipeline works by compressing the AUV imagery into a set of habitat clusters. This is done using an unsupervised variational Dirichlet process (VDP) model. The probabilistic data created by the VDP can then be verified and modified by a human operator. Once the VDP clusters are deemed acceptable, the probabilistic data is used to train a PTLSC. The pipeline is demonstrated on real marine data where it is able to generate a habitat map with minimal human intervention.

7.1.3 Autonomous Survey Planning

Collecting informative data is a relevant consideration when exploring large natural environments. Data collected during robotic surveys of the environment will determine the quality of models which rely on the observed data. Optimising the location of a robotic survey allows the most informative data to be collected. This thesis contributes an offline method for planning benthic surveys. The proposed framework is designed to select the optimal location to place a prespecified survey template. The planning framework can operate in large continuous spaces. This property allows the proposed method to scale up to very large, realistic planning problems. The method is demonstrated in a large marine environment.

7.1.4 Planning Utility Functions

The utility function used to guide exploration should properly reflect the objectives of a survey. For situations where there are no prior observations to exploit, a novel planning heuristic which relies on the Kullback-Leibler divergence (KLD) is proposed. This utility function promotes coverage of the bathymetry feature space and is shown to be an effective heuristic for choosing survey locations. For situations where prior observations exist, it is possible to exploit a habitat model. Under these circumstances an A-optimal utility function is recommended. This utility function rewards survey locations which reduce the expected variance of the habitat map.

7.2 Future Work

The deployment of robots into large-scale, natural environments is becoming more widely adopted. Although interest in the field has seen an increase in the body of literature addressing research problems, challenges still remain. The following section outlines a number of potential avenues for further research with a particular focus on bathymetric habitat mapping.

7.2.1 Feature Selection

It is widely acknowledged that multiple terrain descriptors and spatial scales contribute to accurate bathymetric habitat maps [101]. However there is a gap in the literature detailing how to select appropriate terrain descriptors and spatial scales. A large number of articles published in the field use principal component analysis (PCA) to compress multiple features and scales into a manageable number of dimensions. This approach side-steps the issue of intelligently selecting appropriate features.

Feature learning techniques borrowed from the image processing community may prove to be a fruitful area of research. These techniques could be applied to terrain descriptors or to raw bathymetry patches. Although terrain descriptors capture variation in the environment, the derivation of some terrain features is rather arbitrary. By applying feature learning techniques directly to bathymetry patches it may be possible to implicitly learn terrain features from the environment.

7.2.2 Covariance Functions

The importance of properly selecting a suitable covariance function when using a GP model is often overlooked. The covariance function encodes assumptions that have been made about the space that the underlying function resides in. A poorly chosen covariance function will improperly model correlations in the data.

In this thesis a squared exponential covariance function was used to model the environment. Although this simple covariance function is able to produce reasonable results, it is limited by smoothness and stationary assumptions. In a stationary process, the correlation between data points is assumed to be independent of their location in the input space. In a non-stationary process, the correlation between data points can be modelled as a function of the location in the input space. Non-stationary covariance functions relax the smoothness and stationary assumptions.

In natural environments some process may occur smoothly while other process will exhibit discontinuities. For example, in benthic habitats the interface between rocks and other classes, such as sand, is often a discrete change which violates the smoothness assumption. By using locally varying length scales, non-stationary covariance function are better suited to modelling discrete changes in the data.

Research into covariance functions for benthic habitat mapping will allow correlations within the environment to be modelled more accurately. This is an important considerations for exploration methods which rely on accurate estimates of uncertainty.

7.2.3 Environmental Models

Environments are modelled in this thesis using a data-driven approach. This approach simply finds and models patterns within the data. The approach makes no attempt to understand the processes which create the data. Disciplines within the environmental science community focus on these science questions and produce complex models of the environment. Leveraging the domain knowledge imparted by these models and methods is likely to lead to richer and more accurate models of the environment than relying on observed data alone. In return the robotics and machine learning communities can offer methods for collecting and processing data. The objective is to free scientists from mundane data processing allowing them to focus on interesting science questions.

7.2.4 Exploration Objective Functions

Objective functions for exploration are investigated in this thesis. The KLD utility function proposed in Section 5.3 is a good heuristic for exploration when no environmental model exists. The exact relationship between survey KLD and habitat map accuracy remains an open question. Further research into what the KLD utility function might reveal about the usefulness of a proposed set of data would provide a more definitive answer to how the utility function should be interpreted.

The A-optimal utility function proposed in Section 6.1 designs surveys which are expected to reduce the average uncertainty of a habitat model. This is a general utility function and does not optimise any specific scientific goal. For example, specific exploration tasks like reducing the uncertainty of predictions of a particular species or mapping its distribution cannot be solved using the planning method presented in Chapter 6. Research into how methods such as Bayesian optimisation [68] or adaptive methods [93] can be used to target a specific species within the environment or adapt behaviour online will permit more complex exploration strategies.

Section 6.4 revealed the impact control uncertainty can have on utility. Planning frameworks such as Markov decision processes (MDPs) [95] can propagate control uncertainty during planning. The same idea can be integrated into the planning method proposed in Section 5.1. The survey utility model could be trained using expected utility. Rather than evaluating the survey utility at a training location assuming certainty, the expected utility at a training location could be calculated by integrating over the distribution of possible survey trajectories which might actually occur. The challenge is not that this would add computational complexity to the planning process. The challenge is that designing a useful uncertainty model can be difficult. This is particularly true of marine environments where AUVs are perturbed by the dynamics of the water around them. In this thesis data was gathered from relatively shallow coastal locations where water currents are complex and unpredictable systems. Water current data in these regions is not often available at the precision and scale that is required for planning. Without this data it is difficult to create a realistic uncertainty model.

Another aspect of planning which was overlooked in this thesis is planning under physical costs. Different survey designs will consume different amounts of resources whether the cost is measured in time, energy, money or operational risk. This raises the question of how to efficiently allocate operational costs towards information gathering.

This is a nontrivial task as information gain and resource consumption occur in two competing modalities. Longer surveys will gather more information from the environment but cost more to perform. Conversely, shorter surveys will gather less information but will cost less to perform. Nominating how many units of operational resources are allocated towards gathering a unit of information is a decision robot operators need to make depending on the objectives of the deployment and availability of resources. This approach is adopted in methods [58] where the relative important of various and possibly conflicting objective functions are simply aggregated as a weighted sum.

Normalising the information gained from a particular action or experiment by its operational cost provides a measure of efficiency. For instance, competing marine surveys could be evaluated in terms of information gained per joule of energy or metre travelled. The survey with the higher figure would provide the most efficient use of resources. Although this allows a natural trade-off between information and operational costs, the objective function becomes much more complex.

In conventional shortest path problems, techniques such as dynamic programming can be used to solve the problem in polynomial time [48]. These methods solve problems by breaking them into subproblems which are solved independently and then combined to produce a global solution. This is only possible if the costs are additive. Normalising the information gained from a sequence of actions by their operational cost violates this useful property. Since this objective function depends on a sequence of actions, the problem cannot be easily broken into subproblems. Designing efficient algorithms which operate using this type of objective function becomes a much more difficult task.

7.2.5 Survey Design

The autonomous survey planning algorithm presented in this thesis determines the optimal location of a prespecified survey template. Whilst the proposed method allows the utility of multiple survey templates to be compared, it is unable to design survey templates. A planning framework which relaxes the assumption of a fixed survey template would provide a higher degree of autonomy. The ultimate goal is to relax all restrictions on the shape of a survey. A system with no restrictions would be free to design its own survey trajectories. This is a nontrivial task as providing this level of freedom greatly increases the complexity of the planning problem.

A tractable extension to the planning algorithm would allow survey templates with free variables, which govern the shape of the templates, to be used. The planning algorithm would have to learn how the placement and survey parameters affected survey utility. Such parameters might include survey orientation, length, width and track spacing. Even this small increase in the number of dimensions greatly increases the complexity of the problem due to the curse of dimensionality.

7.2.6 Sequential Design and Optimal Stopping

Scientific missions often involve a sequence of surveys. Extending research into methods for planning a sequence of surveys would aid in the management of autonomous missions. The planning algorithm proposed in this thesis could be adapted to greedily plan a sequence of dives. A more intelligent system may be able to plan a sequence of surveys over a small planning horizon. Again this would greatly increase the complexity of the planning problem as decisions made early in the sequence will affect subsequent decisions.

A related task is to determine when to stop sampling the environment. At some point the information gained from further surveys would not be sufficient to justify the use of more resources. A method for optimal stopping might also provide an indication of how many surveys are needed to efficiently model an environment.

Bibliography

[1] M. Andriluka, L. Weizsäcker, and T. Hofmann. Multi-class classification with dependent Gaussian processes. In *Proceedings of "12th International Conference on Applied Stochastic Models and Data Analysis"*, Crete, Greece, May 2007.

[2] A. C. Atkinson and A. N. Donev. *Optimum experimental designs*. Oxford statistical science series. Oxford University Press, 1992.

[3] S. Barkby, S. B. Williams, O. Pizarro, and M. V. Jakuba. A featureless approach to efficient bathymetric slam using distributed particle mapping. *Journal of Field Robotics*, 28(1):19–39, 2011.

[4] A. Bender, S. B. Williams, and O. Pizarro. Classification with probabilistic targets. In *2012 IEEE/RSJ International Conference on Intelligent Robots and systems*, pages 1 – 7, Algarve, Portugal, October 7-12 2012.

[5] J. O. Berger. *Statistical decision theory and bayesian analysis*. Springer series in statistics. Springer-Verlag, 1985.

[6] B. Bingham, B. Foley, H. Singh, R. Camilli, K. Delaporta, R. Eustice, A. Mallios, D. Mindell, C. Roman, and D. Sakellariou. Robotic tools for deep water archaeology: Surveying an ancient shipwreck with an autonomous underwater vehicle. *Journal of Field Robotics*, 27(6):702–717, 2010.

[7] J. Binney, A. Krause, and G. S. Sukhatme. Optimizing waypoints for monitoring spatiotemporal phenomena. *The International Journal of Robotics Research*, 32(8):873–888, 2013.

[8] P. Blondel and O. G. Sichi. Textural analyses of multibeam sonar imagery from Stanton Banks, Northern Ireland continental shelf. *Applied Acoustics*, 70 (10):1288 – 1297, 2009.

[9] P. Boyle and M. Frean. Dependent gaussian processes. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 217–224, Cambridge, MA, 2005. MIT Press.

[10] C. J. Brown and P. Blondel. Developments in the application of multibeam sonar backscatter for seafloor habitat mapping. *Applied Acoustics*, 70(10):1242 – 1247, 2009.

[11] C. J. Brown and J. S. Collier. Mapping benthic habitat in regions of gradational substrata: An automated approach utilising geophysical, geological, and biological relationships. *Estuarine, Coastal and Shelf Science*, 78(1):203 – 214, 2008.

[12] C. J. Brown, S. J. Smith, P. Lawton, and J. T. Anderson. Benthic habitat mapping: A review of progress towards improved understanding of the spatial ecology of the seafloor using acoustic techniques. *Estuarine, Coastal and Shelf Science*, 92(3):502 – 520, 2011.

[13] C. J. Brown, B. J. Todd, V. E. Kostylev, and R. A. Pickrill. Image-based classification of multibeam sonar backscatter data for objective surficial sediment mapping of Georges Bank, Canada. *Continental Shelf Research*, 31 (2, Supplement 1):S110 – S119, 2011. Geological and Biological Mapping and Characterisation of Benthic Marine Environments.

[14] P. Buhl-Mortensen, M. Dolan, and L. Buhl-Mortensen. Prediction of benthic biotopes on a Norwegian offshore bank using a combination of multivariate analysis and GIS classification. *ICES Journal of Marine Science: Journal du Conseil*, 66(9):2026–2032, 2009.

[15] W. Caselton and J. Zidek. Optimal monitoring network designs. *Statistics & Probability Letters*, 2(4):223 – 227, 1984.

[16] K. Chaloner and I. Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):pp. 273–304, 1995.

[17] G. Chust, M. Grande, I. Galparsoro, A. Uriarte, and Ángel Borja. Capabilities of the bathymetric Hawk Eye LiDAR for coastal habitat mapping: A case study within a Basque estuary. *Estuarine, Coastal and Shelf Science*, 89(3): 200 – 213, 2010.

[18] G. R. Cochrane and K. D. Lafferty. Use of acoustic classification of sidescan sonar data for mapping benthic habitat in the Northern Channel Islands, California. *Continental Shelf Research*, 22(5):683 – 690, 2002.

[19] B. Costa, T. Battista, and S. Pittman. Comparative evaluation of airborne lidar and ship-based multibeam sonar bathymetry and intensity for mapping coral reef ecosystems. *Remote Sensing of Environment*, 113(5):1082 – 1100, 2009.

[20] M. P. Deisenroth, C. E. Rasmussen, and J. Peters. Gaussian process dynamic programming. *Neurocomputing*, 72(7-9):1508 – 1524, March 2009. Advances in Machine Learning and Computational Intelligence - 16th European Symposium on Artificial Neural Networks 2008, 16th European Symposium on Artificial Neural Networks 2008.

[21] P. A. del Giorgio and C. M. Duarte. Respiration in the open ocean. *Nature*, 420(6914):379–384, November 2002.

[22] M. F. Dolan, A. J. Grehan, J. C. Guinan, and C. Brown. Modelling the local distribution of cold-water corals in relation to bathymetric variables: Adding spatial context to deep-sea video data. *Deep Sea Research Part I: Oceanographic Research Papers*, 55(11):1564 – 1579, 2008.

[23] W. DuMouchel and B. Jones. A simple Bayesian modification of d-optimal designs to reduce dependence on an assumed model. *Technometrics*, 36(1):pp. 37–47, 1994.

[24] B. Englot and F. Hover. Planning complex inspection tasks using redundant roadmaps. *International Symposium of Robotics Research*, 2011.

[25] A. Friedman, O. Pizarro, S. B. Williams, and M. Johnson-Roberson. Multi-scale measures of rugosity, slope and aspect from benthic stereo image reconstructions. *PLoS ONE*, 7(12), 12 2012.

[26] M. N. Gibbs. *Bayesian gaussian processes for regression and classification.* PhD thesis, Department of Physics, University of Cambridge, 1997.

[27] M. Girolami and S. Rogers. Variational Bayesian multinomial probit regression with Gaussian process priors. *Neural Computation*, 18(8): 1790–1817, August 2006.

[28] Y. Grandvalet and Y. Bengio. Semi-supervised learning by entropy minimization. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 529–536, Cambridge, MA, 2005. MIT Press.

[29] Y. Guo and R. Greiner. Optimistic active learning using mutual information. In *Proceedings of the 20th international joint conference on Artifical intelligence*, pages 823–829, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[30] L. Hamilton and I. Parnum. Acoustic seabed segmentation from direct statistical clustering of entire multibeam sonar backscatter curves. *Continental Shelf Research*, 31(2):138 – 148, 2011.

[31] A. P. Henriques, A. D. D. Neto, and R. F. Amaral. Classification of multispectral images in coral environments using a hybrid of classifier ensembles. *Neurocomputing*, 73(79):1256 – 1264, 2010.

[32] J. Hershey and P. Olsen. Approximating the Kullback Leibler divergence between Gaussian mixture models. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pages IV–317 –IV–320, april 2007.

[33] G. A. Hollinger, B. Englot, F. S. Hover, U. Mitra, and G. S. Sukhatme. Active planning for underwater inspection and the benefit of adaptivity. *The International Journal of Robotics Research*, 32(1):3–18, 2013.

[34] K. Holmes, K. V. Niel, B. Radford, G. Kendrick, and S. Grove. Modelling distribution of marine benthos from hydroacoustics and underwater video. *Continental Shelf Research*, 28(14):1800 – 1810, 2008.

[35] X. Huan and Y. M. Marzouk. Simulation-based optimal bayesian experimental design for nonlinear systems. *Journal of Computational Physics*, 232(1):288 – 317, 2013.

[36] Z. Huang, B. Brooke, and J. Li. Performance of predictive models in marine benthic environments based on predictions of sponge distribution on the australian continental shelf. *Ecological Informatics*, 6(34):205 – 216, 2011.

[37] D. Ierodiaconou, J. Monk, A. Rattray, L. J. B. Laurenson, and V. L. Versace. Comparison of automated classification techniques for predicting benthic biological communities using hydroacoustics and video observations. *Continental Shelf Research*, 31(2, Supplement 1):S28 – S38, 2011. Geological and Biological Mapping and Characterisation of Benthic Marine Environments.

[38] J. Irish and T. White. Coastal engineering applications of high-resolution lidar bathymetry. *Coastal Engineering*, 35(1-2):47 – 71, 1998.

[39] M. Johnson-Roberson, O. Pizarro, S. B. Williams, and I. Mahon. Generation and visualization of large-scale three-dimensional reconstructions from underwater robotic surveys. *Journal of Field Robotics*, 27(1):21–51, 2010.

[40] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(12): 99–134, 1998.

[41] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell. Active learning with Gaussian processes for object categorization. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1 –8, oct. 2007.

[42] A. Kenny, I. Cato, M. Desprez, G. Fader, R. Schttenhelm, and J. Side. An overview of seabed-mapping technologies in the context of marine habitat classification. *ICES Journal of Marine Science: Journal du Conseil*, 60(2): 411–418, 2003.

[43] V. E. Kostylev, B. J. Todd, G. B. J. F. Fader, R. C. Courtney, G. D. M. Cameron, and R. A. Pickrill. Benthic habitat mapping on the Scotian Shelf based on multibeam bathymetry, surficial geology and sea floor photographs. *Marine Ecology Progress Series*, 219:121–137, 2001.

[44] V. E. Kostylev, J. Erlandsson, M. Y. Ming, and G. A. Williams. The relative importance of habitat complexity and surface area in assessing biodiversity: Fractal application on rocky shores. *Ecological Complexity*, 2(3):272 – 286, 2005.

[45] A. Krause, A. Singh, and C. Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9:235–284, June 2008.

[46] K. Kurihara, M. Welling, and N. Vlassis. Accelerated variational dirichlet process mixtures. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 761–768. MIT Press, Cambridge, MA, 2007.

[47] M. Kuss and C. E. Rasmussen. Assessing approximate inference for binary Gaussian process classification. *Journal of Machine Learning Research*, 6: 1679–1704, 2005.

[48] S. M. LaValle. *Planning algorithms*. Cambridge University Press, Cambridge, U.K., 2006.

[49] S. M. LaValle and J. James J. Kuffner. Randomized kinodynamic planning. *The International Journal of Robotics Research*, 20:378–400, 2001.

[50] T. Le Bas and V. Huvenne. Acquisition and processing of backscatter data for habitat mapping - comparison of multibeam and sidescan systems. *Applied Acoustics*, 70(10):1248 – 1257, 2009.

[51] D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 3–12, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[52] J. Li, A. D. Heap, A. Potter, and J. J. Daniell. Application of machine learning methods to spatial interpolation of environmental variables. *Environmental Modelling & Software*, 26(12):1647 – 1659, 2011.

[53] D. V. Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005, 1956.

[54] T. J. Loredo. Bayesian adaptive exploration. *AIP Conference Proceedings*, 707 (1):330–346, 2004.

[55] E. R. Lundblad, D. J. Wright, J. Miller, E. M. Larkin, R. Rinehart, D. F. Naar, B. T. Donahue, S. M. Anderson, and T. Battista. A benthic terrain classification scheme for American Samoa. *Marine Geodesy*, 29(2):89–111, 2006.

[56] D. J. MacKay. Information-based objective functions for active data selection. *Neural Computation*, 4:590–604, July 1992.

[57] I. Mahon, S. B. Williams, O. Pizarro, and M. Johnson-Roberson. Efficient view-based slam using visual loop closures. *Robotics, IEEE Transactions on*, 24(5):1002–1014, Oct.

[58] A. A. Makarenko, S. B. Williams, F. Bourgault, and H. F. Durrant-Whyte. An experiment in integrated exploration. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, volume 1, pages 534 – 539 vol.1, 2002.

[59] I. Marsh and C. Brown. Neural network classification of multibeam backscatter and bathymetry data from stanton bank (area iv). *Applied Acoustics*, 70(10):1269 – 1276, 2009.

[60] C. McGonigle, J. H. Grabowski, C. J. Brown, T. C. Weber, and R. Quinn. Detection of deep water benthic macroalgae using image-based classification techniques on multibeam backscatter at cashes ledge, gulf of maine, usa. *Estuarine, Coastal and Shelf Science*, 91(1):87 – 101, 2011.

[61] T. P. Minka. *A family of algorithms for approximate bayesian inference*. PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, January 2001.

[62] M. A. Morrison, A. G. Jeffs, and S. H. Hooker. Detection of sub-tidal mussel beds on rocky reefs using acoustics. *International Journal of Remote Sensing*, 31(23):6061–6071, 2010.

[63] R. M. Neal. Monte Carlo implementation of Gaussian process models for Bayesian regression and classification. Technical Report 9702, Department of Statistics, University of Toronto, January 1997.

[64] S. Nichol, T. Anderson, M. McArthur, N. Barrett, A. Heap, P. Siwabessy, and B. Brooke. Southeast Tasmania temperate reef survey, post survey report. Technical Report Record 2009/43, Geoscience Australia, 2009.

[65] H. Nickisch and C. E. Rasmussen. Approximations for binary Gaussian process classification. *Journal of Machine Learning Research*, 9:2035 – 2078, 2008.

[66] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39:1–34, May 2000.

[67] S. O'Callaghan, F. T. Ramos, and H. Durrant-Whyte. Contextual occupancy maps using Gaussian processes. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 1054–1060, 2009.

[68] M. Osborne. *Bayesian Gaussian Processes for Sequential Prediction, Optimisation and Quadrature*. PhD thesis, University of Oxford, 2010.

[69] C. Papadimitriou and J. N. Tsitsiklis. The complexity of markov decision processes. *Mathematics of Operations Research*, 12(3):441–450, 1987.

[70] J. Pineau, G. Gordon, and S. Thrun. Point-based value iteration: An anytime algorithm for POMDPs. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1025 – 1032, August 2003.

[71] M. Pivtoraiko, R. A. Knepper, and A. Kelly. Differentially constrained mobile robot motion planning in state lattices. *Journal of Field Robotics*, 26(3): 308–333, March 2009.

[72] J. C. Platt. Probabilities for SV machines. In *Advances in Large Margin Classifiers*, page 6174. MIT Press, 2000.

[73] J. M. Porta, N. Vlassis, M. T. Spaan, and P. Poupart. Point-based value iteration for continuous POMDPs. *The Journal of Machine Learning Research*, 7:2329–2367, 2006.

[74] C. E. Rasmussen. *Evaluation of gaussian processes and other methods for non-linear regression*. PhD thesis, Graduate Department of Computer Science, University of Toronto, 1996.

[75] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning (adaptive computation and machine learning)*. The MIT Press, 2006.

[76] A. Reid, F. Ramos, and S. Sukkarieh. Multi-class classification of vegetation in natural environments using an unmanned aerial system. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2953 –2959, may 2011.

[77] R. Rifkin and A. Klautau. In defense of one-vs-all classification. *Journal of Machine Learning Research*, 5:101 – 141, 2004.

[78] P. Rigby, O. Pizarro, and S. B. Williams. Toward adaptive benthic habitat mapping using Gaussian process classification. *Journal of Field Robotics*, 27 (6):741–758, 2010.

[79] C. Roman and H. Singh. A self-consistent bathymetric mapping algorithm. *Journal of Field Robotics*, 24(1-2):23–50, 2007.

[80] C. N. Rooper and M. Zimmermann. A bottom-up methodology for integrating underwater video and acoustic mapping for seafloor substrate classification. *Continental Shelf Research*, 27(7):947 – 957, 2007.

[81] A. Rottmann and W. Burgard. Adaptive autonomous control using online value iteration with Gaussian processes. In *Proceedings of the 2009 IEEE international conference on Robotics and Automation*, pages 2106 – 2111, Kobe, Japan, May 2009.

[82] N. Roy, G. Gordon, and S. Thrun. Finding approximate POMDP solutions through belief compression. *Journal of Artificial Intelligence Research*, 23: 1–40, 2005.

[83] P. Sebastiani and H. P. Wynn. Maximum entropy sampling and optimal Bayesian experimental design. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(1):145–157, 2000.

[84] B. Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[85] B. Settles and M. Craven. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 1070–1079, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[86] E. Shumchenia and J. King. Comparison of methods for integrating biological and physical data for marine habitat mapping and classification. *Continental Shelf Research*, 30(16):1717 – 1729, 2010.

[87] R. Sim and N. Roy. Global A-optimal robot exploration in SLAM. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 661–666, 2005.

[88] A. Singh, A. Krause, C. Guestrin, and W. J. Kaiser. Efficient informative sensing using multiple robots. *Journal of Artificial Intelligence Research*, 34: 707–755, April 2009.

[89] H. Singh, R. Armstrong, F. Gilbes, R. Eustice, C. Roman, O. Pizarro, and J. Torres. Imaging coral I: Imaging coral habitats with the seabed auv. *Subsurface Sensing Technologies and Applications*, 5:25 – 42, January 2004.

[90] R. N. Smith, M. Schwager, S. L. Smith, B. H. Jones, D. Rus, and G. S. Sukhatme. Persistent ocean monitoring with underwater gliders: Adapting spatiotemporal sampling resolution. *Journal of Field Robotics*, 28(5):714 – 741, Sep 2011.

[91] D. M. Steinberg, A. Friedman, O. Pizarro, and S. B. Williams. A Bayesian nonparametric approach to clustering data from underwater robotic surveys. In *15th International Symposium on Robotics Research*, Flagstaff, AZ, USA, 28 August – 1 September 2011.

[92] A. Stentz. The focussed D* algorithm for real-time replanning. In *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*, IJCAI'95, pages 1652–1659, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[93] R. Sutton and A. Barto. *Reinforcement learning: an introduction*. Adaptive computation and machine learning. MIT Press, 1998.

[94] D. R. Thompson, D. S. Wettergreen, and F. J. C. Peralta. Autonomous science during large-scale robotic survey. *Journal of Field Robotics*, 28(4): 542–564, 2011.

[95] S. Thrun, W. Burgard, and D. Fox. *Probabilistic robotics*. The MIT Press, August 2005.

[96] H. van Rein, C. Brown, R. Quinn, and J. Breen. A review of sublittoral monitoring methods in temperate waters: a focus on scale. *Underwater Technology: The International Journal of the Society for Underwater*, 28(3): 99–113, 2009.

[97] C. K. I. Williams and D. Barber. Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1342 – 1351, December 1998.

[98] S. Williams, O. Pizarro, M. Jakuba, C. Johnson, N. Barrett, R. Babcock, G. Kendrick, P. Steinberg, A. Heyward, P. Doherty, I. Mahon, M. Johnson-Roberson, D. Steinberg, and A. Friedman. Monitoring of benthic reference sites: Using an autonomous underwater vehicle. *Robotics Automation Magazine, IEEE*, 19(1):73–84, 2012.

[99] S. B. Williams, O. Pizarro, J. M. Webster, R. J. Beaman, I. Mahon, M. Johnson-Roberson, and T. C. L. Bridge. Autonomous underwater vehicle-assisted surveying of drowned reefs on the shelf edge of the Great Barrier Reef, Australia. *Journal of Field Robotics*, 27(5):675–697, 2010.

[100] S. Williams, O. Pizarro, I. Mahon, and M. Johnson-Roberson. Simultaneous localisation and mapping and dense stereoscopic seafloor reconstruction using an auv. In O. Khatib, V. Kumar, and G. Pappas, editors, *Experimental Robotics*, volume 54 of *Springer Tracts in Advanced Robotics*, pages 407–416. Springer Berlin Heidelberg, 2009.

[101] M. F. Wilson, B. O'Connell, C. Brown, J. C. Guinan, and A. J. Grehan. Multiscale terrain analysis of multibeam bathymetry data for habitat mapping on the continental slope. *Marine Geology*, 30:3–35, 2007.

[102] S. Zieger, T. Stieglitz, and S. Kininmonth. Mapping reef features from multibeam sonar data using multiscale morphometric analysis. *Marine Geology*, 264(3-4):209 – 217, 2009.