# Efficient, Multi-Fidelity Perceptual Representations via Hierarchical Gaussian Mixture Models

Shobhit Srivastava
August 7, 2017
CMU-RI-TR-17-44


The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania

**Thesis Committee:**
Nathan Michael, *Chair*
Artur W. Dubrawski
Benjamin Eckart

*Submitted in partial fulfillment of the requirements for the degree of Master of Science in Robotics.*

**Abstract**

Operation of mobile autonomous systems in real-world environments and their participation in the accomplishment of meaningful tasks requires a high-fidelity perceptual representation that enables efficient inference. It is challenging to reason efficiently in the space of sensor observations primarily due to the dependence of measurements on the type of sensor, noise in measurements and in some cases, the prohibitive size of sensor data. A perceptual representation that abstracts out sensor nuances is thus required to enable effective and efficient reasoning in *a priori* unknown environments. This thesis presents a probabilistic environment representation that allows efficient high-fidelity modeling and inference towards enabling informed planning (active perception) on a computationally constrained mobile autonomous system. A major challenge is the need for real-time generation and update of the model given the computational and memory limitations on a mobile robot. This constraint has generally resulted in a compromise on the fidelity of the model in existing literature in the mobile robot community.

To address this challenge, the proposed approach exploits the structure of real world environments and models dependencies between spatially distinct locations. Gaussian Mixture Models are employed to capture these structural dependencies and learn a semi-parametric continuous spatial model from the measurements of the environment. A hierarchy of these arbitrary resolution models enables a multi-fidelity representation with the variation in fidelity quantified via information-theoretic measures. Crucially for active perception, the spatial model is extended to a distribution

over occupancy with a measure of uncertainty incorporated via a variance estimate associated with model predictions. The compact nature and representative capability of the proposed model coupled with a real-time embedded GPU-based implementation enables high-fidelity and memory-efficient modeling and inference as demonstrated on real-world datasets in diverse environments.

# Acknowledgements

I have had the opportunity to work with a bunch of interesting people during my time here at Carnegie Mellon, for which I would forever be grateful.

I would like to start with a strong word of gratitude for my advisor, Prof. Nathan Michael, for providing me a research direction, guiding me throughout the research process and most importantly, teaching me how to think like a researcher.

I would also like to thank all my friends in the lab, especially Wennie Tabib, Vibhav Ganesh and Timothy Lee, who made working at the lab interesting, fun and 'good' (strictly in this order). Special mention to Aditya Dhawale for his unrelenting, determined efforts to achieve reliable operation of all sensors. This work would have not have been possible without your efforts and I sincerely hope that your Ph.D. dissertation manages to revolutionize sensors as we know it.

Also, a big thank you to all my friends outside CMU and family for their continued support that helped me get through tough times.

Lastly, we gracefully acknowledge support from Westinghouse and ARL grant W911NF-08-2-0004 for this work.

# Contents

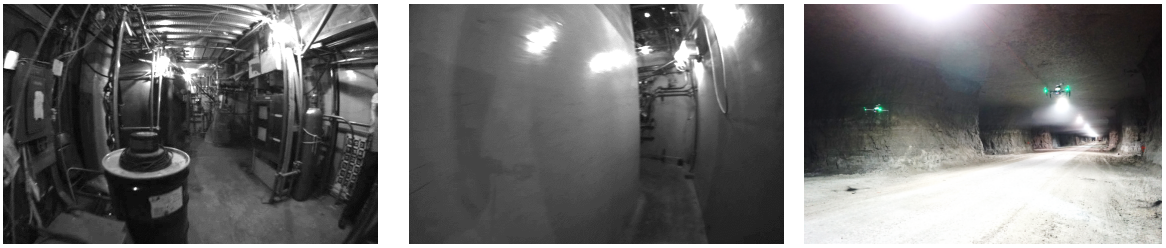
# List of Tables

# List of Figures

# Chapter 1

# Introduction

Autonomous systems are increasingly being deployed for operation in perceptually challenging, diverse and potentially hazardous environments such as monitoring power plants, information-gathering in underground mines and tunnels, and search and rescue operations in disaster-hit areas. The operating environment for such operations is not always known *a priori* and thus a representation of the environment to enable reasoning with respect to the surroundings needs to be generated online. Information pertaining to the environment is made available to the mobile systems via sensors and thus a fundamental requirement to enable operation in real-world environments is efficient reasoning and inference over acquired sensor measurements. It is not always feasible to operate directly on the raw point measurements obtained from the sensors. Major reasons for this include the dependence of the nature of measurements on the characteristics of the sensor (for example, measurements from a LIDAR are typically relatively sparse as compared to those from an RGBD sensor), noise in the sensor measurements, computational tractability of operating on dense sensor data and the inability of points to capture meaningful perceptual correlations in



(a) (b) (c)

Figure 1.1: Diversity in perceptual information. (a) A high-clutter, small-scale engineered environment (average range $\approx 3\ m$), (b) A low-texture, small-scale engineered environment, and (c) A large-scale natural environment (average range $\approx 10\ m$.)

the environment, such as geometric shapes and structural dependencies. Another important requirement to enable planning and navigation in real-world and *a priori* unknown environments is the generation of a precise occupancy representation. A major challenge in the online generation of a high-fidelity representation is posed by the computational constraints on a mobile robot that include restricted processing power and limited memory. Another significant challenge is the perceptual diversity in real-world environments in terms of scale, clutter and type of structure (man-made versus natural) (Fig. 1.1).

A computationally efficient perceptual modeling strategy is required that can scale to potentially large and diverse environments and is able to handle sparsity in sensor measurements. A compact representation of sensor data would enable efficient inference via retention of the information contained in the data at a reduced memory footprint. The representation would also enable homogenization of different sensing modalities into a unified model that would serve as an interface to higher level perceptual algorithms. Further, informed planning in the environment would be enabled if an uncertainty measure is associated with model predictions.

## 1.1 Approach

There are several characteristics that are desired in the perceptual representation to be able to address all the challenges mentioned earlier. The following is a list of desired properties that a sufficient spatial representation should have.

- **High-Fidelity and Generative**: The representation should provide a high-fidelity representation of the sensor data. Also, it should be possible to re-generate the sensor observations from the model as and when required.

- **Memory-Efficient**: The model should have a relatively small memory footprint. This is required to enable scalability to large environments.

- **Computationally-Tractable**: It should be possible to generate and update the model online and in real-time on a compute-constrained mobile robot to enable efficient operation in unfamiliar environments.

- **Generalizable**: The perceptual representation should be generalizable to diverse environments and robust to the peculiarities of a given environment. This implies that minimal parameter-tuning should be required for obtaining a high-fidelity representation across environments exhibiting variation in scale, degree of clutter and the nature of structure.

- **Robust to Sparsity**: The representation should be able to handle sparsity in sensor data and capture spatial dependencies induced by surfaces in the environment.

- **Hierarchical**: The representation should enable operation at different degrees of fidelity depending of the computational budget of the concerned application. Further, it should be possible to quantify the variation in fidelity of the

representation across the hierarchy to enable informed selection of a particular fidelity-level.

- **Uncertainty Measure**: The model should support a measure of uncertainty associated with model predictions. This measure is crucial to enable informed planning (active perception) in the operating environment.

Several approaches presented in literature including occupancy Grids [15] and Normal Distribution Transform [42] make restrictive assumptions, such as conditional independence between locations in space, to enable a computationally-tractable formulation. However, these assumptions adversely impact the accuracy of the representation and scalability of the approach to large environments. Other approaches that capture spatial dependencies, such as Gaussian Process Occupancy Maps ( [34, 52]) are prohibitively expensive to be feasible for large-scale operation. An approach based on a hierarchy of *Gaussian Mixtures* is explored in this thesis to model the raw point sensor observations. Gaussian Mixtures provide a compact semi-parametric representation capable of modeling arbitrary multimodal distributions given the right number of components. Further, Gaussian Mixtures are generative in nature and inherently able to encode spatial correlations and geometric connectivity. Also, the continuous nature of the model enables principled estimation of information-theoretic measures that enables of a hierarchy in terms of the information content of the model.

This thesis develops 3D Gaussian Mixture Models as an approach to large-scale environment representation based on point clouds obtained from range sensors. The 3D model is extended into a probabilistic representation of occupancy via incorporation of free-space information. A thorough evaluation of the proposed approach is provided along with a comparison to state of the art for online surface modeling and

occupancy representation. The implications of the hierarchy are investigated and a methodology to estimate the required number of components based on information-theoretic measures is developed. A measure of uncertainty in the form of an associated variance estimate is incorporated via a regression framework and characterized.

## 1.2 Thesis Contributions

The contribution of this work is the development of a methodology to represent sequential sensor observations as a coherent, compact and high-fidelity world model. The experiments conducted and the results presented in this thesis are limited to 3D point cloud data. However, the proposed formulation is easily extensible to other sensing modalities. Specifically, the thesis offers the following contributions:

- **Large-Scale Generative Modeling**: A methodology to learn a hierarchical generative model for point clouds amenable to incremental updates with sequential point clouds thus enabling a large-scale environment representation.
- **Continuous Probabilistic Occupancy Representation**: A continuous distribution over occupancy, obtained from the generative surface model, via incorporation of free-space information.
- **Online Inference**: An associated measure of uncertainty via a variance estimate to enable online inference with respect to the model.
- **Real-time Implementation**: A GPU-based parallelized implementation capable of operating in real-time on an embedded System-on-Chip (SoC).
- **Evaluation on real-world datasets**: A thorough evaluation on diverse real-world datasets to investigate generalizability, accuracy and memory footprint

of the proposed approach along with a comparison to state of the art.

## 1.3 Thesis Outline

The thesis is structured as follows.

- **Chapter 2**: A description of the approaches for surface modeling and occupancy representation presented in the literature is covered.
- **Chapter 3**: The hierarchical generative spatial model based on Gaussian Mixtures is introduced and developed.
- **Chapter 4**: A model of observed free-space is developed and the surface model is augmented into a probabilistic representation of occupancy via incorporation of free-space information. An uncertainty measure is also introduced.
- **Chapter 5**: The proposed model is extended to incorporate information from multiple sensing modalities and enable efficient multimodal inference.
- **Chapter 6**: The proposed approach is summarized and directions for future work are explored.

# Chapter 2

# Background

A high-fidelity representation of the environment is crucial for operation of mobile robots in real-world environments. Consequently, there has been a myriad of work toward enabling computationally tractable representations that can enable successful accomplishment of the task at hand. A noticeable trend in the development of environment and occupancy representations is the trade-off between computational complexity and model fidelity. Research in the late 1980's and 1990's compromised on model-fidelity to achieve a real-time viable representation [15, 49, 55]. This trend continued till early 2000's when several higher fidelity representations were proposed [76–78], leading to increased emphasis on precise representations. A major motivation toward high-fidelity representations has been the growing availability of industry-grade, high-resolution range sensors capable of providing dense measurements of the environment. This chapter describes some of the significant efforts at environment representation over the years that have influenced the development of the proposed approach.

The approaches that have been proposed in the literature can be classified into one

or more of the following categories: **Voxel-based** representations, that involve tessellation of the environment into cells, **Generative Surface Models**, that involve learning surface models based on sensor data and obtaining an occupancy representation from the models, and **Continuous** representations, that approximate the underlying occupancy distribution via a continuous function.

## 2.1 Voxel Based Representations

### 2.1.1 Occupancy Grids

Occupancy grid maps were introduced by Elfes [15] and Moravec [49] in 1989 and have since been widely used as the spatial representation throughout the mobile robot community. These maps involve discretization of the environment into cells of a predefined fixed size, with the likelihood of occupancy stored per cell. Each cell is classified as either occupied, free or unknown based on the number of sensor rays that pass through the cell. The occupancy status of a cell is represented by the log-odds ratio

$$l_i \equiv log\frac{o_i}{1-o_i} \tag{2.1}$$

where $o_i$ is the probability of occupancy of the cell and the log-odds ratio is updated as

$$l_i \leftarrow l_i \ + \ L(m|z_t) \tag{2.2}$$

where $L$ is the inverse sensor-model [78].

The simplicity and computational efficiency associated with occupancy grid maps comes at the expense of certain restrictive assumptions that adversely affect model fidelity. Specifically, occupancy grids assume that the likelihood of cell occupancy

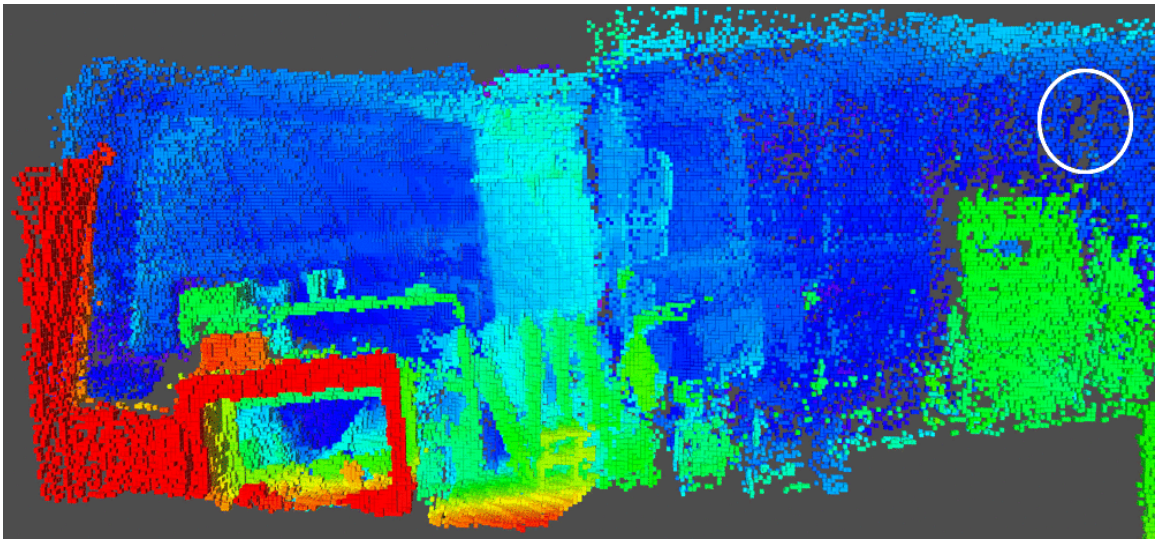


Figure 2.1: Occupancy grid at a resolution of 4 cm. The vulnerability of the representation to measurement sparsity results in holes in the map highlighted via a white ellipse.

depends only on the rays that pass through the cell and is independent of other measurements. This allows the joint probability of occupancy of the map $m$ to be expressed as the product of individual cell occupancy probabilities

$$p(m|x_{1:t}, z_{1:t}) \;=\; \prod_i p(m_i|x_{1:t}, z_{1:t}). \tag{2.3}$$

where $x_{1:t}$ is the history of robot states and $z_{1:t}$ is the history of sensor measurements. This conditional independence assumption makes occupancy grid maps computationally efficient but also precludes the model from capturing spatial dependencies, resulting in incorrect classifications due to sparsity in sensor measurements. Sensor sparsity typically results in holes in the occupancy map in regions that did not experience sensor rays. Further, the size of a voxel in an occupancy grid is a user-defined parameter, generally referred to as resolution, that needs to be predefined. This limits the the generalizability of the model to environments exhibiting structural diversity as one fixed cell-size might fail to capture fine structural detail in one region of the

environment and consume unnecessary memory in some other region. A high-fidelity occupancy grid for a complex environment calls for a small cell-size that in turn leads to a high memory footprint and increased vulnerability to sparsity of measurements (Fig. 2.1).

### 2.1.2 Octrees

Octrees were introduced by Moravec for 3D computer graphics in order to compute Fast Fourier Transforms [48]. Payeur et al. [55] proposed an approach for 3D modeling in a robotics context based on Octrees [47] to address the memory concerns associated with occupancy grids and achieve compactness via on-demand sub-division of cells. Each cell in this representation gets divided into eight octets, if the cell experiences partial hits and partial misses. A similar approach was used by Fournier et al. [20] and Fairfield et al. [17] and extended by Hornung et al. [28], who incorporated online map compression via pruning of cells with the same occupancy status. Octrees serve to reduce the memory footprint in environments with a varying degree of clutter. However, the leaf voxel-size of the tree needs to be specified as a parameter which may impact representation fidelity in highly cluttered environments and require prior knowledge of the environment that may not always be available. Also, the representation assumes conditional independence between cells that affects representation fidelity.

### 2.1.3 Elevation Maps

An approach to obtain relatively compact representation leverages elevation maps that are a 2.5D parameterization of space obtained by associating height values to cells organized in a 2D grid. However, this approach has representation limitations

due to a single height value per cell. Specifically, it is challenging to obtain a high-fidelity representation of a curved surface with an elevation map. Triebel et al. [82] extend the 2.5D representation to incorporate multiple height values per cell while Lang et al. [38] and Plagemann et al. [58] formulate the height value at each cell as a non-parametric Bayesian Regression. 2.5D representations have been extensively used for terrain modeling to enable humanoid robot locomotion [2, 26, 30]. However, a 2.5D representation is limited by the resolution of the 2D grid and is vulnerable to discretization errors.

### 2.1.4 Forward Sensor Models

The vulnerability to sparsity in sensor measurements can be attributed to the conditional independence assumption made by occupancy grids (2.3). Specifically, occupancy grids assume that the likelihood of cell occupancy depends only on the rays that pass through the cell and is independent of other measurements. This assumption has been addressed by Thrun [77] who proposed an approach for map updates based on forward models to transform the updates into a latent-variable optimization and thus maintain dependencies between cells. A forward sensor model considers $L(z_t|m)$ instead of the $L(m|z_t)$ considered in inverse sensor models (2.2). This allows a sensor measurement to contribute to more than one voxels in the representation making it relatively more robust to sensor sparsity. Forward sensor models have since been employed for efficient sensor-fusion [54] and robust sonar sensing [40]. The proposed optimization, however, lacks an analytic solution and Expectation-Maximization (EM) [10] is employed to iteratively maximize the likelihood of sensor observations given the map and the set of latent variables. An unfortunate drawback of this approach is the requirement to optimize in a high-dimensional space, corre-

sponding to the size of the map, to realize the map updates which may be challenging for online operation.

## 2.2 Generative Surface Models

This class of techniques seeks to approximate the environment via models learned based on acquired sensor data. The parametric surface models enable a compact representation that is generative in nature and are capable of leveraging structural dependencies to learn low complexity models.

### 2.2.1 Latent-Variable Optimization Methods

Thrun et al. [76] fit a set of 3D planes to represent the environment with the number of planes estimated via a Bayesian prior that penalizes complex maps. Expectation Maximization is employed to obtain a maximum likelihood assignment of raw sensor points to planes aided by a set of binary correspondence variables. The model provides a compact 3D representation of indoor environments and the generative nature allows high-fidelity reconstruction of the sensor data. The approach, however, makes an assumption of planarity of structure in the environment that precludes generalizability to diverse environments. Also, it is non-trivial to incorporate free-space information into the model and obtain an occupancy representation.

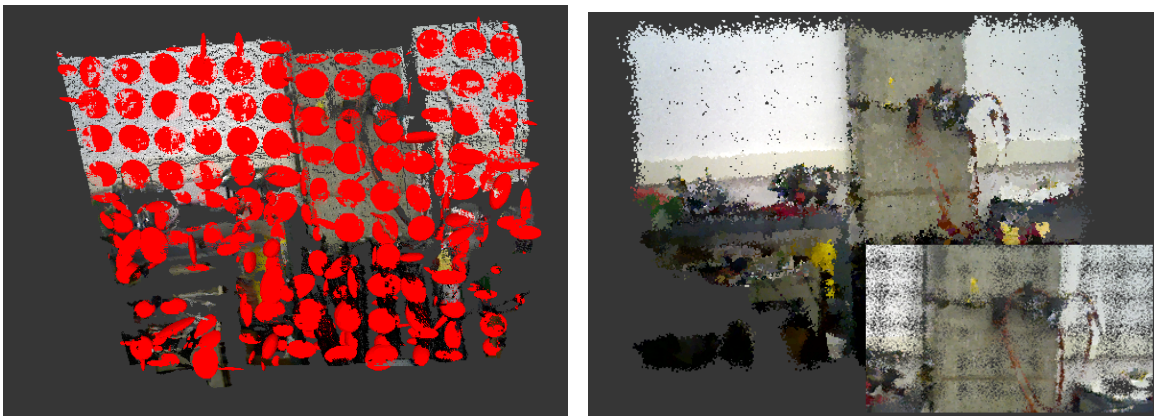
Veeck et al. [84] proposed a model based on polylines as a continuous environment representation. Specifically, the environment is represented by a set of line segments obtained via an optimization over the distance of the segments to the scan-points. The optimization is initialized via the Bayesian Information Criterion [50] and the initial set of line segments is then optimized to obtain the desired representation. This

work was extended by Lakaemper et al. [37] to incorporate incremental updates via fusion in a multi-robot context. The approach, however, is geared toward modeling 2D laser scans and has been found to be prone to consistency issues [56]. The work of Veeck et al. was extended by Paskin et al. [53] to enable inference over occupancy via a framework based on polygonal random fields. A polygonal coloring scheme is employed to represent occupied and free regions as polygons with the color discontinuities forming line segments. A drawback of this approach is the computational cost of generating the map that, as noted by the authors, is prohibitive for online operation.

### 2.2.2 Normal Distribution Transform

Normal Distribution Transform (NDT) is a spatial representation initially proposed by Biber et al. [4] for 2D scan matching. The approach involves subdivision of the 2D plane into cells and a a normal distribution, that locally models the probability of measuring a point, is assigned to each cell. The result is a piece-wise continuous and differentiable probability density that can be used to match another scan using Newton's method [19]. The approach was extended to 3D by Magnusson et al. [42] and has since been used for fast scan-matching [72], loop-closure detection [41], anomaly detection [1], and point cloud segmentation [24]. NDT is a hybrid strategy as it augments a tessellation-based approach with a generative Gaussian distribution. Representing the laser scans that pass through a given voxel via a Gaussian distribution enables a higher-fidelity representation than an occupancy grid of the same resolution.

The perceptual representation was extended to an occupancy mapping framework, Normal Distribution Transform Occupancy Map (NDT-OM), by Saarinen et al. [68]



(a) (b)

Figure 2.2: Illustration of the Normal Distribution Transform (NDT) representation (a) on a point cloud dataset representing a cluttered engineered environment. The red ellipses represent the covariance of the Gaussian distributions learned per cell (cell-size: 10 cm). The representation limitations are highlighted in (b) where the sensor data is reconstructed via sampling. Higher uncertainties are observed at cell-boundaries (in the zoomed-in view) as a consequence of clipped Gaussian distributions employed by the representation. *RGB information is only for illustration.*

via incorporation of a log-likelihood based occupancy update per cell. Free-space is explicitly modeled via an occupancy grid and a multi-scale representation is enabled through moment-based merging of Gaussian distributions. The representation is better able to capture surfaces in the environment as compared to an occupancy grid and is more robust to tessellation artifacts. The framework has been applied extensively for mapping and localization [3, 67, 83]. However, the model assumes conditional cell independence that restricts the support of the Gaussian distributions to the parent cell leading to higher uncertainty at cell boundaries [70] (Fig. 2.2). Also, the representation is sensitive to the size of the voxel and typically requires prior knowledge of the operating environment to obtain desired performance (Sect. 3.4.1).

## 2.3 Continuous Occupancy Representations

This class of approaches represent occupancy as a continuous probability distribution. This makes the representation robust to sensor sparsity as the underlying distribution, from which the sensor measurements are sampled, is being approximated. The two noteworthy approaches presented in the literature that approximate the environment as a continuous distribution are discriminative in nature.

### 2.3.1 Gaussian Process Occupancy Maps

A Gaussian process (GP) is a particular kind of statistical model where observations occur in a continuous domain. In a Gaussian process, every point in some continuous input space is associated with a normally distributed random variable. Moreover, every finite collection of those random variables has a multivariate normal distribution, i.e. every finite linear combination of them is normally distributed. GPs are non-parametric approaches in that they do not specify an explicit functional model between the input and output. They can be viewed as a Gaussian probability distribution in function space and are characterized by a mean function, $\mu(x)$, and the covariance function, $k(x, x^*)$, where $x$ and $x^*$ are both input vectors. Hence, the process itself can be thought of as a distribution over an infinite number of possible functions and inference takes place directly in the space of functions. By assuming that the target data is jointly Gaussian,

$$f(x^*) \;=\; \mathcal{N}(\mu, \sigma) \tag{2.4}$$

where

$$\mu \;=\; k(x^*, X)\,[k(X, X) \;+\; \sigma_n^2 I]^{-1}\, y \tag{2.5}$$

$$\sigma \;=\; k(x^*, x^*) \;-\; k(x^*, X)\,[k(X, X) \;+\; \sigma_n^2 I]^{-1}\, k(X, x^*) \tag{2.6}$$

Here $X$ is a $D$ x $n$ matrix representing the training input data where D is the dimensionality of the data and $n$ corresponds to the total number of measurements employed by model. $x^*$ refers to a query (or test) location. Here, $y$ represents noisy observations of the function at the training locations, $f(X)$; $\sigma_n^2$ is the variance of the global noise; k( X , X ), or simply K, is the matrix of the covariance function values evaluated at all pairs of training inputs. The vector $k(X, x^*)$ is the covariance between the training set and the test set defined depending on a covariance function $k$ that is parameterized by hyper-parameters $\theta$. A detailed treatment of Gaussian Processes can be found in the work of Rasmussen et al. [63].

O'Callaghan et al. [52] employ Gaussian Processes (GPs) as a non-parametric Bayesian learning technique to model occupancy in the environment. The predictive mean and variance are squashed via a sigmoid function to obtain probability of occupancy at a location. The non-parametric nature of the approach enables arbitrary resolution representations of complex structure. GP based occupancy representations have been used for path-planning [43] and exploration [21, 29]. The major drawback of this approach is the high memory footprint resulting from the need to cache the training data and the computational complexity which grows cubically with the size of training data. Kim et al. [34] propose an occupancy mapping formulation leveraging sparse local Gaussian Processes. The training data is partitioned into grid blocks of size given by the characteristic length of the covariance function and the predictive mean and variance is estimated via an extended block. Incremental up-

dates via Bayesian Committee Machines [81] has been proposed by Kim et al. [35] and extended by Wang et al. [85] to further reduce the computational complexity via test-data octrees. The scalability of Gaussian Process based mapping to potentially large environments remains a concern as training data needs be to cached to estimate mean and variance for a query location. Estimation of predictive mean and variance requires inversion of the covariance matrix (2.5), the size of which depends on the training data. Also, offline training of hyper-parameters requires prior knowledge of the operating environment that may not always be available.

### 2.3.2 Hilbert Maps

Recently, Hilbert maps have been proposed by Ramos et al. [62] as an efficient alternative to GP based mapping. Occupancy is represented as a linear discriminative model operating on a high-dimensional feature vector obtained by projecting observations into a reproducing kernel Hilbert space. Advances in kernel approximations, such as Random Fourier Features [61] and Nystrom Approximations [86], are leveraged to obtain feature vectors used to train the model via stochastic gradient descent. The work has since been extended by Guizilini et al. [25] to make feature selection more principled via k-means clustering and local queries more efficient via a KD-tree. However, the number of clusters (or inducing points) required and the size of the neighborhood for querying the model are user-defined parameters that might affect the model accuracy. Also, extension of the approach to associate an uncertainty measure with model predictions is not trivial.

## 2.4 Comparison with the Proposed Approach

The approach proposed in this work borrows elements from the techniques described in this chapter. The approach involves learning a continuous spatial model via a latent variable optimization similar to the techniques presented in Sect. 2.2.1. Gaussian Mixture Models (GMMs) are employed as a semi-parametric learning technique to capture spatial dependencies and obtain a high-fidelity, compact environment representation. The proposed methodology is similar in spirit to the work of Thrun [76] in that Expectation Maximization is used to maximize joint likelihood of the data and correspondence variables. However, a mixture of coupled Gaussian distributions is learned instead of a set of decoupled 3D planes that enables the proposed representation to operate in non-planar environments. Also, information-theoretic measures are employed to estimate the required number of components and a multi-fidelity representation is enabled via a hierarchy of Gaussian Mixture Models. Occupancy in the environment is approximated by a continuous distribution similar to the approaches proposed in Sect. 2.3. However, a semi-parametric GMM is employed for both occupied and free space instead of a non-parametric Gaussian Process (Sect. 2.3.1) or a kernel logistic regression (Sect. 2.3.2). Hierarchical Gaussian Mixture Models have been proposed as a precise representation for point cloud data by Eckart et al. [12, 14]. A top-down hierarchy is proposed with each Gaussian component at level $l$ further divided into $m$ components at level $l + 1$. However, the work is aimed at point cloud registration and does not consider the challenges associated with large-scale environment mapping.

## 2.5 Choice of Model

The proposed approach employs a hierarchy of Gaussian Mixture Models as a multi-fidelity representation of information pertaining to the environment acquired via sensors. This section discusses the other candidate modeling techniques and provides justification for using GMMs. Considering the high-degree of non-linearity in the environment, simplistic models like linear regression are clearly not complex enough to represent surfaces in the environment. High-degree polynomials might provide the desired representative capability. However, the diversity in environments in terms of the complexity of perceptual information makes estimating the required degree of the polynomial challenging. Further, a hierarchical representation based on polynomial models is non-trivial.

Kernel-based non-parametric methodologies are motivated as they do assume a fixed functional form and are capable of approximating arbitrary functions given enough data. Kernel-based approaches have been leveraged via Gaussian Process Occupancy Mapping [34, 35, 52] and Hilbert Maps [11, 25, 62]. Non-parametric techniques are inherently data-driven, and consequently, pose a challenge in terms of scalability to large environments. More the size of data, more is the associated computational complexity. A modeling technique that can approximate the fidelity of a non-parametric approach but at a reduced computational and memory footprint is thus required. A Gaussian Mixture Model (GMM) is an intermediate semi-parametric model that can approximate a Gaussian Process given sufficient number of components. Further, the associated memory footprint is significantly smaller as a result of the parametric form of a Gaussian component. One major advantage with GMMs is that the continuous GMM density function enables principled generation of a hi-

erarchy based on the information-content of the pdf. A noteworthy approach to hierarchical representation is based on Hierarchical Dirichlet Processes (HDPs) [75], an extension of Dirichlet Processes that enables reasoning about hierarchies. However, HDPs are more suited to model data that exhibits a natural grouping of the same set of shared elements (called atoms) across groups. Real-world environments do not inherently exhibit such grouping making HDPs not so suitable for surface modeling.

## 2.6 Summary

An overview of the various approaches to enable online surface modeling and occupancy mapping presented in the literature is provided in this chapter. The techniques can be categorized into three main classes: (a) Voxel-Based representations that discretize the environment into voxels and maintain the likelihood of occupancy per cell assuming conditional independence with other cells. Several improvements have been proposed to this representation to address memory concerns (Octomap), artifacts of conditional independence (forward sensor models) and compactness of the model (Elevation maps). (b) Generative surface models that learn a parametric model over the environment to enable a compact representation. A hybrid strategy, that is a combination of voxel-based representations and generative models is the Normal Distribution Transform (NDT) that learns a Gaussian distribution over the rays that pass through a voxel. (c) Continuous Occupancy representations that learn a continuous distribution over occupancy in the environment. Two approaches of significance include Gaussian Process Occupancy Maps that employ Gaussian Processes to estimate the occupancy distribution, and Hilbert maps that project the sensor data into

a higher-dimensional space and employ logistic regression as a discriminative model over occupancy.

# Chapter 3

# The Spatial Model

The 3D multi-fidelity generative model based on a hierarchy of Gaussian Mixtures is developed in this chapter. The number of components in the mixture is crucial in determining the representation capability of the the model. An approach based on information-theoretic principles is developed to estimate the mixture size required for a high-fidelity and compact representation of the environment. An incremental update strategy based on the novelty of information acquired via sensors is developed. The approach enables the model to scale with the information-content of the environment instead of the size of the environment as is generally observed with voxel-based approaches. In other words, more the number of objects and structural elements to represent, more is the size of the model and vice versa.

The chapter begins with a discussion of Gaussian Mixture Models and the procedure for training them via Expectation Maximization. Information-theoretic measures are then introduced that enable estimation of the number of components required for a high-fidelity representation (referred to as fidelity threshold $\lambda_f$). Algorithms to generate and incrementally update the hierarchy are presented followed by

an evaluation of the approach on real-world datasets.

## 3.1 Gaussian Mixture Models

Gaussian Mixture Models (GMMs) have been established as powerful statistical tool to model multimodal probability distributions and have been used extensively in Machine Learning literature. Given the right number of components, GMMs are capable of modeling arbitrarily complex distributions and this is the main reason for their widespread application to a diverse range of tasks. In image and video processing, Gaussian Mixture Models have been used for background subtraction [39, 71, 89], image-segmentation [5, 57, 88], and object-detection and classification [18, 27, 45, 46]. In speech and natural language processing, they have been used for speaker identification and verification [64–66], speech recognition [7, 59], and language identification [79, 80]. In robotics, Gaussian Mixture Models have been employed for learning dynamical systems [32, 33], and learning and representation of tasks and policies [8, 9]. GMMs have also been employed for point cloud representation by Eckart et al. [14] with occupancy grids generated via quantization of samples from the distribution. However, to the best of our knowledge, this is the first time GMMs are being employed to enable a probabilistic representation of occupancy via principled incorporation of free space information.

### 3.1.1 Definition

A Gaussian Mixture Model is a parametric probability density function represented as a weighted sum of Gaussian component densities. For this work, we want to learn a model to represent 3D point cloud data. Let a GMM, $\mathcal{G}$, contain $J$ component

Gaussian distributions specified by parameters, $\Theta_j = (\mu_j,\ \Sigma_j,\ \pi_j)$, where $\mu_j$, $\Sigma_j$, and $\pi_j$ represent the mean, covariance, and mixing weight for the $j^{\text{th}}$ component, with $j \in \{1, J\}$. Given a 3D point cloud, $\mathcal{Z}$, of size $N$, with points $z_i \in \mathcal{Z}$ and assuming that the points are i.i.d. samples of the surface being modeled, the likelihood of $\mathcal{Z}$ being generated by $\mathcal{G}$ is

$$p(\mathcal{Z} \mid \Theta) = \prod_{i=1}^{N} p(z_i \mid \Theta) \tag{3.1}$$

$$= \prod_{i=1}^{N} \sum_{j=1}^{J} \pi_j\, p(z_i \mid \mu_j,\ \Sigma_j) \tag{3.2}$$

where

$$p(z_i \mid \mu_j,\ \Sigma_j) = \mathcal{N}(z_i \mid \mu_j,\ \Sigma_j) \tag{3.3}$$

Generally, for numerical stability, the log of the likelihood is used in optimization. The log-likelihood is given as

$$ln\ p(\mathcal{Z} \mid \Theta) = \sum_{i=1}^{N} ln \sum_{j=1}^{J} \pi_j\, p(z_i \mid \mu_j,\ \Sigma_j) \tag{3.4}$$

### 3.1.2 Training

The log-likelihood of the data, $\mathcal{Z}$, given the GMM parameters, $\Theta$, (3.4) produces an analytic gradient that is unsuitable for optimization because as no closed-form solution exists for the minimum. As explained by Eckart et al. [13], a set of $N$ x $J$ binary correspondence variables $C = c_{ij} \in \{0, 1\}$ representing the assignment of each point, $z_i$, to a mixture component, $\Theta_j$ are incorporated to produce a tractable

likelihood function. In other words, each point $z_i$ will have $J$ binary correspondence variables, representing the degree to which a point belongs to a mixture component. The log-likelihood can then be factored as

$$ln\ p(Z, C|\Theta)\ =\ \sum_{i=1}^{N}\sum_{j=1}^{J}\ c_{ij}\{ln\ \pi_j\ +\ ln\ p(z_i|\Theta_j)\} \tag{3.5}$$

It is not feasible to solve the factored form because the value of $C$ is unknown. However, if the correspondence variable $c_{ij}$ was known for each $z_i$ , the joint likelihood could be maximized by setting $\Theta_j$ to the sample mean and sample covariance based on the points for which $c_{ij} \neq 0$. Conversely, if $\Theta$ was known, the probability of a points correspondence with a mixture could be estimated via Bayes' rule

$$p(c_{ij} \mid z_i,\ \Theta_j)\ =\ \frac{p(z_i \mid c_{ij},\ \Theta_j)\ p(c_{ij} \mid \Theta_j)}{p(z_i \mid \Theta)} \tag{3.6}$$

$$=\ \frac{\pi_j\ \mathcal{N}(z_i \mid \Theta_j)}{\sum_{j'}\ \pi_{j'}\ \mathcal{N}(z_i \mid \Theta_{j'})} \tag{3.7}$$

The above equation is a typical instance of Bayes' rule where the probability of $c_{ij}$ to take on a certain value is given by the product of the likelihood of the data point $z_i$ to be represented by a mixture component $\Theta_j$ and the prior probability of $c_{ij}$, normalized by the total probability given all other mixture components. Also, $p(z_i|\Theta)$ can be seen as a marginalization over mixture components of the joint distribution $p(z_i, c_i|\Theta)$ where

$$p(c_{ij} \mid \Theta_j)\ =\ \pi_j$$

and

$$p(z_i \mid \Theta) \;=\; \sum_j p(c_{ij} \mid \Theta_j)\; p(z_i \mid c_{ij},\; \Theta_j)$$

However, since neither the correspondence labels $C$ nor the model parameters $\Theta$ are known, Expectation Maximization (EM) [10] is employed to iteratively estimate both the correspondence variables and the component parameters.

EM has been established as a way to iteratively maximize the joint likelihood of the data and an associated set of latent variables. Two steps are involved in every iteration. The **E-Step** involves calculation of the expected value of the correspondence variables given the current mixture parameters at the $(k+1)^{\text{th}}$-iteration.

$$E[c_{ij}] \;=\; \frac{\pi_j^k\; p(z_i \mid \mu_j^k,\; \Sigma_j^k)}{\sum_{j'=1}^{J}\; \pi_{j'}^k p(z_i \mid \mu_{j'}^k,\; \Sigma_{j'}^k)} \tag{3.8}$$

The **M-Step** involves maximizing the expected log-likelihood with respect to $\Theta$ considering $E[c_{ij}] \stackrel{\text{def}}{=} \gamma_{ij}$ to be a constant.

$$\Theta^{k+1} \;=\; \underset{\Theta}{\text{argmax}} \sum_{ij}\; \gamma_{ij}\; \{ln\; \pi_j \;+\; ln\; p(z_i \mid \Theta_j)\} \tag{3.9}$$

This optimization reduces to an analytic solution for the mixture parameters given

as

$$\mu_j^{k+1} = \frac{\sum_i^N \gamma_{ij}\ z_i}{\sum_i^N \gamma_{ij}} \tag{3.10}$$

$$\Sigma_j^{k+1} = \frac{\sum_i^N \gamma_{ij}\ z_i z_i^T}{\sum_i^N \gamma_{ij}} - \mu_j^{k+1}\ \mu_j^{k+1^T} \tag{3.11}$$

$$\pi_j^{k+1} = \sum_i^N \frac{\gamma_{ij}}{N} \tag{3.12}$$

### 3.1.3 Initialization

The procedure to learn the parameters of a Gaussian Mixture Model via Expectation-Maximization involves a non-trivial initialization step. EM requires the number of Gaussian components and initial values for the parameters for each of the Gaussian components as input. The number of components is crucial to the representation fidelity of the GMM for a given dataset. The approaches generally used to estimate the number of components include priors like the Bayesian Information Criterion [50], Akaike Information Criterion [6] with k-means employed to initialize the parameters for the components. Considering the significance of the size of the mixture model, an information-theoretic approach is developed in this work as a principled initialization strategy for the mixture model.

## 3.2 Information Theoretic Measures

Tools in information-theory enable study of transmission, processing, utilization, and extraction of information. The basic entity, called entropy, relates to the uncertainty associated with a random variable [69]. One of the major goals of this work is to enable reasoning in terms of the information content of the environment instead of the size of the environment. Reasoning in terms of information aligns with the

approach that humans adopt when confronted with a complex environment. Humans tend to reason about their environment in units of information such as a wall, cups, chairs and so on. Approaches to environment representation that are based on a fixed quantization (such as voxel-grids) scale with the size of the environment and are not amenable to reasoning in terms of information. Efforts have been made in literature to compress a voxel-based representation based on the affect of such compression on the information-content of the map [51]. However, the base representation still involves a tessellation of the environment that obfuscates the structural characteristics observed in real-world environments.

The proposed approach leverages information-theoretic measures to estimate the required model complexity to enable a high-fidelity representation of the environment. Specifically, divergence measures are employed to estimate the similarity of Gaussian Mixture Models and Gaussian components with in a mixture model. These measures are reproduced here and used in Sect. 3.3 to enable hierarchy generation.

### 3.2.1 Kullback-Leibler Divergence

Divergence measures seek to provide a measure of distance or dissimilarity between two pdfs. Here, we are interested in the divergence between two Gaussian distributions and between two GMMs. The most well-known form of divergence is the Kullback-Leibler divergence [36]. For a random variable $X$, the Kullback-Leibler divergence, $D_{KL}$, between two distributions, $p(x)$ and $q(x)$, is given by

$$D_{KL}(p||q) \;=\; \sum_{x \in X} p(x) \log_2 \frac{p(x)}{q(x)} \tag{3.13}$$

The closed form solution for Kullback-Leibler divergence between two Gaussian

distributions $f = \mathcal{N}(\mu_f, \Sigma_f)$ and $g = \mathcal{N}(\mu_g, \Sigma_g)$ for $D$-dimensional data is given as

$$\begin{aligned} D_{KL}(f||g) \;=\; & \frac{1}{2}(log\frac{|\Sigma_g|}{|\Sigma_f|} \;+\; trace(\Sigma_g^{-1}\Sigma_f) \\ & +\; (\mu_f \;-\; \mu_g)^T\Sigma_g^{-1}(\mu_f \;-\; \mu_g) \;-\; D) \end{aligned} \tag{3.14}$$

A closed-form approximation for KL Divergence between GMMs has been proposed by Goldberger et al. [22]. For two GMMs, $p$ and $q$, with M and K components respectively and parameters $(\pi_m, \mu_m, \Lambda_m)$ and $(\tau_k, \nu_k, \Omega_k)$, it is given as

$$D_{KL}(q,p) \;\approx\; \sum_{i=1}^{M} \pi_i \min_{j\in\{1,K\}} (D_{KL}(p_i \;||\; q_j) + log\frac{\pi_i}{\tau_j}) \tag{3.15}$$

### 3.2.2 Cauchy-Schwarz Divergence

Cauchy-Schwarz divergence is a non-negative distance metric that takes on a value of zero when its arguments are the same distribution. Unlike Kullback-Leibler divergence, Cauchy-Schwarz divergence is symmetric in its arguments. For a random variable $X$, the Cauchy-Schwarz divergence, $D_{CS}$, between two distributions, $p(x)$ and $q(x)$, is given by

$$D_{CS}(p \;||\; q) \;=\; log\, \frac{\sum_{x\in X}\; p^2(x)\; \sum_{x\in X}\; q^2(x)}{(\sum_{x\in X}\; p(x)\; q(x))^2} \tag{3.16}$$

for the discrete case and

$$D_{CS}(p \;||\; q) \;=\; log\, \frac{\int_x p^2(x)dx\; \int_x q^2(x)dx}{(\int_x\; p(x)q(x)\; dx)^2} \tag{3.17}$$

for the continuous case.

A closed-form solution for CS Divergence between GMMs has been proposed by Kampa et al. [31]. For two GMMs, $p$ and $q$, with M and K components respectively and parameters $(\pi_m, \mu_m, \Lambda_m)$ and $(\tau_k, \nu_k, \Omega_k)$, it is given as

$$
\begin{aligned}
D_{CS}(q,p) = & \\
& - log(\sum_{m=1}^{M}\sum_{k=1}^{K} \pi_m \tau_k z_{mk}) \\
& + \frac{1}{2} log(\sum_{m=1}^{M} \frac{\pi_m^2 |\Lambda_m|^{1/2}, (2\pi)^{D/2}}{} + 2\sum_{m=1}^{M}\sum_{m'<m} \pi_m \pi_{m'} z_{mm'}) \\
& + \frac{1}{2} log(\sum_{k=1}^{K} \frac{\tau_k^2 |\Omega_k|^{1/2}, (2\pi)^{D/2}}{} + 2\sum_{k=1}^{K}\sum_{k'<k} \tau_k \tau_{k'} z_{kk'})
\end{aligned}
\tag{3.18}
$$

where

$$
\begin{aligned}
z_{mk} &= \mathcal{N}(\mu_m \mid \nu_k \ , \ (\Lambda_m^{-1} + \Omega_k^{-1})) \\
z_{mm'} &= \mathcal{N}(\mu_m \mid \mu_{m'} \ , \ (\Lambda_m^{-1} + \Lambda_{m'}^{-1})) \\
z_{kk'} &= \mathcal{N}(\nu_k \mid \nu_{k'} \ , \ (\Omega_k^{-1} + \Omega_{k'}^{-1}))
\end{aligned}
$$

## 3.3 Hierarchical Spatial Model

The proposed model consists of a hierarchy of Gaussian Mixture Models representing 3D space $X \in \mathbb{R}^3$, and the mixture at any given level of the hierarchy differs in size and fidelity from other levels. The GMM size decreases as one moves up the hierarchy and corresponds to a decrease in fidelity of the representation. Expectation-Maximization is employed to initiate model generation and information-theoretic measures are used to estimate the number of components for every level of the hierarchy. Figure 3.1

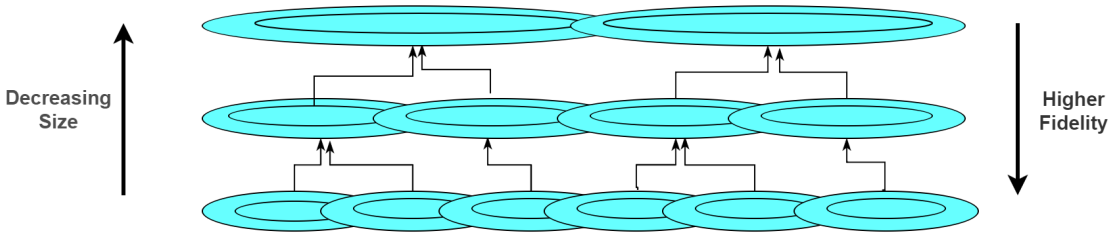



Figure 3.1: The proposed hierarchy of Gaussian Mixture Models. Each level is a sufficient environment model with the lowest level ($l = 0$) providing the highest fidelity representation. The size of the mixtures on moving up in the hierarchy corresponding to a reduction in fidelity. Components at level $l$ are merged (shown by black arrows) to form components for level $l + 1$.

provides an overview of the hierarchy.

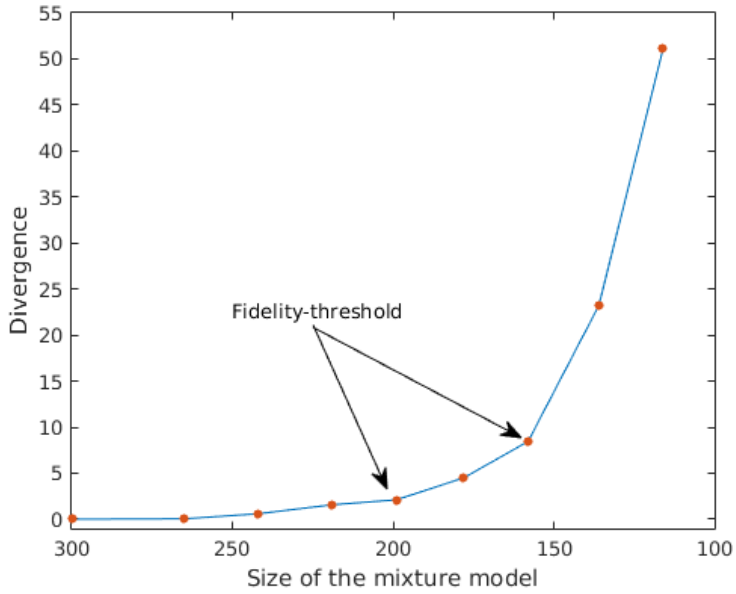
### 3.3.1 Model Definition

Let the $l^{\text{th}}$ level of the hierarchy ($l \in \{1, L\}$) be given by the GMM $\mathcal{G}_l$. Let $\mathcal{G}_l$ contain $J_l$ component Gaussian distributions specified by parameters $\Theta_j = (\mu_j, \Sigma_j, \pi_j)$ where $\mu_j$, $\Sigma_j$, and $\pi_j$ represent the mean, covariance, and mixing weight for the $j^{\text{th}}$ component, with $j \in \{1, J_l\}$. Given a 3D point cloud, $\mathcal{Z}$, of size $N$, with points $z_i \in \mathcal{Z}$ and assuming that the points are i.i.d. samples of the space being modeled, the likelihood of $\mathcal{Z}$ being generated by $\mathcal{G}_l$ is given by (3.1).

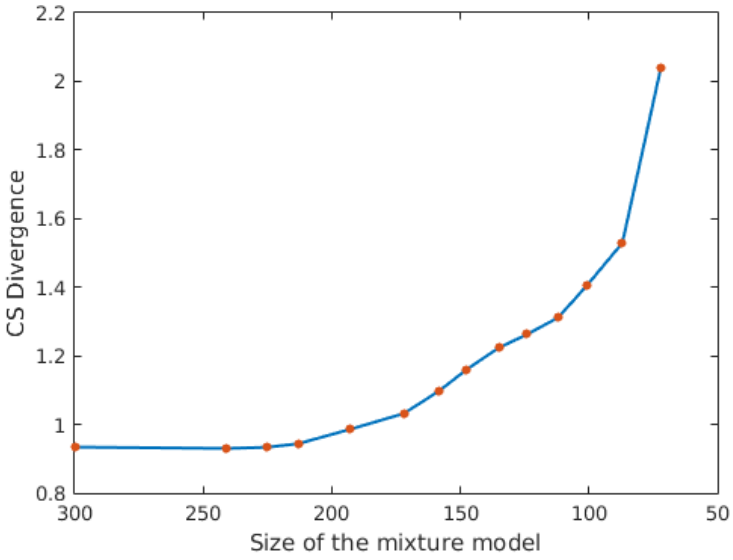
### 3.3.2 Model Generation

#### Estimation of the Fidelity Threshold

A crucial step in the generation of the multi-fidelity model is the estimation of the number of components required to obtain a precise representation. An iterative strategy is proposed to obtain the size of the GMM that best approximates the underlying spatial distribution. A key observation is that there is a threshold on the GMM




(a) Kullback-Leibler Divergence




(b) Cauchy-Schwarz Divergence

Figure 3.2: Variation of KL-Divergence (a) and CS-Divergence(b) for GMMs of size varying from 300 to 116 with respect to the largest GMM of size 300. The possible fidelity thresholds are highlighted. Increasing the size of the GMM beyond these thresholds does not significantly affect the fidelity of representation as indicated by the small decrease in divergence.

size (hereafter referred to as fidelity threshold, $\lambda_f$) beyond which the model fidelity does not vary significantly even if more components are added. To leverage this observation, a measure to quantify variation in model fidelity is required. The divergence between Gaussian Mixture pdfs serves as the measure of relative model fidelity and is demonstrated in Fig. 3.2 that shows the variation in KL-divergence and CS-Divergence between a reference GMM and reduced size GMMs trained via EM on the same dataset. The knee point is distinctly visible in the plot for KL-Divergence which motivates using KL-Divergence for fidelity-threshold estimation in this work. The presence of a distinct knee-point enables an iterative bottom-up approach that, when initialized with a GMM of size greater than $\lambda_f$, generates GMMs of reduced size until the fidelity threshold is obtained. The GMM of size equal to the fidelity threshold forms the lowermost (highest-fidelity) layer of the hierarchy. The iterative algorithm can be naturally extended to generate more levels in the hierarchy of reduced-fidelity with the difference quantified via KL-Divergence.

**Iterative Hierarchy Generation**

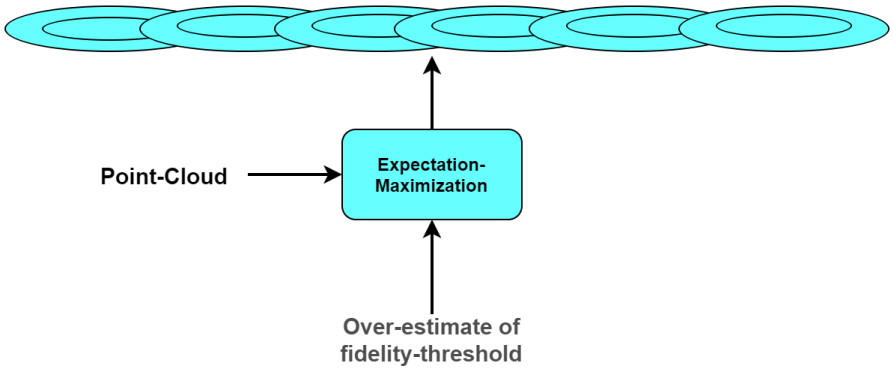
Algorithm 1 and Figs. 3.3 and 3.4 outline the proposed bottom-up approach to generate a hierarchy of Gaussian Mixtures forming a multi-fidelity environment representation. Expectation-Maximization is employed to initialize the algorithm via parameter estimation for the initial reference GMM (Line 4, Fig. 3.3a). The size of the reference GMM needs to be greater than the fidelity threshold for correct estimation of the $\lambda_f$. This size is referred to as the overestimate of fidelity-threshold, ($\lambda_{fo}$), and is provided as a parameter to Algorithm 1. At every step of the iteration (Lines 6-16), a reduced size GMM, $\mathcal{M}$, is generated given the most recent GMM in the hierarchy,

**Algorithm 1:** HGMM Generation

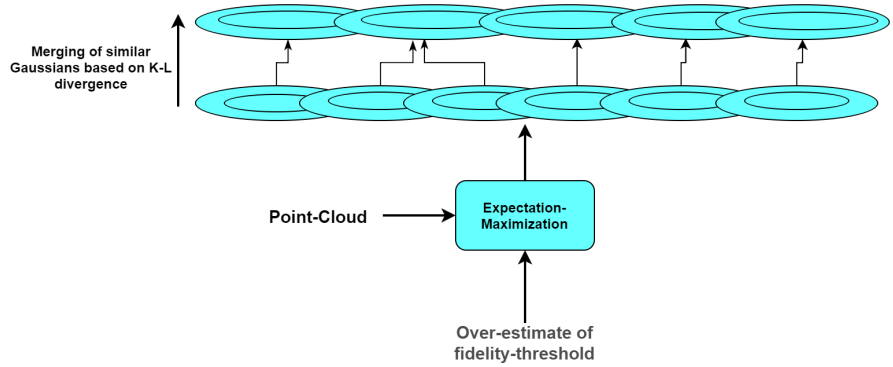
**Result:** HGMM $\mathcal{G}$

$\lambda_d$, $\lambda_{fo}$, $L$, $\mathcal{Z} \leftarrow Input$;
$\mathcal{G} \leftarrow \{\emptyset\}$;
$div \leftarrow 0$, $l \leftarrow 0$;
$\mathcal{G} \leftarrow EM(\mathcal{Z}, \lambda_{fo})$;
**while** *true* **do**
    $\mathcal{M} \leftarrow \{\emptyset\}$;
    **for** $i \leftarrow 1...|\mathcal{G}_l|$ **do**
        $\theta_i, w_i \leftarrow \mathcal{L}_{l,i}$;
        **for** $j \leftarrow i{+}1...|\mathcal{G}_l|$ **do**
            $\theta_j \leftarrow \mathcal{G}_{l,j}$;
            **if** $KLDivergence(\theta_i, \theta_j) < \lambda_d$ **then**
                $\{\theta_i\} \leftarrow Merge(\theta_i, \theta_j)$;
            **end**
        **end**
        $\mathcal{M} \leftarrow \mathcal{M} \cup \{\theta_i\}$;
    **end**
    $\mathcal{G} \leftarrow \mathcal{G} \cup \mathcal{M}$;
    div $\leftarrow KLDivergence(\mathcal{G}_l, \mathcal{G}_0)$;
    **if** $\neg Pruned\ AND\ IsKneePoint(div)$ **then**
        $\lambda_f \leftarrow |\mathcal{G}_{l-1}|$;
        $Prune(\mathcal{G}, \lambda_f)$;
    **end**
    **if** $|\mathcal{G}_l| < L$ **then**
        break;
    **end**
**end**

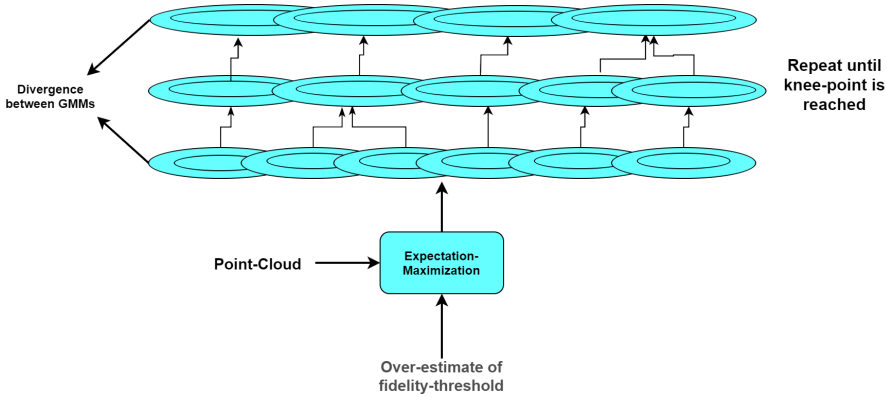
$\mathcal{G}_l$ (Fig. 3.3b). This is achieved by merging *similar* Gaussian components where the measure of similarity is given by Kullback-Leibler (KL) Divergence between Gaus-




(a) Initialization

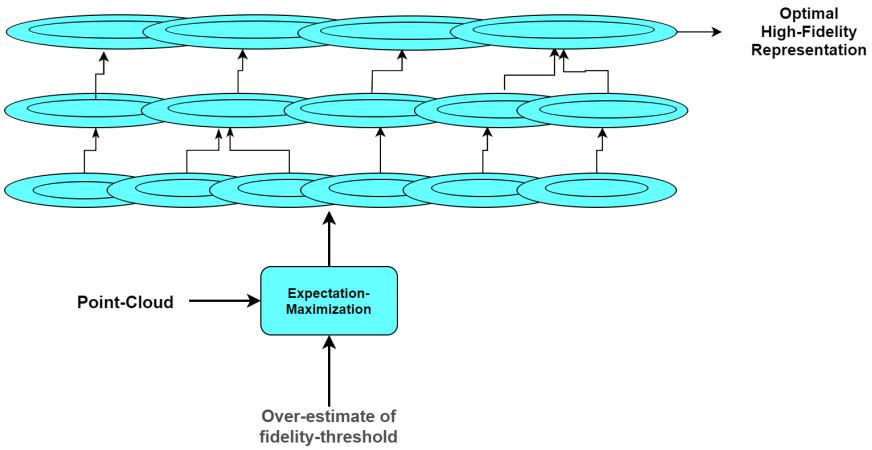



(b) Merging of Gaussian Components

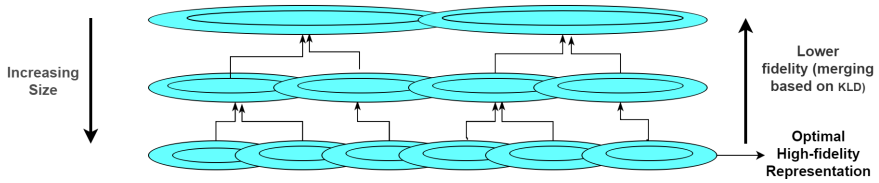



(c) Divergence calculation

Figure 3.3: Iterative Hierarchy Generation. The algorithm is initialized via Expectation Maximization (a) followed by merging of components to generate reduced-size GMMs (b) and calculation of KL-Divergence to estimate fidelity-threshold (c).




(a) Obtaining the Lowest Level




(b) Lesser-Fidelity GMM Generation

Figure 3.4: Iterative Hierarchy Generation. The GMM corresponding to the fidelity threshold is the high-fidelity representation (a) and forms the lowest level in the hierarchy (b). The lesser fidelity levels are generated by continuing the merging-based procedure.

sian distributions (3.14) and similarity itself is based on KL Divergence being less than the similarity threshold, $\lambda_d$. KL Divergence between the current level, $\mathcal{G}_l$, and the bottom most level, $\mathcal{G}_0$, (Line 18) is used to estimate whether the knee-point and thus the fidelity-threshold has been reached (Lines 19-20, Figs. 3.3c and 3.4a). Once estimated, all levels of the hierarchy with size more than $\lambda_f$ are pruned (Line 21) and the GMM with size $\lambda_f$ forms the bottom-most level of the hierarchy (Fig. 3.4b). The algorithm terminates when the desired number of hierarchy levels ($L$) have been generated (Line 23).

### 3.3.3 Model Update

A consequence of the sequential motion of mobile robots is that a portion of the point cloud, $\mathcal{Z}$, obtained at any instant, contains information that has already been modeled and the remaining portion contains novel information. The novel and redundant portions of $\mathcal{Z}$ are estimated via calculation of the log-likelihood of the points, $z_i \in \mathcal{Z}$, to be generated by the existing model, $\mathcal{G}_{l=0}$ (3.4). Inspired by the work of Engel et al. [16], an empirically determined novelty threshold, $\lambda_n$, is used to categorize points as novel ($\mathcal{Z}_n$) versus redundant ($\mathcal{Z}_r$).

The update of $\mathcal{G}_{l=0}$ with $Z_r$ proceeds via boot-strapping of EM with the already learned parameters $\Theta$. *M-step* is slightly modified to incorporate the posterior from the previous point clouds as well as $\mathcal{Z}_r$. Let the support size of $\mathcal{G}_{l=0}$ be $N$. Then, following entities are defined for $\{\Theta_j = (\mu_j,\ \Sigma_j, \pi_j)\}$,

$$
\begin{aligned}
S_{\pi_j} &= \sum_{i}^{N} \gamma_{ij} \ = \ N\pi_j \\
S_{\mu_j} &= \sum_{i}^{N} \gamma_{ij}\ z_i \ = \ S_{\pi_j}\mu_j \\
S_{\Sigma_j} &= \sum_{i}^{N} \gamma_{ij}\ z_i z_i^T \ = \ S_{\pi_j}(\Sigma_j \ + \ \mu_j\ \mu_j^T)
\end{aligned}
$$

The updated mean, covariance and weights, given $|\mathcal{Z}_r| = N'$ are

$$S'_{\pi_j} \ = \ S_{\pi_j} \ + \ \sum_{i}^{N'} p_{ij} \tag{3.19}$$

$$\pi'_j \ = \ \frac{S'_{\pi_j}}{N + N'} \tag{3.20}$$

$$\mu'_j \ = \ \frac{S_{\mu_j} \ + \ \sum_i^{N'} p_{ij}\ z_i}{S'_{\pi_j}} \tag{3.21}$$

$$\Sigma'_j \ = \ \frac{(S_{\Sigma_j} \ + \sum_i^{N'} p_{ij}\ z_i z_i^T)}{S'_{\pi_j}} \ - \ \mu'_j\ \mu_j'^T \tag{3.22}$$

Higher levels of the hierarchy are re-generated based on the procedure outlined in Algorithm 1 (Lines 6-16).

A fresh HGMM, $\mathcal{G}'$, is learned for the novel point cloud, $\mathcal{Z}_n$, following the same procedure as outlined in Algorithm 1. The levels of the existing HGMM, $\mathcal{G}$, are then augmented with the components of the corresponding levels of $\mathcal{G}'$ followed by weight normalization. The required insight is that the points in $\mathcal{Z}_n$ are minimally influenced by the components in $\mathcal{G}$ as evidenced by the log-likelihood based novelty check. Thus, a naive augmentation closely approximates the distribution that would be learned if trained as a whole. The updated weight vector, $\pi_{\mathcal{G}_l}$, for $\mathcal{G}_l$ with a support set of size $N_{\mathcal{G}_l}$ is

$$\pi_{\mathcal{G}_l} = \frac{\pi_{\mathcal{G}_l} N_{\mathcal{G}_l}}{N_{\mathcal{G}_l} + N_{\mathcal{G}'_l}} \tag{3.23}$$

where $N_{\mathcal{G}'_l}$ is the support-set size of the $\mathcal{G}'_l$.

The computational cost of calculating the log-likelihood for the purpose of novelty detection, grows with the size of the model. To ensure real-time nature of the updates, a relatively small sub-model of $\mathcal{G}_{l=0}$ is maintained and used for novelty-check. Let the sub-model be called the local model $\mathcal{L}$. The local model is generated by discarding the components from $\mathcal{G}$ that have a negligible contribution to the log-likelihood for the current point cloud $\mathcal{Z}$. Thus, only the components of $\mathcal{G}$ that have a non-negligible maximum pdf value $\mathcal{N}(z_i \,|\Theta_j)$ over $z_i \in \mathcal{Z}$ in (3.4) are retained in $\mathcal{L}$.

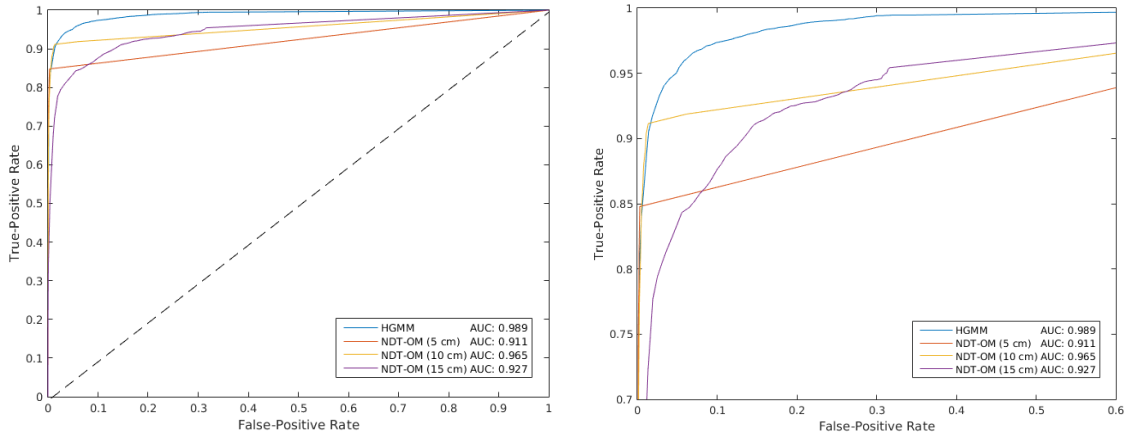
### 3.3.4 A Note on Parameters

The similarity threshold, $\lambda_d$, regulates the rate of merging of Gaussian components to form higher layers of the hierarchy. A higher value of $\lambda_d$ causes more components to be merged at each level thereby increasing the difference in fidelity between subsequent levels. The divergence between Gaussian components tends to increase as the number of GMM components used to represent the distribution decreases. Thus, $\lambda_d$ is incremented as the levels of the hierarchy are generated. It is, however, important to note that the parameter needs to be tuned only once. The same values for $\lambda_d$ are observed to hold across all datasets on which the proposed approach is evaluated. The overestimate of fidelity-threshold, $\lambda_{fo}$ affects the accuracy of the model if it is not a strict overestimate. Conversely, a very large value affects computational complexity. The strategy used here involves applying a voxel-grid filter to the point cloud. The size of the filtered point cloud is indicative of the overestimate for $\lambda_f$.

## 3.4 Evaluation

A quantitative and qualitative evaluation of the fidelity of the proposed spatial model is presented in this section. Also, a comparison to an implementation[1] of the Normal Distribution Transform surface model (Sect. 2.2.2) in terms of fidelity is provided. Two datasets have been used for this purpose. The first is a point cloud dataset (**D1**) collected using an Asus Xtion Pro RGB-D sensor. The dataset represents an environment exhibiting varying levels of clutter with low-texture walls in the background and numerous objects in the foreground. The data-set contains 186 point clouds with

[1]NDT software. `https://github.com/OrebroUniversity/perception\_oru-release` [Accessed on 28$^{\text{th}}$ June, 2017]




(a) (b)

Figure 3.5: Receiver Operating Characteristic (ROC) Curves for the highest-fidelity level of the HGMM spatial model and NDT representation (cell-size 5 cm, 10 cm and 15 cm) for **D2** dataset (a) and the zoomed-in view of the relevant region of the curve (b) . The proposed approach is observed to have a higher True-Positive rate and Area Under the Curve (AUC) than NDT. The NDT representation is also observed to be sensitive to cell-size.

odometry obtained from a motion-capture system. Figure 3.8a provides a representative snapshot of the dataset. The second dataset (**D2**) is a publicly available point cloud dataset from University of Freiburg [73]. The dataset has 86 point clouds and odometry at 1 Hz and captures the insides of a cluttered room (Fig. 3.6a).

### 3.4.1 Fidelity of the Spatial Model

The proposed model is trained on a sequence of subsampled point clouds from **D2** and the highest-fidelity level is queried for the likelihood of the test-points to be represented by the model. The test-set consists of the set of points not made available to the algorithm during training and free-space points sampled along the rays corresponding to the test-set. The performance is compared to the Normal Distribution

(a) Input point cloud data

(b) Reconstruction via sampling

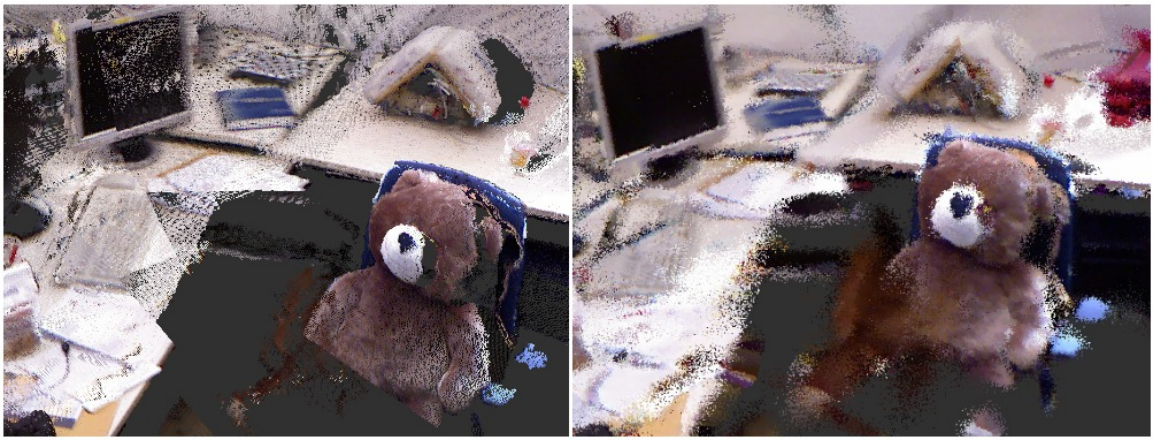
Figure 3.6: The cumulative point cloud of the cluttered room environment from University of **D2** (a) and the reconstructed point cloud (b) obtained by sampling from the GMM with an average size of 116 components forming the lowest level of the HGMM. RGB information is for illustration only and obtained via nearest neighbor association.

Transform [42] surface model (cell-size 5 cm, 10 cm and 15 cm) via Receiver Operating Characteristic curves shown in Fig. 3.5 and generated by using the log-likelihood value as a threshold to turn the generative model into a classifier. The proposed approach is observed to achieve a higher true positive rate and Area Under the Curve than the NDT representation. This can be attributed to the restricted support of the Gaussian distributions due to the cell independence assumption in NDT that leads to higher uncertainties at cell boundaries. Also, the accuracy of the NDT representation is observed to be sensitive to the resolution of the voxel grid.

Figures 3.6 and 3.7 provide a qualitative evaluation of the spatial model. Point cloud data from **D2** is used to train the proposed hierarchical model. The point cloud is then reconstructed via sampling from the highest-fidelity level of the surface-model. Figure 3.7 shows the point cloud from the dataset and the high-fidelity reconstruction via sampling. Snapshots of the same dataset are shown in Fig. 3.7 to highlight the accuracy of the surface model.

### 3.4.2 Metric map from continuous distribution

The proposed technique can be considered an arbitrary resolution representation of the environment. It is, thus, possible to generate any desired resolution representation via sampling from the surface model. This sections shows an occupancy grid generated from the HGMM model and compares it to the occupancy grids generated from competing techniques. To generate an occupancy grid from a continuous belief distribution, samples are drawn and binned into voxels of the desired size. Sampling from each component ensures coverage of the occupied space and as no points are drawn from free space, a relatively small set of points needs to be sampled to generate

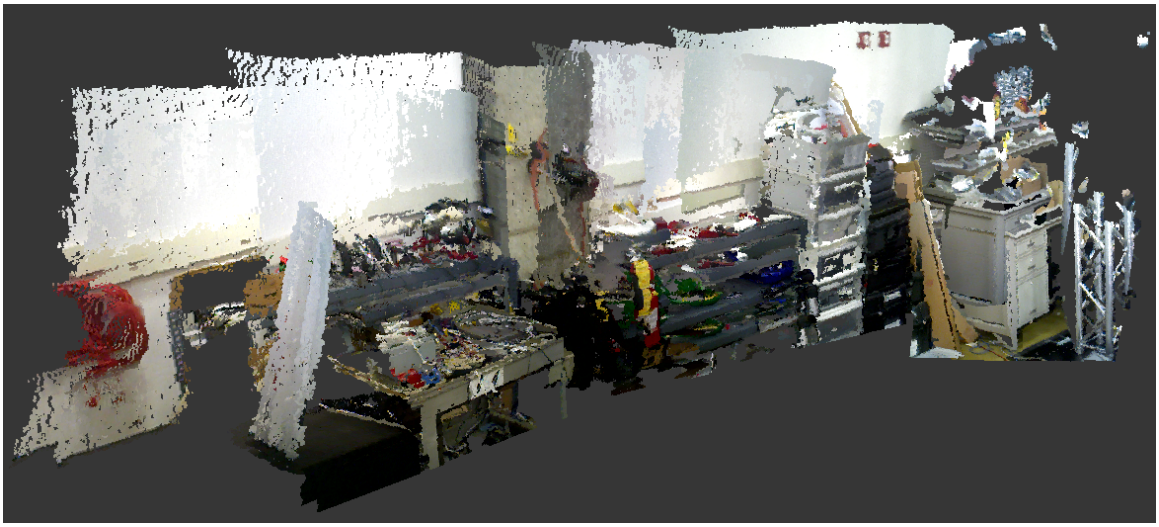


(a)

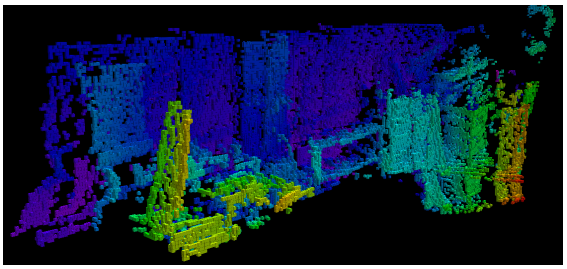
(b)

Figure 3.7: Qualitative visual evaluation for the spatial model. (a,b) A snapshot from the point cloud dataset **D2** (left) and the corresponding reconstruction via sampling from the lowest level of the hierarchical model (right). With an average size of 116 components per point cloud, the HGMM based model provides a high-fidelity reconstruction and is also able to handle sparsity in sensor data (b). RGB information is for illustration only and obtained via nearest neighbor association.

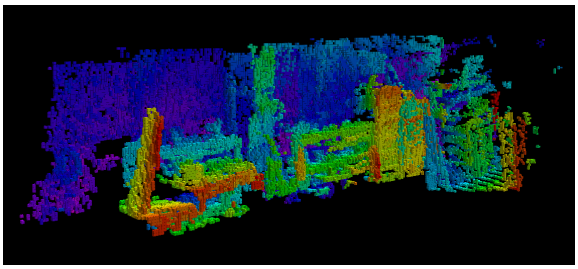
the occupancy grid. Once the points have been sampled, they are binned into voxels of the desired size.



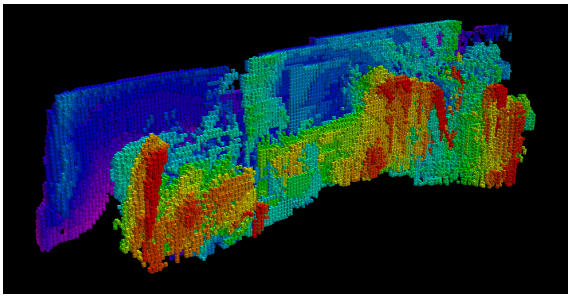
(a) Input point cloud



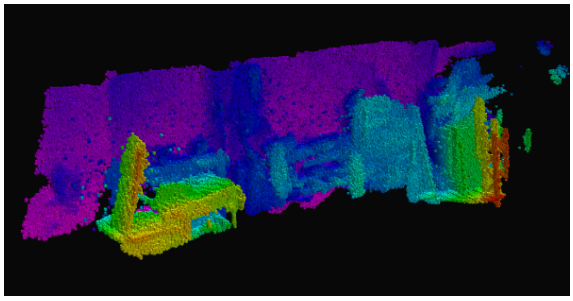
(b) Grid



(c) NDT



(d) GPmap



(e) HGMM

Figure 3.8: The occupancy map at 5 cm resolution for dataset **D1** generated using an occupancy grid (b), NDT-OM (c), GPmap (d) and HGMM (e). The gaps in the naive grid and NDT-OM approach are clearly visible. GPmap and HGMM produce dense occupancy grids due to the continuous distribution learned.

Figure 3.8 shows the occupancy map at 5 cm resolution generated by a grid, NDT-OM, an implementation of GPmap [34] and the proposed approach. The map

for NDT-OM is generated by querying the model for likelihood and categorizing based on a threshold of $1e^{-5}$. The occupancy grid for GPmap is obtained by regressing from test-points defined at a resolution of 5 cm.

## 3.5 Summary

A multi-fidelity spatial model is developed in this chapter via a hierarchy of Gaussian Mixture Models (GMMs). The hierarchy consists of a GMM per level and each GMM differs from the others in terms of the number of components and the fidelity of representation. A methodology, governed by information-theoretic principles, to estimate the required number of components in the levels of the hierarchy is presented. The distinct knee-point in the growth of divergence when the size of the GMM is reduced is leveraged to obtain the fidelity threshold, $\lambda_f$. An iterative algorithm to generate the level $l$ of the hierarchy by merging similar components in the level $l-1$ is presented as a principled strategy to generate an information hierarchy.

The proposed surface model is also compared in terms of fidelity to Normal Distribution Transform (NDT) surface representation and shown to be more accurate with less manual tuning. A qualitative evaluation is done to demonstrate the fidelity of the representation via reconstruction of input point cloud data leveraging the generative properties of the model.

# Chapter 4

# Probabilistic Representation of Occupancy

The spatial model presented in Chapter 3 does not support queries for the probability of occupancy at a given location. The model lacks free-space information and therefore, does not suffice to be a probabilistic representation of occupancy. In other words, the spatial model is a generative representation of sensor data and a discriminative model of occupancy is required to enable queries for occupancy probability.

This chapter extends the spatial model into a probabilistic representation of occupancy. The 3D HGMM is extended to a 4-tuple model to enable a conditional probability distribution over occupancy. An explicit free-space model based on GMMs is developed to incorporate free-space information into the model. Further, an uncertainty measure in the form of a variance estimate, associated with model predictions, is incorporated to enable online inference with respect to the model.

## 4.1 Augmented Spatial Model

### 4.1.1 Distance Function

The 3D spatial model is augmented to a 4-tuple HGMM to enable reasoning over occupancy. The additional variable $W$ is a function of the 3D point cloud $\mathcal{Z}$ and the sensor pose $P$ and is called the distance function. The distance function maps every point on the surface to zero and points outward from the surface to a distance from the surface along a ray emanating from the sensor. Given a point on the surface, $Q$, the distance for a point in free-space, $F$, along the ray $\vec{QP}$ is $||\vec{QF}||$. Thus, $W$ is zero for all points on the surface and positive for the points in free-space.

The additional variable $W$ enables expression of occupancy as a conditional distribution over spatial location. Specifically, considering occupancy as a binary variable *occ*, the probability of occupancy at a 3D location $X = x$ is

$$P(occ = 1 \mid X = x) \;=\; P(W = 0 \mid X = x) \tag{4.1}$$

and similarly the probability of free-space at a location $X = x$ is

$$P(occ = 0 \mid X = x) \;=\; P(W > 0 \mid X = x) \tag{4.2}$$

### 4.1.2 The Conditional PDF

The conditional pdf for $W$ over $X$ is derived as follows based on the work of Sung [74]. The joint density, $p_{X,W}$, can be represented as a GMM with $J$ components specified by parameters $\Theta_j = (\pi_j,\ \mu_j,\ \Sigma_j)$ where $\pi_j$ , $\mu_j$ and $\Sigma_j$ represent the mixing weight,

mean and covariance matrix as

$$p_{X,W}(x,w) \;=\; \sum_{j=1}^{J} \pi_j \phi(x,w;\mu_j,\Sigma_j) \tag{4.3}$$

where

$$\sum_{j=1}^{J} \pi_j = 1, \quad \mu_j = \begin{bmatrix} \mu_{jX} \\ \mu_{jW} \end{bmatrix}, \quad \Sigma_j = \begin{bmatrix} \Sigma_{jXX} & \Sigma_{jXW} \\ \Sigma_{jWX} & \Sigma_{jWW} \end{bmatrix},$$

and $\phi(x,w;\ \mu_j,\Sigma_j)$ is the 4-tuple Gaussian distribution $\mathcal{N}(x,w;\ \mu_j,\Sigma_j)$ representing the pdf of the $j^{\text{th}}$ component. The joint density can be decomposed as follows by partitioning each Gaussian component as proposed by Mardia et al. [44],

$$\begin{aligned} & p_{X,W}(x,w) \\ &\quad = \; p_{W|X}(w|x)\; p_X(x) \\ &\quad = \; \sum_{j=1}^{J} \pi_j \phi(w|x; m_j(x), \sigma_j^2)\; \phi(x;\mu_{jX},\Sigma_{jXX}) \end{aligned} \tag{4.4}$$

where

$$m_j(x) \;=\; \mu_{jW} \;+\; \Sigma_{jWX}\; \Sigma_{jXX}^{-1}(x-\mu_{jX}) \tag{4.5}$$

$$\sigma_j^2 \;=\; \Sigma_{jWW} - \Sigma_{jWX}\; \Sigma_{jXX}^{-1}\; \Sigma_{jXW} \tag{4.6}$$

The marginal density of X is obtained from (4.4) as

$$p_X(x) \;=\; \int p_{X,W}(x,w)\; dw = \; \sum_{j=1}^{J} \pi_j \phi(x;\mu_{jX},\Sigma_{jX}) \tag{4.7}$$

The conditional density $p_{W|X}(w|x)$ follows from (4.4)

$$p_{W|X}(w|x) \;=\; \sum_{j=1}^{J} w_j(x)\; \phi(w; m_j(x), \sigma_j^2), \tag{4.8}$$

with the mixing weight

$$w_j(x) \;=\; \frac{\pi_j \phi(x; \mu_{jX}, \Sigma_{jXX})}{\sum_{k=1}^{K} \pi_k \phi(x; \mu_{kX}, \Sigma_{kXX})} \tag{4.9}$$

### 4.1.3 Model Training

The generation of the augmented HGMM follows essentially the same procedure as outline in Algorithm 1. However, a small perturbation is required to the input point cloud, $\mathcal{Z}$. By definition, the value of the distance function is zero for all points $z_i$ in $\mathcal{Z}$. The point cloud, thus, needs to be augmented with points in free-space to enable the model to learn the correlation between $X$ and $W$. Typically, only a small set of points in free-space are required to learn this correlation. The free-space points are generated at a small distance $\epsilon$ (typically 1 $mm$) from the points in $\mathcal{Z}$.

As a consequence of the addition of free-space points, the distribution over the input point cloud is slightly altered. The vanilla spatial model corresponds to the $W \;=\; 0$ and is given as

$$p_{X|W}(x|0) \;= \sum_{j=1}^{J} \; w_j(0) \phi(m_j(0), \Sigma_j) \tag{4.10}$$

where

$$m_j(W=0) \;=\; \mu_{jX} \;+\; \Sigma_{jXW}\,\Sigma_{jWW}^{-1}(0-\mu_{jW}) \tag{4.11}$$

$$\Sigma_j \;=\; \Sigma_{jXX} - \Sigma_{jXW}\,\Sigma_{jWW}^{-1}\,\Sigma_{jWX} \tag{4.12}$$

$$w_j(W=0) \;=\; \frac{\pi_j \phi(0;\mu_{jW},\Sigma_{jWW})}{\sum_{j'=1}^{J}\pi_{j'}\phi(0;\mu_{j'W},\Sigma_{j'WW})} \tag{4.13}$$

## 4.2 Free-Space Model

Free-space, in the context of a mobile robot, implies the space present between the sensor and the observed surfaces in the environment. This space is indirectly observed by sensor rays passing through it. It is bounded on one side by the observed point cloud $\mathcal{Z}$ and on the other side by the sensor. The free-space is not structurally complex and this fact can be leveraged to obtain an efficient representation.

### 4.2.1 Model Training

Structural-sparsity in free-space is leveraged to develop a constant time algorithm for learning a 4-tuple HGMM $\mathcal{F}$ to model indirectly observed free-space. The surface model $\mathcal{G}$ is used as a prior for the size of the free-space model. Specifically, for every Gaussian component in the GMM $\mathcal{G}_l$, there is a corresponding component in $\mathcal{F}_l$.

The parameter estimation of the $j^{\text{th}}$ component of $\mathcal{F}_l$ proceeds by sampling a set of points, $S_j$, from the $j^{\text{th}}$ component of $\mathcal{G}_l$. The set, $S_j$, together with the sensor-pose, $P$, are used to sample points in free-space, at a fixed resolution, along the rays originating at the $P$ and terminating at the points in $S_j$. The values for $W$ are obtained by calculating the distance function from the sampled points and the

sensor-pose. The mean and covariance matrix for the $j^{\text{th}}$ component in $\mathcal{F}_l$ are then estimated from the set of sampled free-space points. The weight of the $j^{\text{th}}$ component is set equal to the weight of the corresponding component in $\mathcal{G}_l$.

The procedure outlined above is repeated for every component in $\mathcal{G}_l$ and for every layer in $\mathcal{G}$ resulting in the generation of the free-space HGMM $\mathcal{F}$. Clearly, the value of distance-function for the free-space model is always positive and varies up to the maximum distance of the sensor from the surface.

## 4.3 Unified Model

A unified model of occupancy is obtained via assimilation of information from both occupied space and free space. One strategy to obtain a unified representation would be to generate a unified dataset consisting of the point cloud $\mathcal{Z}$ and the free-space points sampled based on $z_i$ and training the proposed HGMM on the unified dataset. However, this approach is computationally expensive as the required number of components and the size of the dataset is higher. Also, this approach affects the accuracy of the spatial model as it gets diluted by the free-space model and the degree of this effect depends on the density of free-space observations. Considering these challenges, an approach is proposed that retains the high-fidelity generative model while enabling probabilistic representation of occupancy.

The 4-tuple spatial model, $\mathcal{G}$, (Sect. 4.1) and the free-space model, $\mathcal{F}$, (Sect. 4.2) are merged to form a unified model of occupancy, $\mathcal{H}$, for the environment. The merging proceeds via weight normalization. The updated weight vector, $\pi_{\mathcal{G}_l}$, for the $l^{\text{th}}$ layer of the surface model, $\mathcal{G}_l$, with a support set of size $N_{\mathcal{G}_l}$ is

$$\pi_{\mathcal{G}_l} \;=\; \frac{\pi_{\mathcal{G}_l} N_{\mathcal{G}_l}}{N_{\mathcal{G}_l} \;+\; N_{\mathcal{F}_l}} \tag{4.14}$$

where $N_{\mathcal{F}_l}$ is the support-set size of the $\mathcal{F}_l$. This strategy is preferred over training the hierarchy directly on a set containing hit-points and sampled free-space points in order to cap the computational complexity and retain the high-fidelity surface model. Given the unified occupancy model for the environment, the probability of occupancy at any location $x$ is expressed as

$$\begin{aligned} P(occ = 1 \mid X = x) \;&=\; P(W = 0 \mid X = x) \\ &= \int_{W=-\epsilon}^{W=\epsilon} p_{W|X}(w|x)\; dw \\ &= \int_{W=-\epsilon}^{W=\epsilon} \sum_{j=1}^{J} w_j(x)\; \phi(w; m_j(x), \sigma_j^2) \end{aligned} \tag{4.15}$$

where $\epsilon$ is a small integration interval. The probability of a location to be in free-space (4.2) is expressed as

$$\begin{aligned} P(occ = 0 \mid X = x) \;&=\; P(W > 0 \mid X = x) \\ &= \int_{W=\epsilon}^{W=\infty} p_{W|X}(w|x)\; dw \\ &= \int_{W=\epsilon}^{W=\infty} \sum_{j=1}^{J} w_j(x)\; \phi(w; m_j(x), \sigma_j^2) \end{aligned} \tag{4.16}$$

### 4.3.1 Variance Estimate

A mean function and a variance estimate is regressed from the unified occupancy distribution, based on the work of Sung [74]. A mean function is obtained from (4.5) and (4.9) as the weighted average of component-wise means.

$$m(x) \;=\; E[W|X=x] \;=\; \sum_{j=1}^{J} w_j(x)\; m_j(x) \tag{4.17}$$

A variance estimate associated with the regressed mean is obtained from (4.17) as

$$\begin{aligned} v(x) = \;& E[(W|X=x)^2] - E[W|X=x]^2 \\ =\; & \sum_{j=1}^{J} w_j(x)(m_j(x)^2 + \sigma_j^2) - (\sum_{j=1}^{J} w_j(x) m_j(x))^2 \end{aligned} \tag{4.18}$$

The variance estimate forms a measure of uncertainty associated with model predictions that enables informed planning for the purpose of active perception.

## 4.4 Results and Analysis

The proposed approach is evaluated in this section to assess the fidelity of the occupancy representation along with the associated memory-footprint. The correctness of the variance estimate is investigated and the real-time viability of the proposed formulation on a computationally-constrained processor is assessed. Three datasets are used for this evaluation. The first dataset **FR_ROOM** is publicly available [73] and is collected using an RGBD sensor. The dataset represents a small-scale cluttered environment where the measured depth ranges up to 3.5 m. The second dataset

**MINE** represents a larger-scale environment. This dataset is collected using a Velodyne VLP-32 LIDAR in an underground mine with the average depth ranging from 8 m to 9 m. The dataset extends over 1 km in length and contains significantly less structural detail. The third dataset **PIT** is collected using a Velodyne VLP-16 LIDAR in an open pit and represents an unstructured environment with the average measured depth ranging from 15 m to 17 m.

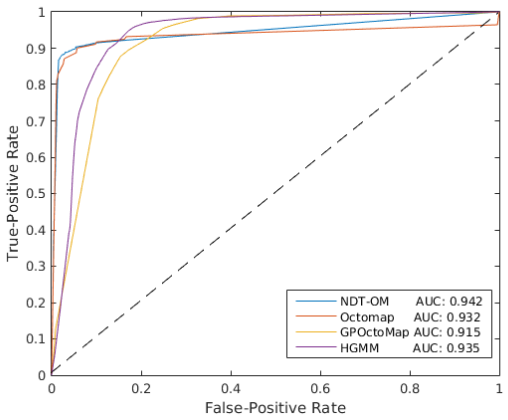
The proposed framework was implemented in C++ using the Robot Operating System (ROS) framework [60] and leveraging the ArrayFire library [87] for a Graphics Processing Unit (GPU)-based parallelized implementation. A comparison to the implementations of GPOctoMap [85], NDT-OM[1] [68] and Octomap[2] [28] in terms of fidelity, memory-footprint and generalizability is also provided. It is important to note that the same set of parameters are used for the HGMM approach for all the experiments reported here. The novelty threshold is set to $-10.5$ and the similarity threshold is initialized to 0.9 and iteratively incremented by 0.2 (Sect. 3.3.4). The parameters for the competing techniques are tuned per dataset and stated when required in the following sub-sections.

### 4.4.1 Fidelity of the Occupancy Representation

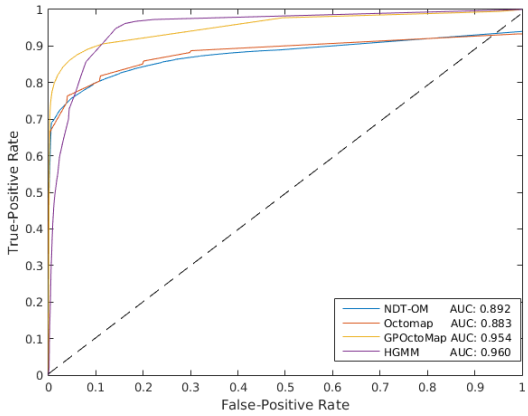
The accuracy of the unified occupancy model, $\mathcal{H}$, is characterized in diverse environments represented by the three datasets (**FR_ROOM, MINE, PIT**) and compared to Octomap, NDT-OM and GPOctoMap. Figure 4.1 presents the ROC curves for the three datasets. For **FR_ROOM**, the cell-size for Octomap is set to 5 cm and

[1]NDT-OM software. `https://github.com/OrebroUniversity/perception\_oru-release` [Accessed on 28$^{\text{th}}$ June, 2017]

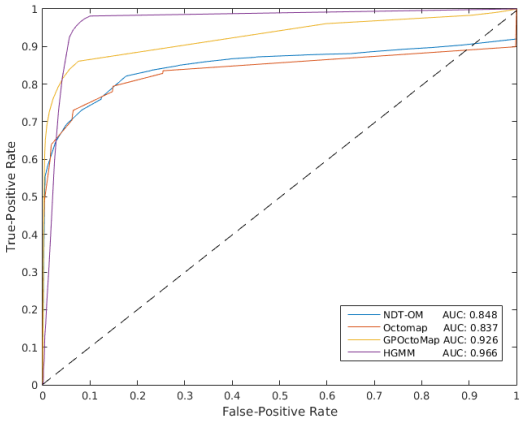
[2]Octomap software. `https://github.com/OctoMap/octomap` [Accessed on 28$^{\text{th}}$ June, 2017]




(a) FR_ROOM




(b) MINE




(c) PIT

Figure 4.1: Receiver Operating Characteristic curves for the proposed probabilistic occupancy representation (HGMM) and GPOctoMap, NDT-OM and Octomap. The HGMM approach is observed to maintain level of accuracy across all three datasets, **FR_ROOM** (a), **MINE** (b) and **PIT** (c), while the competing techniques appear to be sensitive to environmental traits and sensor characteristics.

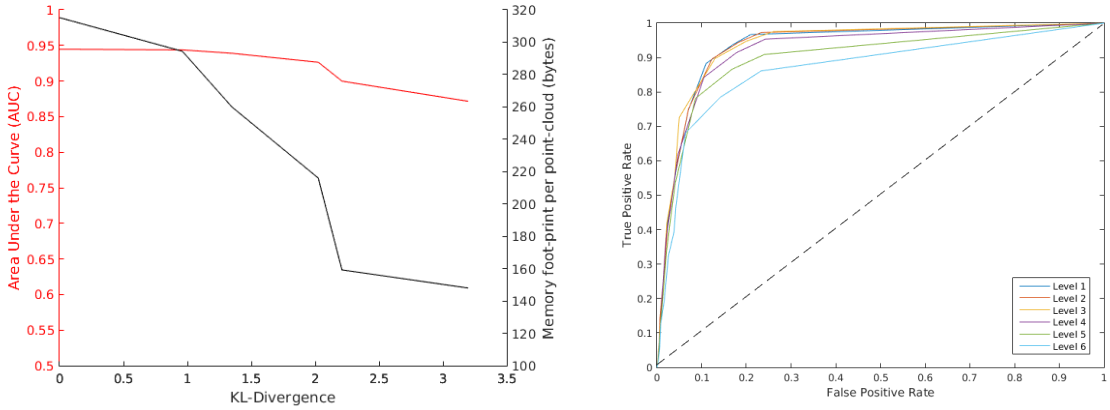
that of NDT-OM is set to 10 cm. The hyper-parameter training for GPOctoMap is done on a subsampled version of the data. The characteristic length and $\sigma_f$ are 0.1 m and 0.5 respectively. It is observed (Fig. 4.1a) that the proposed approach matches the performance of NDT-OM and Octomap in terms of AUC measure, but has a higher true-positive rate. The continuous nature of the HGMM approach makes it

more robust to sensor sparsity leading to more correct classifications. GPOctoMap is observed to have the highest false-positive rate. This is indicative of the fact that the fixed characteristic length used for GP Regression affects generalization of the approach to the whole environment. The HGMM approach has a smaller false positive rate as it is not restricted by a fixed characteristic length.

For **MINE**, the cell-size for Octomap is increased to 15 cm and that of NDT-OM to 20 cm. The characteristic length for GPOctoMap is found to be 0.3 m. It is observed (Fig. 4.1b) that sparsity of the data, induced by a larger-scale environment and the nature of the sensor, significantly affects the accuracy of both NDT-OM and Octomap while the proposed approach maintains its precision, matching that of GPOctoMap. The same set of parameters, as used for **MINE**, are used for the dataset **PIT** for all techniques. The performance of Octomap, NDT-OM and GPOctoMap are observed to deteriorate while the HGMM approach maintains its level of fidelity, as shown in Fig. 4.1c. It can be concluded from Fig. 4.1 that the proposed approach is able to generalize to diverse environments while the performance of the state of the art is affected by the environment and sensor characteristics.

### 4.4.2 Multi-Fidelity Representation

The implications of the multi-fidelity representation are quantitatively evaluated in Fig. 4.2 for **PIT**. It is observed that the reduction in memory footprint by 50% (from 320 to 160 bytes per point cloud) corresponds to a drop in AUC by 5% (from 0.95 to 0.9). A qualitative visualization of the affect of the hierarchy in terms of reduction in fidelity is shown in Fig. 4.3 via a plot of the probability of occupancy predicted by different layers of the hierarchy. The predictions corresponding to the level $l = 0$ and $l = 4$ are shown with the difference in predictions as a consequence of the drop in



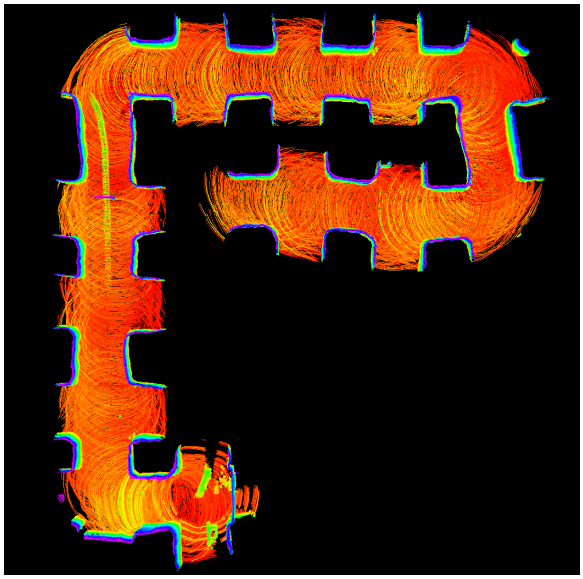

(a) (b)

Figure 4.2: Variation of the accuracy and memory footprint of the models at different levels of the hierarchy plotted against the variation in KL-Divergence (a). A reduction in AUC of 0.05 is observed corresponding to a reduction in memory footprint by 50%. The corresponding ROC curves are shown in (b).

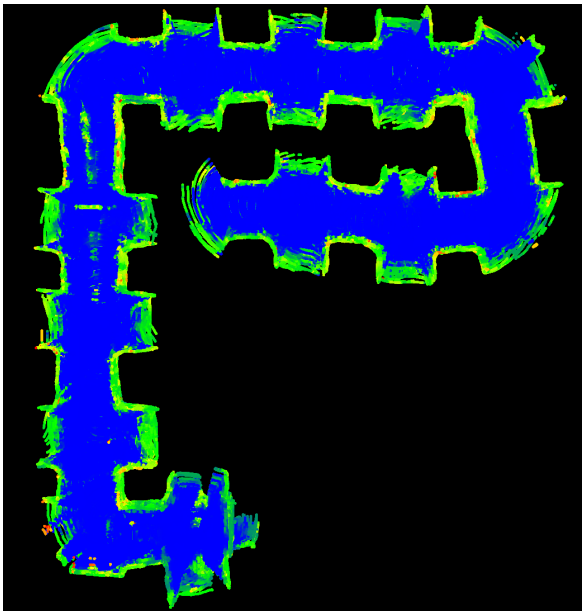
fidelity, highlighted with ellipses. It is observed that the model at level $l = 4$ is slightly noisier than that at level $l = 0$. Specifically, the probability distribution is less sharp in some sections (shown by white ellipses) and the model struggles to capture the observations corresponding to the vehicle (shown by red ellipses). A similar pattern is observed in Fig. 4.4 that evaluates the fidelity of the surface model for **FR_ROOM** with the GMM at level $l = 4$ generating slightly lesser fidelity reconstructions as highlighted by red ellipses.

### 4.4.3 Memory Footprint

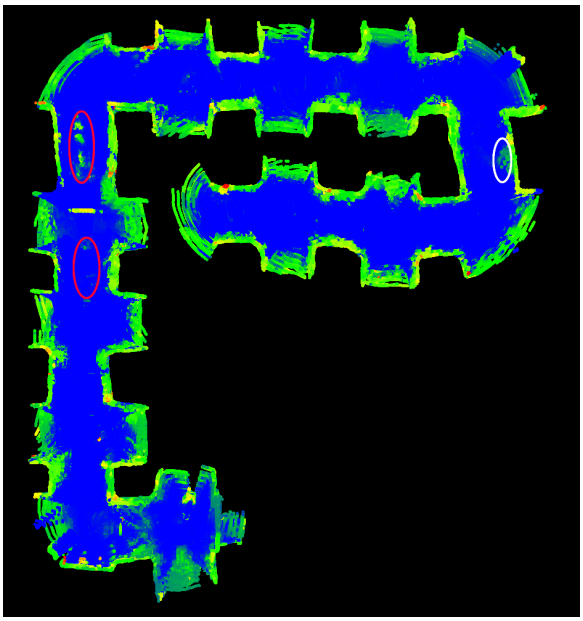
A comparison of the memory footprint of the HGMM approach to the state-of-the-art approaches is shown in Table 4.1. The same set of parameters, as mentioned in Sect. 4.4.1, are used and the corresponding model fidelity is reported in Fig. 4.1. For



(a)

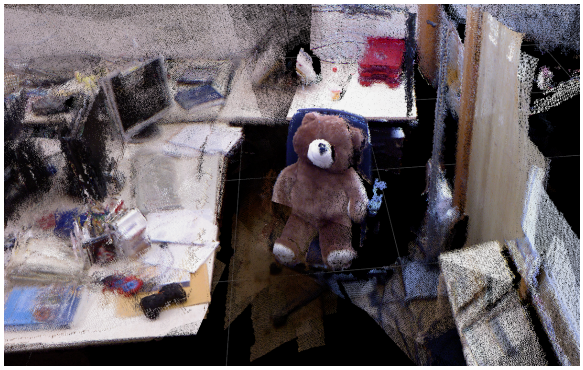


(b)



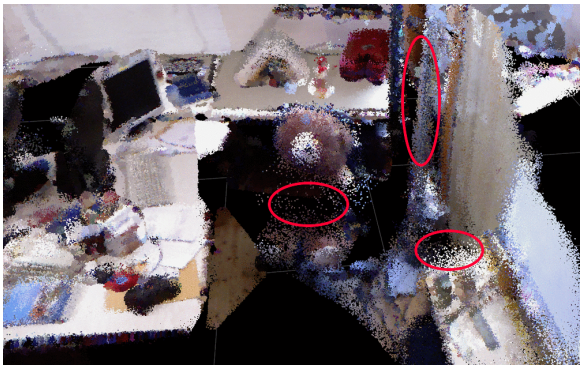
(c)

Figure 4.3: Qualitative evaluation of the implications of the hierarchy on the occupancy distribution. The probability of occupancy, visualized via a heat-map (probability increases from blue to green), for the levels $l = 0$ (b) and $l = 4$ (c) for the input point cloud dataset, **PIT** (a). The lower-fidelity model is observed to be less sharp (white ellipse) and tends to miss on the vehicle in the dataset (red ellipse).



(a)

(b)



(c)

Figure 4.4: Qualitative evaluation of the implications of the hierarchy on the surface model. The reconstruction of a snapshot from **FR_ROOM** (a), for the higher-fidelity level $l = 0$ (b) and lower-fidelity level $l = 4$ (c). The lower-fidelity model is observed to generate noisier reconstructions for complex surfaces (red ellipses) as compared to the higher-fidelity representation. RGB information is for illustration only and not obtained from the model.

all techniques, the footprint increases with the scale of the environment. However, the memory footprint of the proposed approach is observed to be significantly less than all the other techniques for all datasets. The proposed approach is thus able to provide a high-fidelity representation at significantly reduced memory footprint.

| Dataset | Scans | Method | Memory (KB) | Memory / Scan (KB) |
|---|---|---|---|---|
| FR_ROOM | 1361 | Octomap | 8987 | 6.60 |
| | | NDT-OM | 1020 | 0.75 |
| | | GPOctoMap | 3794 | 2.79 |
| | | HGMM | 274 | 0.20 |
| MINE | 3043 | Octomap | 191126 | 62.41 |
| | | NDT-OM | 41785 | 13.73 |
| | | GPOctoMap | 145550 | 47.83 |
| | | HGMM | 1792 | 0.59 |
| PIT | 2006 | Octomap | 646718 | 322.42 |
| | | NDT-OM | 83263 | 41.53 |
| | | GPOctoMap | 156218 | 77.88 |
| | | HGMM | 2362 | 1.18 |

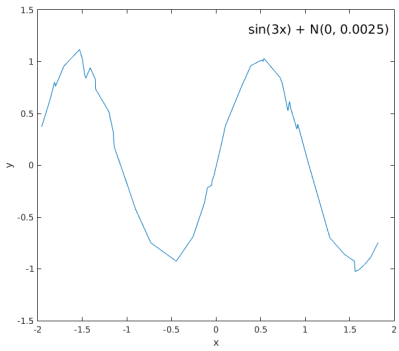
Table 4.1: Comparison of the memory footprint for the lowest level of the proposed hierarchy with competing techniques for three datasets (corresponding to Fig. 4.1). The HGMM approach is observed to have a significantly reduced memory footprint for all datasets.

### 4.4.4 Variance Estimate Characterization

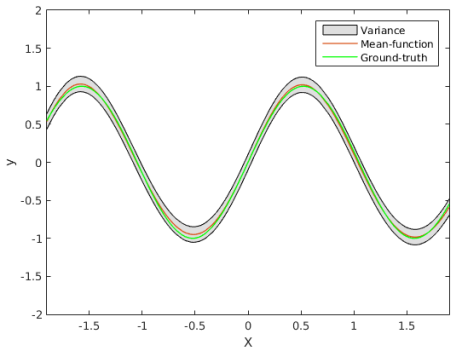
A comparison of the proposed technique with GP Regression is provided to assess correctness of the variance estimate associated with the model predictions. For this, a dataset is generated consisting of samples $x_i \in \mathbb{R}$ and the target function value corrupted with noise given as

$$y_i \;=\; \sin(3x_i) + \mathcal{N}(0, \sigma^2) \tag{4.19}$$

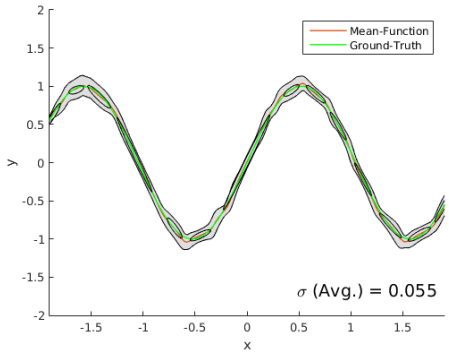
Both the HGMM model and GP Regression are trained with a sequence of 25 sample-sets with each set containing 500 uniformly sampled values $x_i \in [-2, 2]$ and corresponding corrupted output with $\sigma = 0.05$. The models are then queried for mean and



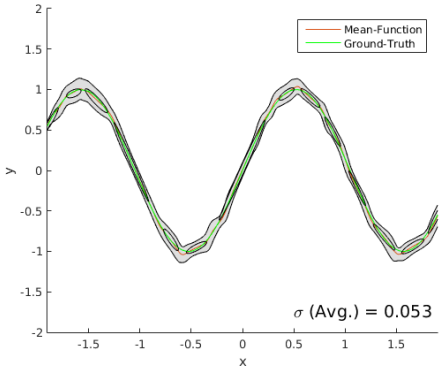

(a) Noisy Input



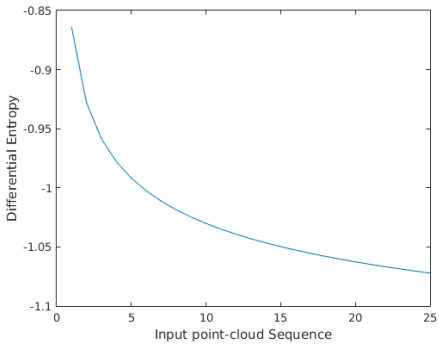

(b) Converged GP



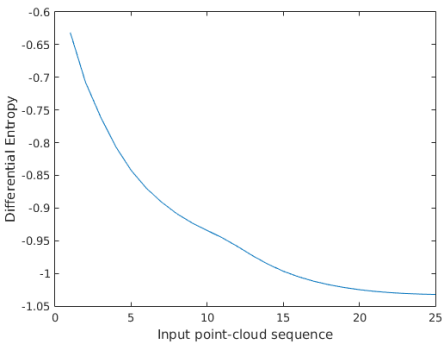

(c) GMM: Point cloud 1




(d) GMM: Point cloud 25




(e) GP: Entropy Curve




(f) GMM: Entropy Curve

Figure 4.5: Characterization of the variance estimate from the proposed framework and Gaussian Process Regression. (a) A sequence of 25 sample-sets each consisting of 500 samples from the simulated noisy function (4.19) ($\sigma = 0.05$) is provided as input to a GP and the proposed framework. The converged GP mean and variance for a test-set (b) and the initial and final state of the GMM with $\lambda_f = 14$ (d,e) are shown. The rate of convergence is demonstrated via differential entropy curves (c) and (f). Both approaches converge to the correct variance estimate and similar entropy values.

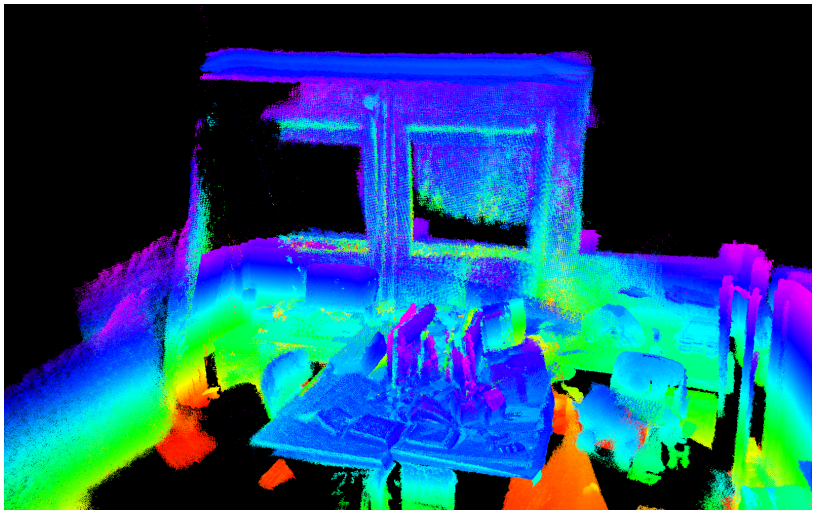
variance estimates for a fixed test set. Figure 4.5 shows the results of this experiment.

The proposed approach estimates the fidelity threshold as $\lambda_f = 14$. The initial and final configuration of the GMM components is shown in Figs. 4.5c and 4.5d and it is observed that the proposed approach converges to a variance estimate same as the injected input noise. The rate of convergence of the proposed approach is compared to that of GP Regression in Figs. 4.5f and 4.5e via a differential entropy curve. It is observed that given the same sequence of points, both approaches converge to a similar entropy value, even though the initial entropy for GP Regression is lower than that of the HGMM. The proposed approach, thus, yields similar performance to a GP Regression framework while eliminating the need to store input data in memory.

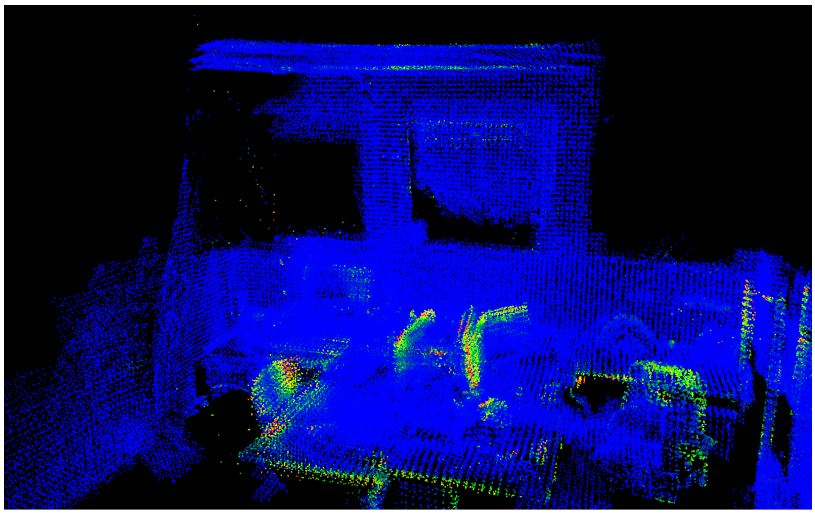
A qualitative visualization of the measure of uncertainty is provided in Fig. 4.6 via a heat-map of the variance estimate associated with the model. The model is trained on dataset **FR_ROOM** and queried for variance at a set of uniformly sampled points on the surface. It is observed that the regions with a higher variance estimate correspond to regions with noisier sensor observations such as table-edges and edges of the monitor (caused by state-estimation noise) which aligns with reasonable expectation.

### 4.4.5 Real-time viability

The high-degree of parallelizability of Expectation-Maximization, likelihood estimation, and posterior probability calculation is exploited via a GPU based implementation of the proposed framework. Also, the sequential nature of the motion of mobile robots (aerial or ground) helps to bound the computation required per point cloud, as the percentage of novel information is limited. The run-time complexity of the framework thus scales with the amount of novelty in subsequent sensor observations.



(a)



(b)

Figure 4.6: Qualitative evaluation of the variance estimate obtained from the proposed framework. For a snapshot from **FR_ROOM** (a), the variance estimate is calculated for a set of uniformly sampled locations on the surface and visualized via a heat-map (b) ( variance growing from blue to yellow). The variance estimate is higher at locations where the sensor measurements are expected to be noisy (table-edges) and the monitor surface which is visibly noisy in the input point cloud.

The robustness of the HGMM formulation to sparsity of measurements is leveraged via subsampling of the input point cloud.

The proposed approach was implemented on NVIDIA Jetson TX2[3], an embedded level System-on-Chip with a discrete CUDA-enabled GPU designed for constrained autonomous systems. The framework operated at a rate of **24** point clouds per second on the TX2 for **FR_ROOM** dataset and **18** point clouds per second for **MINE** and **PIT** datasets.

## 4.5 Summary

The spatial model is extended into a probabilistic representation of occupancy in this chapter. To accomplish this, the surface model is augmented into a 4-tuple HGMM with the fourth variable being the distance function. The distance function maps every point on the surface to zero and points outward from the surface to a distance from the surface along a ray emanating from the sensor.Thus, it is zero for points on the surface and positive for points in free-space. A natural formulation to enable a probabilistic representation of occupancy based on the distance function is introduced. The training procedure is slightly perturbed to introduce free-space observations to enable correlation between $X$ and $W$ to be learned.

A free-space model is introduced to enable incorporation of free-space information into the model. Free-space is defined as the space through which sensor rays pass. In other words, it is space bounded by the point cloud on one end and the sensor on the other. The surface model is leveraged to estimate the number of components in free-space model and the structural sparsity in free-space is leveraged to obtain a constant-time algorithm to generate the model. The free-space model and the surface

[3]NVIDIA Jetson TX2. `https://developer.nvidia.com/embedded/buy/jetson-tx2` [Accessed on 9$^{th}$ July, 2017]

model are then merged to obtain a unified probabilistic model of occupancy.

The proposed representation is evaluated in terms of fidelity, memory footprint and generalizability to diverse environments and compared to the state-of-the-art approaches including Octomap, GPOctomap and NDT-OM. The proposed approach is shown to be more robust to environment peculiarities than competing techniques while having a significantly smaller memory footprint. The implicatons of the hierarchy are also investigated and the real-time viability of the proposed approach is assessed via a GPU-based implementation.

# Chapter 5

# Multimodal Belief Distribution

A homogeneous representation of information from multiple sensing modalities that allows efficient reasoning of the correlation between the modes of information enables absolute abstraction of the underlying sensor and its characteristics. Also, homogenization of information from multiple channels introduces robustness to sporadic loss of data resulting from sensor malfunction or adverse environment conditions . A multimodal representation enables a compact representation of the various properties of the operating environment, such as color, temperature, pressure, and texture, that, in turn, would enable numerous diverse robotic applications ranging from manipulation to active perception. A key challenge in modeling multimodal information is the dependence of computational complexity of any learning technique on the dimensionality of the data. This computational burden associated with training high-dimensional data renders online learning of a model practically infeasible. A multimodal model is, however, essential to enable reasoning over the correlation between different information modalities. This chapter develops an efficient strategy to generate multimodal models of the environment that gets around the curse of dimensionality by enabling

principled and efficient approximation of the correlation between the modes of information based on prior belief. Preliminary results demonstrating the implications of the proposed approach are also presented.

## 5.1 Multimodal Model

The proposed approach enables an efficient multi-fidelity, multimodal representation of the environment by training a set of $J$ Hierarchical Gaussian Mixture Models (HGMMs) for $J$ information modalities, instead of learning a single $J$-tuple HGMM. Employing a set of HGMMs is computationally feasible as the training for each model is independent of the others enabling parallelization of the training procedure. However, learning independent models for each sensing modality precludes the ability to learn correlations between the information modalities. An approach to enable approximation of the correlation via inference based on prior observations is proposed and developed in this section.

### 5.1.1 Definition

Let a location in space be represented by the random variable $X \in \mathbb{R}^3$. Let there be $J$ modes of information available as input and the $i^{\text{th}}$ mode be given as $\Lambda_i, i \in \{1..J\}$. It is assumed that the data from different sensors is registered. This implies that, for instance, the R, G, and B values at each location in space observed by the range sensor is known. It is also assumed that the sensor observations for all information modalities are real-valued, ($\Lambda_i \in \mathbb{R}$).

The proposed multimodal model consists of a set of Hierarchical Gaussian Mixture models, one per information modality. For each sensing modality, an HGMM to

represent the joint density $p(X, \Lambda_i)$ is learnt based on the input data. This results in $J$ 4-tuple Hierarchical Gaussian Mixture Models. Considering the independence of the hierarchy generation on multimodal inference, the discussion going forward is based on the lowest level of the HGMM. Let the lowest level GMM corresponding to the $i^{\text{th}}$ modality contain $K$ component Gaussian distributions specified by parameters, $\Theta_k = (\mu_k,\ \Sigma_k,\ \pi_k)$, where $\mu_k$, $\Sigma_k$, and $\pi_k$ represent the mean, covariance, and mixing weight for the $k^{\text{th}}$ component. Then, the $i^{\text{th}}$ model is expressed as

$$p(X, \Lambda_i) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x, \lambda_i; \mu_k, \Sigma_k) \tag{5.1}$$

where

$$\sum_{k=1}^{K} \pi_k = 1 \qquad \mu_k = \begin{bmatrix} \mu_{kX} \\ \mu_{k\Lambda_i} \end{bmatrix} \qquad \Sigma_k = \begin{bmatrix} \Sigma_{kXX} & \Sigma_{kX\Lambda_i} \\ \Sigma_{k\Lambda_i X} & \Sigma_{k\Lambda_i\Lambda_i} \end{bmatrix}$$

Based on the regression framework developed in Sect. 4.1.2, the value of $\Lambda_i$ at any spatial location $X = x$ can be obtained as the expected value of

$$p_{\Lambda_i|X}(\lambda|x) = \sum_{k=1}^{K} w_k(x)\ \phi(\lambda; m_k(x), \sigma_k^2), \tag{5.2}$$

with the mixing weight

$$w_k(x) = \frac{\pi_k \phi(x; \mu_{kX}, \Sigma_{kXX})}{\sum_{k'=1}^{K} \pi_{k'} \phi(x; \mu_{k'X}, \Sigma_{k'XX})} \tag{5.3}$$

and

$$m_k(x) = \mu_{k\Lambda_i} + \Sigma_{k\Lambda_i X}\ \Sigma_{kXX}^{-1}(x - \mu_{kX}) \tag{5.4}$$

$$\sigma_k^2 = \Sigma_{k\Lambda_i\Lambda_i} - \Sigma_{k\Lambda_i X}\ \Sigma_{kXX}^{-1}\ \Sigma_{kX\Lambda_i} \tag{5.5}$$

The expected value (for 5.2) is given as

$$m(x) \;=\; E[\Lambda_i | X = x] \;=\; \sum_{k=1}^{K} w_k(x)\; m_k(x) \tag{5.6}$$

and the associated variance estimate as

$$\begin{aligned} v(x) = \;& E[(\Lambda_i | X = x)^2] - E[\Lambda_i | X = x]^2 \\ = \;& \sum_{k=1}^{K} w_k(x)(m_k(x)^2 + \sigma_k^2) - (\sum_{k=1}^{K} w_k(x) m_k(x))^2 \end{aligned} \tag{5.7}$$

### 5.1.2 Training

The training for each HGMM essentially follows the same procedure outlined in Algorithm 1 with the only difference being that a 4-tuple HGMM is learned instead of a 3D model. Registered point cloud data and $\Lambda_i$ values are used for training the models. The training dataset consists of 4-tuple data-points of the form $\{X \in \mathbb{R}^3, \Lambda_i \in \mathbb{R}\}$. No augmentation via sampled data as proposed in Sect. 4.1.3 is required for training.

## 5.2 Cross-modal Inference

The proposed approach learns independent HGMMs for the input information modalities. This precludes the approach from learning the correlation between the modalities which in turn disables querying for the value of one modality given the value of another. Correlation between input modalities enables inference of the value of a missing modality (for instance, due to sensor malfunction), given the values of the other modalities resulting in a robust environment representation. The proposed approach

enables approximation, via inference, of the correlation between input modalities thereby enabling a robust representation at a reduced computational cost.

### 5.2.1 Location-based Priors

The key idea that is leveraged to enable inference of one modality based on another is that the observations acquired via sensors pertaining to the various modalities are tied to a physical location in the environment. These observations obtained at some location in the past can be leveraged as prior belief to infer a missing modality at the query location. This mechanism based on prior belief is inspired from everyday human behavior. Humans tend to develop beliefs based on experiences that are then used to inform their choices and actions in everyday life. For instance, a person who has operated a car before and comes across another can infer the kind of sound it would make if turned on. Here, the visual information modality is enabling inference of the audio modality based on prior belief. A similar framework is proposed in this work with the prior belief associated with spatial location instead of time. The system develops a belief distribution as it observes the environment and employs the belief to infer missing information when required.

### 5.2.2 Cross-modal Queries

The proposed framework enables inference of correlation via exploiting the prior belief developed while generating the model. The spatial association of belief is exploited via the variable, $X$, that is shared among the 4-tuple joint distributions for all modalities (5.1). In other words, the correlation between two modalities, $\Lambda_i$ and $\Lambda_j$, can be inferred from the corresponding distributions of $\Lambda_i$ and $\Lambda_j$ over $X$.

Consider the task of estimating the value of the modality, $\Lambda_i$, at some location,

$x_q$, given the value of another modality, $\Lambda_j = \lambda_j$. The first step in leveraging prior belief is to obtain the locations in space at which a similar value of $\Lambda_j$ was observed. This is achieved by obtaining the distribution of $X$ over $\Lambda_j$ from (5.2) as

$$p_{X|\Lambda_j}(x_q|\lambda_j) = \sum_{k=1}^{K} w_k(\lambda_j)\ \phi(x_q; m_k(\lambda_j), \Sigma_k) \tag{5.8}$$

where

$$w_k(\lambda_j) = \frac{\pi_k\ \phi(\lambda_j;\ \mu_{k\Lambda_j},\ \Sigma_{k\Lambda_j\Lambda_j})}{\sum_{k'=1}^{K} \pi'_k\ \phi(\lambda_j;\ \mu_{k'\Lambda_j},\ \Sigma_{k'\Lambda_j\Lambda_j})} \tag{5.9}$$

and

$$m_k(\lambda_j) = \mu_{kX} + \Sigma_{kX\Lambda_j}\ \Sigma^{-1}_{k\Lambda_j\Lambda_j}(\lambda_j - \mu_{k\Lambda_j}) \tag{5.10}$$

$$\sigma^2_k = \Sigma_{kXX} - \Sigma_{kX\Lambda_j}\ \Sigma^{-1}_{k\Lambda_j\Lambda_j}\ \Sigma_{k\Lambda_j X} \tag{5.11}$$

Based on (5.9), the set of components, $S$, that have a non-zero weight for $\Lambda_j = \lambda_j$ is obtained. These components represent regions in the environment where the value of $\Lambda_j \approx \lambda_j$ has been observed. A set of candidate locations, $L$, is then obtained via calculation of the expected value of $X$ for every component in $S$ based on (5.10).

From the set of locations, $L$, where $\Lambda_j$ was observed to be close to $\lambda_j$, the location that provides the most relevant prior, $x_p$, is selected via likelihood maximization.

$$x_p = \underset{x \in L}{\operatorname{argmax}}\ p(\lambda_j|x) \tag{5.12}$$

Having obtained the most likely location $x_p$ to be used as a prior, the expected value of $\Lambda_i$ is regressed based on (5.4) and (5.6).

$$E[\Lambda_i = \lambda_i|X = x_p] = \sum_{k=1}^{K} w_k(x_p)\ m_k(x_p) \tag{5.13}$$

### 5.2.3 Multiple Priors

The formulation developed in Sect. 5.2.2 can be extended to incorporate multiple priors. Information from multiple other sensing modalities is beneficial when inferring the expected value of the target modality, $\Lambda_i$, at some location, $x_q$, where the model for $\Lambda_i$ does not exist, or is lesser fidelity than desired. Absence of desired model-fidelity can occur as a consequence of sensor malfunction, high degree of sparsity, or adverse environment conditions.

Let there be $J$ observed information modalities, expressed as $\Lambda_j,\ j \in \{1, J\}$, at the query location, $x_q$. The target modality is $\Lambda_i, i \notin \{1, J\}$. To incorporate information from multiple priors, the set of locations, L, (Sect. 5.2.2) is augmented to contain candidate locations based on the models of each of the available modalities, $\Lambda_j$. The most pertinent prior location, $x_p$, is chosen via maximization of the sum of likelihood of the models given $L$.

$$x_p \ = \ \underset{x \in L}{\operatorname{argmax}} \sum_{j=1}^{J} p(\Lambda_j = \lambda_j \mid x) \tag{5.14}$$

The expected value of $\Lambda_i$ is regressed based on (5.4) and (5.6).

It is important to note that the proposed formulation is naturally able to handle contradictory priors. If two locations are equally relevant to be used as priors, the formulation will arbitrarily select one of them. This approach aligns with human behavior when confronted with contradicting equal-priority choices.

## 5.3 Results and Analysis

A qualitative and quantitative evaluation of the proposed approach is presented in this section. The other modes of information considered include the R, G, and B channels

of the color spectrum. The dataset used for this evaluation is the publicly available **FR_ROOM** dataset [73] collected using an RGBD sensor. The dataset represents a small-scale cluttered environment and provides RGBD data with ground-truth state estimates.

### 5.3.1 Fidelity of the Model

A qualitative evaluation of the fidelity of the model is presented in Fig. 5.1. The proposed multimodal model is trained for R,G, and B channels for a snapshot of the dataset. The model is then queried for the color information at a set of points on the observed surface, based on (5.6). The point cloud with R,G, and B information obtained from the model is shown in Fig. 5.1b. It is observed that the proposed approach is able to provide a high-fidelity representation of the data. Noise in the reconstruction is observed at some locations (for instance, on the monitor screen). However, the associated variance estimate allows quantification and localization of the noise and enables remedial actions to improve the fidelity of the model at the noisy locations. Figs. 5.1c, 5.1d, and 5.1e plot the associated variance estimates for the *red*, *green*, and *blue* models respectively. It is observed that the noise on the monitor screen is primarily due to a lower fidelity R-model and the noise along the window-sill is caused by the B-model. The root cause of the noise is deduced to be related to initialization and is discussed in Sect. 5.3.3.

A quantitative evaluation of the model fidelity is provided in Table 5.1. The second column provides the Coefficient of Determination[1] ($R^2$-score) for the three

[1]Coefficient of Determination `https://en.wikipedia.org/wiki/Coefficient_of_determination` [Accessed on July 19$^{\text{th}}$, 2017].

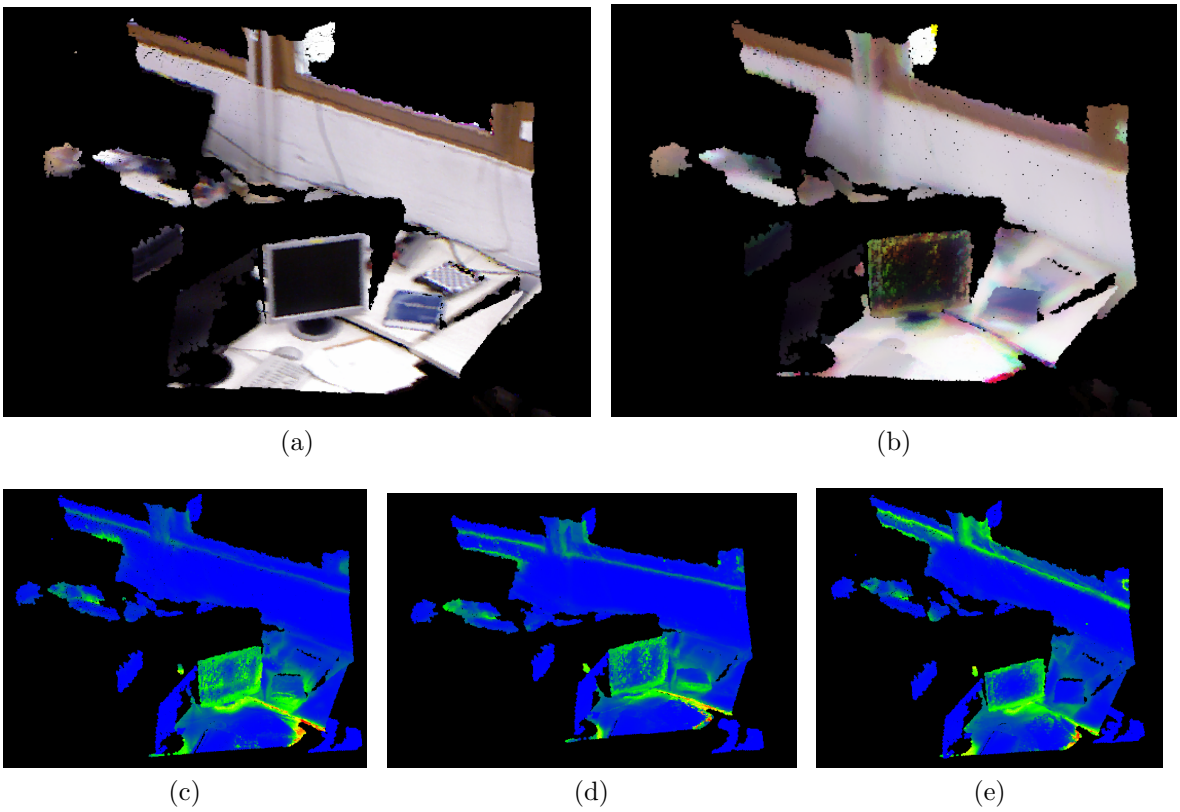


Figure 5.1: Qualitative evaluation of model fidelity. The proposed model is trained on a snapshot of **FR_ROOM** (a) and R, G, and B values are regressed to reconstruct the input point cloud (b). The noise in reconstruction is quantified via variance estimates (heat-map with variance growing from blue to yellow) for *Red* (c), *Green* (d), and *Blue* channels (e) respectively that enable deduction of the source of noise. For instance, the noisy monitor screen is attributed to a low-fidelity Red model.

channels with the input point cloud used as ground-truth. It is observed that the proposed model is able to explain the variance in the input sensor data.

### 5.3.2 Cross-modal Queries

A qualitative evaluation of the ability to infer one modality based on another is presented in Fig. 5.2. The models for the R and G channel are trained on a snapshot of **FR_ROOM** and the model for B is trained on a heavily subsampled version of the

| Modality | $R^2$-Score (Reconstruction) | $R^2$-Score (Inference) |
|---|---|---|
| Red | 0.853 | 0.814 |
| Green | 0.851 | 0.803 |
| Blue | 0.852 | 0.796 |

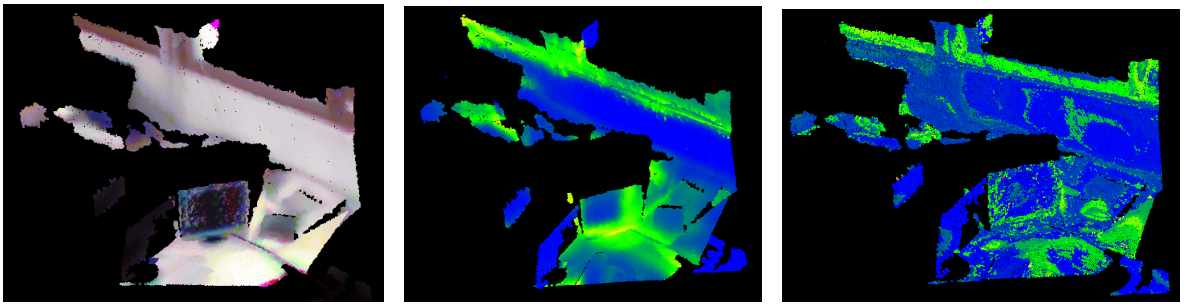
Table 5.1: Coefficient of Determination ($R^2$-scores) for the R, G, and B channels when reconstructed via regression from the model and when inferred based on other modalities.

data, containing 10% of the original set of points. The value for color *blue* is then inferred at all points in the point cloud based on the values of *red* and *green*. Fig. 5.2b shows the variance plot associated with the low-fidelity blue-model trained on sub-sampled data. It is observed that the model is significantly noisier than Fig. 5.1e. Fig. 5.2a shows the reconstructed point cloud with the B channel inferred from the R and G channel and the variance estimates associated with the inferred blue values is shown in Fig. 5.2c. It is observed that the multimodal model enables cross-modal queries and a significant improvement is observed in terms of the *blue* color fidelity as evidenced by a comparison of Figs. 5.2b and 5.2c.

A quantitative evaluation is provided in Table 5.1. The third column provides the $R^2$-score for the R, G, and B channels when inferred based on the other two channels. It is observed that the proposed approach is able to leverage prior belief in a principled manner to approximate the correlation between modalities and infer values of a modality at a location via cross-modal queries.

### 5.3.3 Discussion

Modeling of multimodal information via the procedure outlined in this work is slightly limited by the increased dependence of Expectation-Maximization on the initial pa-



(a) (b) (c)

Figure 5.2: Quantitative evaluation of cross-modal inference. The model for B channel is trained on 10% of the training data and results in a low-fidelity model as shown by the associated variance estimates (b). The point cloud is reconstructed via inference of *blue* values based on the *red* and *green* models (a). A significant improvement in model's belief over *blue* color is observed (c).

rameters for other modes of information. The simplistic strategy to initialize the component means and covariance for the surface model proposed in Sect. 3.3.4 does not generalize in terms of performance to other modes of information such as color. This results in degraded model-fidelity as evidenced by the noise in Fig. 5.1. However, it must be noted that the associated variance estimate enables localization of the source of noise (Figs. 5.1c, 5.1d, and 5.1e) that can be leveraged to trigger local training of the model to improve fidelity. Also, alternative strategies, based on Dirichlet Processes, known to provide principled estimates for initial parameters [23], are being investigated.

## 5.4 Summary

A framework to incorporate multimodal information into the model is developed in the chapter. The ability to model information from multiple sensing modalities not only enables a more powerful representation of the environment via homogenization of multimodal information but also makes the model robust to sensor malfunction.

The proposed modeling approach enables cross-modal inference by leveraging the dependence of prior belief on locations in space. A principled strategy that identifies the most likely location, based on other available modalities, to enable inference is proposed and developed. The representation fidelity of the multimodal model is evaluated by modeling R, G, and B channels of the color spectrum and reconstructing the input data via regression. Also, cross-modal inference is evaluated via inferring one of the channels based on the belief distribution over its correlation with other modalities.

# Chapter 6

# Conclusion

## 6.1 Summary

Autonomous systems are increasingly being deployed for operation in perceptually challenging, diverse and potentially hazardous environments such as for monitoring in power-plants, information-gathering in underground mines and tunnels, and search and rescue operations in disaster-hit areas. The operating environment for such operations is not always known *a priori* and thus a representation of the environment to enable reasoning with respect to the surroundings needs to be generated online. A probabilistic environment representation that allows efficient high-fidelity modeling and inference towards enabling informed planning (active perception) on a computationally constrained mobile autonomous system is proposed in this work. The traits of the technique include its generative nature, high-fidelity, small memory footprint, hierarchical and support for uncertainty measure.

An overview of the various approaches to enable online spatial modeling and occupancy mapping presented in the literature is provided in Chapter 2 . The techniques

can be categorized into three main classes: (a) Voxel-Based representations that discretize the environment into voxels and maintain the likelihood of occupancy per cell assuming conditional independence with other cells. Several improvements have been proposed to this representation to address memory concerns (Octomap), artifacts of conditional independence (forward sensor models) and compactness of the model (Elevation maps). (b) Generative spatial models that learn a parametric model over the environment to enable a compact representation. A hybrid strategy, that is a combination of voxel-based representations and generative models is the Normal Distribution Transform (NDT) that learns a Gaussian distribution over the rays that pass through a voxel. (c) Continuous Occupancy representations that learn a continuous distribution over occupancy in the environment. Two approaches of significance include Gaussian Process Occupancy Maps that employ Gaussian Processes to estimate the occupancy distribution, and Hilbert maps that project the sensor data into a higher-dimensional space and employ logistic regression as a discriminative model over occupancy

A multi-fidelity spatial model via a hierarchy of Gaussian Mixture Models is developed in Chapter 3. The hierarchy consists of a GMM per level and each GMM differs from the others in terms of the number of components and the fidelity of representation. A methodology, governed by information-theoretic principles, to estimate the required number of components in the levels of the hierarchy is presented. The distinct knee-point in the growth of divergence when the size of the GMM is reduced is leveraged to obtain the fidelity threshold, $\lambda_f$. An iterative algorithm to generate the level $l$ of the hierarchy by merging similar components in the level $l-1$ is presented as a principled strategy to generate an information hierarchy.

The proposed spatial model is also compared in terms of fidelity to Normal Distri-

bution Transform (NDT) spatial representation and shown to be more accurate with less manual tuning. A qualitative evaluation is done to demonstrate the fidelity of the representation via reconstruction of input point-cloud data leveraging the generative properties of the model

The spatial model is extended into a probabilistic representation of occupancy in Chapter 4. To accomplish this, the spatial model is augmented into a 4-tuple HGMM with the fourth variable being the distance function. The distance function maps every point on the spatial to zero and points outward from the spatial to a distance from the spatial along a ray emanating from the sensor.Thus, it is zero for points on the spatial and positive for points in free-space. A natural formulation to enable a probabilistic representation of occupancy based on the distance function is introduced. The training procedure is slightly perturbed to introduce free-space observations to enable correlation between $X$ and $W$ to be learned.

A free-space model is introduced to enable incorporation of free-space information into the model. Free-space is defined as the space through which sensor rays pass. In other words, it is space bounded by the point-cloud on one end and the sensor on the other. The spatial model is leveraged to estimate the number of components in free-space model and the structural sparsity in free-space is leveraged to obtain a constant-time algorithm to generate the model. The free-space model and the spatial model are then merged to obtain a unified probabilistic model of occupancy.

The proposed representation is evaluated in terms of fidelity, memory footprint and generalizability to diverse environments and compared to the state-of-the-art approaches including Octomap, GPOctomap and NDT-OM. The proposed approach is shown to be more robust to environment peculiarities than competing techniques while having a significantly smaller memory footprint. The implicatons of the hier-

archy are also investigated and the real-time viability of the proposed approach is assessed via a GPU-based implementation.

A framework to incorporate multimodal information into the model is developed in the chapter. The ability to model information from multiple sensing modalities not only enables a more powerful representation of the environment via homogenization of multimodal information but also makes the model robust to sensor malfunction. The proposed modeling approach enables cross-modal inference by leveraging the dependence of prior belief on locations in space. A principled strategy that identifies the most likely location, based on other available modalities, to enable inference is proposed and developed. The representation fidelity of the multimodal model is evaluated by modeling R, G, and B channels of the color spectrum and reconstructing the input data via regression. Also, cross-modal inference is evaluated via inferring one of the channels based on the belief distribution over its correlation with other modalities.

## 6.2 Future Work

The objective of this work is to develop a a large-scale high-fidelity environment representation that scales with the information content of the environment. In order to achieve this goal, a methodology to incorporate global consistency into the model is essential. Traditionally, global consistency of the map has been achieved via loop-closure detection followed by bundle-adjustment. A coupled state-estimation and modeling framework is thus called for that would enable the traditional SLAM formulation in the space of the continuous belief distribution. This is one significant avenues of work that would be required to enable deployment of the model on a

large-scale.

The proposed approach enables representation of the world as a continuous belief distribution. Another interesting future direction could be to investigate formulating inspection (or equivalently exploration) as an optimization over the continuous belief distribution that should ideally eliminate all restrictive assumptions generally made when using occupancy grid as the map representation. The associated measure of uncertainty naturally enables inspection with respect to the environment and could turn out to be even more powerful if the continuous nature of the representation is leveraged.

# Bibliography

[1] H. Andreasson, M. Magnusson, and A. Lilienthal. Has something changed here? Autonomous Difference Detection for Security Patrol Robots. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3429–3435, Oct 2007.

[2] Dominik Belter and Piotr Skrzypczyński. Rough Terrain Mapping and Classification for Foothold Selection in a Walking Robot. *Journal of Field Robotics*, 28(4):497–528, 2011.

[3] V. H. Bennetts, E. Schaffernicht, T. Stoyanov, A. J. Lilienthal, and M. Trincavelli. Robot Assisted Gas Tomography - Localizing Methane Leaks in Outdoor Environments. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6362–6367, May 2014.

[4] Peter Biber and Wolfgang Straßer. The Normal Distributions Transform: A new approach to Laser Scan Matching. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 3, pages 2743–2748. IEEE, 2003.

[5] Andrew Blake, Carsten Rother, M. Brown, Patrick Perez, and Philip Torr. *Interactive Image Segmentation Using an Adaptive GMMRF Model*, pages 428–441.

Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[6] Hamparsum Bozdogan. Akaike's information Criterion and Recent Developments in Information Complexity. *Journal of mathematical psychology*, 44(1):62–91, 2000.

[7] L. Burget, P. Schwarz, M. Agarwal, P. Akyazi, K. Feng, A. Ghoshal, O. Glembek, N. Goel, M. Karafit, D. Povey, A. Rastrow, R. C. Rose, and S. Thomas. Multilingual Acoustic Modeling for Speech Recognition based on Subspace Gaussian Mixture Models. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4334–4337, March 2010.

[8] S. Calinon, F. Guenter, and A. Billard. On Learning, Representing, and Generalizing a Task in a Humanoid Robot. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(2):286–298, April 2007.

[9] Sonia Chernova and Manuela Veloso. Confidence-based Policy Learning from Demonstration Using Gaussian Mixture Models. In *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '07, pages 233:1–233:8, New York, NY, USA, 2007. ACM.

[10] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

[11] Kevin Doherty, Jinkun Wang, and Brendan Englot. Probabilistic Map Fusion for Fast, Incremental Occupancy Mapping with 3d Hilbert Maps. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 1011–1018. IEEE, 2016.

[12] Ben Eckart, Kihwan Kim, Alejandro Troccoli, Alonzo Kelly, and Jan Kautz. MLMD: Maximum Likelihood Mixture Decoupling for Fast and Accurate Point Cloud Registration. In *3D Vision (3DV), 2015 International Conference on*, pages 241–249. IEEE, 2015.

[13] Benjamin Eckart and Alonzo Kelly. Rem-seg: A robust em algorithm for parallel segmentation and registration of point clouds. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 4355–4362. IEEE, 2013.

[14] Benjamin Eckart, Kihwan Kim, Alejandro Troccoli, Alonzo Kelly, and Jan Kautz. Accelerated Generative Models for 3D Point Cloud Data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5497–5505, 2016.

[15] Alberto Elfes. *Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 1989.

[16] Paulo Martins Engel and Milton Roberto Heinen. Incremental learning of Multivariate Gaussian Mixture Models. In *Brazilian Symposium on Artificial Intelligence*, pages 82–91. Springer, 2010.

[17] Nathaniel Fairfield, George Kantor, and David Wettergreen. Real-Time SLAM with Octree Evidence Grids for Exploration in Underwater Tunnels. *Journal of Field Robotics*, 24(1-2):03–21, 2007.

[18] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning Generative Visual Models from few Training Examples: An Incremental Bayesian Approach Tested on 101

Object Categories. *Computer Vision and Image Understanding*, 106(1):59 – 70, 2007. Special issue on Generative Model Based Vision.

[19] R. Fletcher. *Practical Methods of Optimization*. 1987.

[20] J. Fournier, B. Ricard, and D. Laurendeau. Mapping and exploration of complex environments using persistent 3d model. In *Computer and Robot Vision, 2007. CRV '07. Fourth Canadian Conference on*, pages 403–410, May 2007.

[21] M Ghaffari Jadidi, Jaime Valls Miró, Rafael Valencia, Juan Andrade-Cetto, and Gamini Dissanayake. Exploration in Information Distribution Maps. In *Robotics Science and Systems*. Technische Universitat Berlin, 2013.

[22] Jacob Goldberger, Shiri Gordon, and Hayit Greenspan. An efficient image similarity measure based on approximations of KL-divergence between two Gaussian mixtures. In *Proceedings of Ninth IEEE International Conference on Computer Vision*, pages 487–493, 2003.

[23] Dilan Görür and Carl Edward Rasmussen. Dirichlet Process Gaussian Mixture Models: Choice of the Base Distribution. *Journal of Computer Science and Technology*, 25(4):653–664, 2010.

[24] W. R. Green and H. Grobler. Normal Distribution Transform Graph-based Point Cloud Segmentation. In *2015 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech)*, pages 54–59, Nov 2015.

[25] Vitor Guizilini and Fabio Ramos. Large-scale 3D Scene Reconstruction with Hilbert Maps. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 3247–3254. IEEE, 2016.

[26] J-S Gutmann, Masaki Fukuchi, and Masahiro Fujita. A Floor and Obstacle Height Map for 3D Navigation of a Humanoid Robot. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 1066–1071. IEEE, 2005.

[27] M. Heikkila and M. Pietikainen. A Texture-Based Method for Modeling the Background and Detecting Moving Objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):657–662, April 2006.

[28] Armin Hornung, Kai M Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. Octomap: An Efficient Probabilistic 3D Mapping Framework based on Octrees. *Autonomous Robots*, 34(3):189–206, 2013.

[29] Maani Ghaffari Jadidi, Jaime Valls Miró, Rafael Valencia, and Juan Andrade-Cetto. Exploration on Continuous Gaussian Process Frontier Maps. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 6077–6082. IEEE, 2014.

[30] Satoshi Kagami, Koichi Nishiwaki, James J Kuffner, Kei Okada, Masayuki Inaba, and Hirochika Inoue. Vision-ased 2.5 D Terrain Modeling for Humanoid Locomotion. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 2, pages 2141–2146. IEEE, 2003.

[31] Kittipat Kampa, Erion Hasanbelliu, and Jose C Principe. Closed-form Cauchy-Schwarz PDF Divergence for Mixture of Gaussians. In *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pages 2578–2585. IEEE, 2011.

[32] S. M. Khansari-Zadeh and A. Billard. BM: An iterative Algorithm to learn Stable Non-linear Dynamical Systems with Gaussian Mixture Models. In *2010 IEEE International Conference on Robotics and Automation*, pages 2381–2388, May 2010.

[33] S. M. Khansari-Zadeh and A. Billard. Learning Stable Nonlinear Dynamical Systems With Gaussian Mixture Models. *IEEE Transactions on Robotics*, 27(5):943–957, Oct 2011.

[34] Soohwan Kim and Jonghyuk Kim. GPmap: A Unified Framework for Robotic Mapping based on Sparse Gaussian Processes. In *Field and Service Robotics*, pages 319–332. Springer, 2015.

[35] Soohwan Kim, Jonghyuk Kim, et al. Recursive Bayesian Updates for Occupancy Mapping and Surface Reconstruction. In *Proceedings of the Australasian Conference on Robotics and Automation*, 2014.

[36] Solomon Kullback and Richard A Leibler. On Information and Sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[37] R. Lakaemper, L. J. Latecki, and D. Wolter. Incremental Multi-Robot Mapping. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3846–3851, Aug 2005.

[38] Tobias Lang, Christian Plagemann, and Wolfram Burgard. Adaptive Non-Stationary Kernel Regression for Terrain Modeling. In *Robotics: Science and Systems*, 2007.

[39] Dar-Shyang Lee. Effective Gaussian Mixture Learning for Video Background Subtraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5):827–832, May 2005.

[40] Kyoungmin Lee and Wan Kyun Chung. Effective Maximum Likelihood Grid Map with Conflict Evaluation Filter using Sonar Sensors. *IEEE Transactions on Robotics*, 25(4):887–901, 2009.

[41] Martin Magnusson, Henrik Andreasson, Andreas Nchter, and Achim J. Lilienthal. Automatic Appearance-based Loop Detection from Three-Dimensional Laser Data using the Normal Distributions Transform. *Journal of Field Robotics*, 26(11-12):892–914, 2009.

[42] Martin Magnusson, Achim Lilienthal, and Tom Duckett. Scan Registration for Autonomous Mining Vehicles using 3D-NDT. *Journal of Field Robotics*, 24(10):803–827, 2007.

[43] Roman Marchant and Fabio Ramos. Bayesian Optimisation for Informative Continuous Path Planning. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 6136–6143. IEEE, 2014.

[44] Kantilal Varichand Mardia, John T Kent, and John M Bibby. Multivariate Analysis. page 63, 1980.

[45] N. Martel-Brisson and A. Zaccarin. Moving cast Shadow Detection from a Gaussian Mixture Shadow Model. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 643–648 vol. 2, June 2005.

[46] Stephen J. McKenna, Yogesh Raja, and Shaogang Gong. Tracking Colour Objects using Adaptive Mixture Models. *Image and Vision Computing*, 17(3):225 – 231, 1999.

[47] Donald Meagher. Geometric Modeling using Octree Encoding. *Computer graphics and image processing*, 19(2):129–147, 1982.

[48] Hans P Moravec. 3D Graphics and the Wave Theory. *ACM SIGGRAPH computer graphics*, 15(3):289–296, 1981.

[49] Hans P Moravec. Sensor Fusion in Certainty Grids for Mobile Robots. *AI Magazine*, 9(2):61, 1988.

[50] Andrew A. Neath and Joseph E. Cavanaugh. The Bayesian Information Criterion: Background, Derivation, and Applications. *Wiley Interdisciplinary Reviews: Computational Statistics*, 4(2):199–203, 2012.

[51] E. Nelson and N. Michael. Information-theoretic Occupancy Grid Compression for High-speed Information-based Exploration. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4976–4982, Sept 2015.

[52] Simon T OCallaghan and Fabio T Ramos. Gaussian Process Occupancy Maps. *The International Journal of Robotics Research*, 31(1):42–62, 2012.

[53] Mark Paskin and Sebastian Thrun. Robotic Mapping with Polygonal Random Fields. *arXiv preprint arXiv:1207.1399*, 2012.

[54] Kaustubh Pathak, Andreas Birk, Jann Poppinga, and Soren Schwertfeger. 3D Forward Sensor Modeling and Application to Occupancy Grid based Sensor Fu-

sion. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2059–2064. IEEE, 2007.

[55] P. Payeur, P. Hebert, D. Laurendeau, and C. M. Gosselin. Probabilistic Octree Modeling of a 3D Dynamic Environment. In *Proceedings of International Conference on Robotics and Automation*, volume 2, pages 1289–1296 vol.2, Apr 1997.

[56] Luis Pedraza, Diego Rodriguez-Losada, Fernando Matia, Gamini Dissanayake, and Jaime Valls Miró. Extending the Limits of Feature-based SLAM with B-splines. *IEEE Transactions on Robotics*, 25(2):353–366, 2009.

[57] Haim Permuter, Joseph Francos, and Ian Jermyn. A Study of Gaussian Mixture Models of Color and Texture Features for Image Classification and Segmentation. *Pattern Recognition*, 39(4):695 – 706, 2006. Graph-based Representations.

[58] C. Plagemann, S. Mischke, S. Prentice, K. Kersting, N. Roy, and W. Burgard. Learning Predictive Terrain Models for Legged Robot Locomotion. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3545–3552, Sept 2008.

[59] D. Povey, L. Burget, M. Agarwal, P. Akyazi, K. Feng, A. Ghoshal, O. Glembek, N. K. Goel, M. Karafit, A. Rastrow, R. C. Rose, P. Schwarz, and S. Thomas. Subspace Gaussian Mixture Models for speech recognition. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4330–4333, March 2010.

[60] Morgan Quigley, Josh Faust, Tully Foote, and Jeremy Leibs. ROS: an open-source Robot Operating System. In *Workshop open source software (Vol. 3, No. 3.2, p. 5), International Conference on Robotics and Automation*, May 2009.

[61] Ali Rahimi and Ben Recht. Random Features for Large-Scale Kernel Machines. In *In Neural Infomration Processing Systems*, 2007.

[62] Fabio Ramos and Lionel Ott. Hilbert Maps: Scalable Continuous Occupancy Mapping with Stochastic Gradient Descent. *The International Journal of Robotics Research*, 35(14):1717–1730, 2016.

[63] Carl Edward Rasmussen and Christopher KI Williams. *Gaussian Processes for Machine Learning*, volume 1. MIT press Cambridge, 2006.

[64] D. A. Reynolds and R. C. Rose. Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models. *IEEE Transactions on Speech and Audio Processing*, 3(1):72–83, Jan 1995.

[65] Douglas A. Reynolds. Speaker Identification and Verification using Gaussian Mixture Speaker Models. *Speech Communication*, 17(1):91 – 108, 1995.

[66] Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn. Speaker Verification Using Adapted Gaussian Mixture Models. *Digital Signal Processing*, 10(1):19 – 41, 2000.

[67] J. Saarinen, H. Andreasson, T. Stoyanov, and A. J. Lilienthal. Normal Distributions Transform Monte-Carlo Localization (NDT-MCL). In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 382–389, Nov 2013.

[68] Jari P Saarinen, Henrik Andreasson, Todor Stoyanov, and Achim J Lilienthal. 3D Normal Distributions Transform Occupancy Maps: An Efficient Representation for Mapping in Dynamic Environments. *The International Journal of Robotics Research*, 32(14):1627–1644, 2013.

[69] Claude E Shannon and Warren Weaver. *The Mathematical Theory of Communication*. University of Illinois press, 1998.

[70] S. Srivastava and N. Michael. Approximate Continuous Belief Distributions for Precise Autonomous Inspection. In *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 74–80, Oct 2016.

[71] C. Stauffer and W. E. L. Grimson. Adaptive Background Mixture Models for Real-Time Tracking. In *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, volume 2, page 252 Vol. 2, 1999.

[72] Todor Stoyanov, Martin Magnusson, Henrik Andreasson, and Achim J Lilienthal. Fast and Accurate Scan Registration through Minimization of the Distance between Compact 3D NDT Representations. *The International Journal of Robotics Research*, 31(12):1377–1393, 2012.

[73] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A Benchmark for the Evaluation of RGB-D SLAM Systems. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[74] Hsi Guang Sung. *Gaussian Mixture Regression and Classification*. PhD thesis, Rice University, 2004.

[75] Yee W Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Sharing Clusters among Related Groups: Hierarchical Dirichlet processes. In *Advances in neural information processing systems*, pages 1385–1392, 2005.

[76] S. Thrun, C. Martin, Yufeng Liu, D. Hahnel, R. Emery-Montemerlo, D. Chakrabarti, and W. Burgard. A Real-Time Expectation-Maximization Algorithm for Acquiring Multiplanar Maps of Indoor Environments with Mobile Robots. *IEEE Transactions on Robotics and Automation*, 20(3):433–443, June 2004.

[77] Sebastian Thrun. Learning Occupancy Grids with Forward Models. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, volume 3, pages 1676–1681. IEEE, 2001.

[78] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. 2005.

[79] P. A. Torres-Carrasquillo, D. A. Reynolds, and J. R. Deller. Language Identification using Gaussian Mixture Model Tokenization. In *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages I–757–I–760, May 2002.

[80] Pedro A Torres-Carrasquillo, Elliot Singer, Mary A Kohler, Richard J Greene, Douglas A Reynolds, and John R Deller Jr. Approaches to Language Identification using Gaussian Mixture Models and Shifted Delta Cepstral Features. In *Interspeech*, 2002.

[81] Volker Tresp. A Bayesian Committee Machine. *Neural computation*, 12(11):2719–2741, 2000.

[82] R. Triebel, P. Pfaff, and W. Burgard. Multi-Level Surface Maps for Outdoor Terrain Mapping and Loop Closing. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2276–2282, Oct 2006.

[83] R. Valencia, J. Saarinen, H. Andreasson, J. Vallv, J. Andrade-Cetto, and A. J. Lilienthal. Localization in Highly Dynamic Environments using Dual-Timescale NDT-MCL. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3956–3962, May 2014.

[84] Michael Veeck and Wolfram Veeck. Learning Polyline Maps from Range Scan Data acquired with Mobile Robots. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 2, pages 1065–1070. IEEE.

[85] J. Wang and B. Englot. Fast, Accurate Gaussian Process Occupancy Maps via Test-data Octrees and Nested Bayesian Fusion. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1003–1010, May 2016.

[86] Christopher K. I. Williams and Matthias Seeger. Using the NyströM Method to Speed Up Kernel Machines. In *Proceedings of the 13th International Conference on Neural Information Processing Systems*, NIPS'00, pages 661–667, Cambridge, MA, USA, 2000. MIT Press.

[87] Pavan Yalamanchili, Umar Arshad, Zakiuddin Mohammed, Pradeep Garigipati, Peter Entschev, Brian Kloppenborg, James Malcolm, and John Melonakos. ArrayFire - A high performance software library for parallel computing with an easy-to-use API, 2015.

[88] Y. Zhang, M. Brady, and S. Smith. Segmentation of Brain MR Images through a Hidden Markov Random Field Model and the Expectation-Maximization Algorithm. *IEEE Transactions on Medical Imaging*, 20(1):45–57, Jan 2001.

[89] Z. Zivkovic. Improved Adaptive Gaussian Mixture Model for Background Subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 2, pages 28–31 Vol.2, Aug 2004.