FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Novelty Detection for Semantic Place Categorization

André Susano Pinto

Master in Informatics and Computing Engineering

Supervisor: Andrzej Pronobis[1] (Postdoctoral Researcher)

Supervisor FEUP: Luís Paulo Reis[2] (Assistant Professor)

[1] Centre for Autonomous Systems, The Royal Institute of Technology
SE100-44 Stockholm, Sweden

[2] LIACC - Artificial Intelligence And Computer Science Lab. of the University of Porto
Rua Dr. Roberto Frias, s/n 4200-465 Porto, Portugal

18 July, 2011

Novelty Detection for Semantic Place Categorization

André Susano Pinto

Master in Informatics and Computing Engineering

Approved in oral examination by the committee:

Chair: António Fernando Vasconcelos Cunha Castro Coelho (Assistant Professor of FEUP[1])

External Examiner: Luis Miguel Parreira Correia (Associate Professor of FCUL[2])

Supervisor: Luis Paulo Gonçalves dos Reis (Assistant Professor of FEUP[1])

18 July, 2011

[1] Faculdade de Engenharia da Universidade do Porto

[2] Faculdade de Ciências da Universidade de Lisboa

Abstract

For a long time humanity has dreamed that one day robots will be among us. They will explore our world and interact with us, understand our concepts and reason. An important step in that direction is endowing robots with knowledge about human concepts and semantics. However, it is unrealistic to believe that the human world can be fully modeled in the robot's brain at the design stage. Therefore, robots must be able to adapt and learn when confronted with novel situations. Detection of novel situations, where the knowledge of the robot is not sufficient plays an important role in adaptation and learning of new concepts and is the main topic of this thesis.

In the context of mobile robotics, spatial concepts and semantics are crucial to enable the robot to perform complex human-like tasks and human interactions. For handling those, a robot builds a representation of space extended with semantic properties, process which is known as semantic mapping. This representation identifies spatial entities and classifies them according to their meaning to humans allowing the robot to reason at a high abstraction level. For example, humans categorize spaces as kitchens, bedrooms, corridors, offices, theaters, computer labs, etc. based on spatial properties, objects and actions that are characteristic of those spaces. Similarly, if a robot had a knowledge about those spatial concepts, it could answer such questions as: "Where are cornflakes?"

This thesis studies a semantic mapping process that employs spatial semantic knowledge and represents it using a probabilistic graphical model. It develops methods for performing detection of novel room categories which were not previously known to the robot, using for that graphical models. Finally, it draws attention to the usefulness of unlabelled data for the novelty detection process in order to considerably increase the detection accuracy.

Resumo

Já há muitos anos que a humanidade sonha que um dia robôs caminharam lado a lado com os humanos. Acredita-se que eles serão capaz de se mover e interagir connosco, compreender os nossos conceitos e serem capazes de pensar. Um passo importante nesta direcção é a criação de robôs com conhecimento de conceitos e significados próprios do mundo humano. No entanto é irrealista acreditar que será possível descrever e representar em toda a sua extensão tais conceitos. Assim os agentes deverão ser capazes de aprender e adaptar o seu conhecimento quando confrontados com novas situações. A detecção destas novas situações, onde o conhecimento de um agente não é suficiente, têm um papel importante na adaptação e aprendizagem de novos conceitos e significados e é por isso o tópico principal desta tese.

No contexto de agentes moveis, para facilitar o deslocamento e planeamento, os agentes constroem uma representação do ambiente. Quando essa representação é estendida com conceitos e categorias com significado para humanos, as tarefas de alto nível ficam mais fáceis. Por exemplo, a uma escala de quartos, os humanos categorizam o espaço em: cozinhas, quartos, corredores, escritórios, etc... Ao relacionar quartos com estas categorias que descrevem propriedades e acções percebidas por humanos, um agente consegue resolver tarefas como: "Onde estão os cereais?"

Esta tese estuda um processo, que usa conhecimento de conceitos espaciais e categorias com significado para humanos, para realizar mapeamento semântico do ambiente através de modelos gráficos probabilísticos. Apresenta depois técnicas para realizar detecção de novas categorias. Chama também a atenção para a utilização de dados não supervisionados para efectivamente melhorar a exactidão da detecção.

Acknowledgements

This thesis would have not been possible without the help of some other persons. I would like to thank to Andrzej Pronobis, for introducing me to the problem and directing it towards my interests in computer science, for the meetings, insights and endless reviews of my work. Also to Carl Henrik Ek, for enthusiastically taking part on the meetings discussing graphical models and related work.

The Computer Vision and Active Perception Lab at Royal Institute of Technology in Stockholm were also a key piece for providing such a staff and facilities that surrounded me during one year of thesis and exchange studies, and whose courses replenish my interests in machine learning. Also Faculdade de Engenharia of University of Porto who made it possible for me to study and perform my thesis abroad. Thanks to Luis Paulo Reis, for accepting to supervise my thesis and pushing me to submit a paper.

Virgile Högman, for besides handling me as a work colleague and chatter-box, becoming a friend. And Erik Ass for all the interesting coding and math discussions on totally random problems. I cannot also avoid to thank my neighbours back in Nockeby, in special to Diogo Gonçalves and Elise Löbker that kept chatting with me through the nights during my thesis work.

Thanks to Mariatorget people who kept asking how my thesis was going and when it was going to be finished. And for all the parties, drinking, relaxing and nights out! Thanks to Elina Säfsten, for being a cool neighbour with whom I had the pleasure to spent quite some time talking and partying with her and her Swedish friends. Thanks to Manuel Gattermayr and Tommaso Facchini for being awesome neighbours as well.

I am also grateful to Bernhard Schwaighofer, Manuel, Tommaso and others for an amazing road trip, night fires, sunsets and sunrises on Sweden. Special thanks to Bernie for during the road trip promising to wake me up before my sleeping bag starts to burn and melt with my skin and eventually turning me water proof.

And thanks to all the party people, not forgetting Andrzej, who showed me how to do a PhD party in case I ever decide to seek one.

Last but not the least, thanks to my family and friends back in Porto and other parts of the world, who kept talking and cheered me up.

"I am thankful to all the awesome people who were part of this Stockholm chapter of my life" – *André Susano Pinto*

Contents

Acronyms

CRFH Composed Receptive Field Histogram. 10, 22

Dora Dora: The Explorer. 17, 18, 21, 22

K-PCA Kernel Principal Component Analysis. 28

MAP Maximum a Posteriori. 15

PCA Principal Component Analysis. 28

SIFT Scalar Invariant Feature Transform. 9, 22

SLAM Simultaneous Localization and Mapping. 21

SVM Support Vector Machine. 6–8, 10, 24, 28

Chapter 1

Introduction

There has been several efforts in the areas of artificial intelligence and mobile robotics in creating robots that are able to interact with humans and their environments. Those robots would be able to explore our houses and offices. They would be able to communicate with humans and perform typical human-like tasks.

In order to reason about and perform actions in the surrounding environment, the robot first has to obtain a manageable representation of it. Such a representation provides the robot with understanding of how the spatial entities relate to each other and spatial concepts, and, if persistent, allow the robot to reason beyond its sensory horizon.

One may argue that robots and computers can already represent and understand environments, but they do so by using representations that are not human friendly: e.g. a robot can uniquely identify the location of a certain landmark using internal coordinates, but those coordinates are not understandable for humans. A robot may be able to detect a tag placed on an object, but such an approach does not scale outside structured environments: new objects are created everyday, there is huge amount of already existing objects and concepts by their nature are dynamic. For this reason, methods need to be developed that will allow robots to represent real-world environments and allow human concepts to be transferred into the robot representations.

By gathering semantic knowledge a robot can learn how to augment the internal representation with a set of human concepts. Web and other common databases, that have been created by humans, are a great source of semantic information, and can be a valuable source of knowledge for the semantic mapping problem. Common approaches to semantic mapping are based on classification, i.e., a robot has a set of defined categories and selects the one that best represents the sensed data. This approach often gives satisfactory results, but only when applied in a controlled environment where the robot has knowledge about all of the categories that exist.

This assumption about the complete knowledge of all the categories that an entity can be mapped to is too strong to hold on real-world environments. More specifically, a mapping defined over the human semantics will always be incomplete due to the infeasibility to map all the concepts or even due to its dynamic nature where concepts change not only over time but also between sociocultural aspects.

Consequently, a robot should be able to identify gaps in its own knowledge and detect novelty. Developing methods for this problem helps to provide robots with novelty signals that can be used to improve reliability, decide when to initiate active learning or even provide basic information for keeping knowledge continuously adapted in long-term or life-long scenarios.

The rest of this chapter presents the context in which this thesis addresses the problem of novelty detection. Then, it presents the specifics of the problem and defines the goals to achieve. Finally, it outlines the structure of the remaining chapters.

1.1 Context

One of the most crucial problems in mobile robotics is that of representing space in which the robot operates. Having a representation of all the spatial entities and relations between them allows the robot to navigate and interact with the environment. Maps are the default tool for representing those spaces, and computers excel at creating close to exact representations of spatial environments. However, those representations usually fail to capture aspects important for human-like spatial understanding.

Different spatial knowledge types can be identified depending on the scale, abstraction level, and required generalization. Examples include geometric aspects of the world, object knowledge (which is fundamental for humans, but difficult to extract for robots), segmentation of spatial areas, and finally, relations between different spatial entities e.g. a specific object (book) is on top of another object (table) that is located inside an area (room-library) that is inside a another area (university). We see that humans do not limit themselves to representing entities of space and relating them spatially, but they also attribute semantic concept to them.

It is those semantic concepts attributed by humans to spatial entities that are interesting for endowing robots with knowledge on how to interpret and reason about human environments. By understanding which objects are expected to be found in a room, what are the properties that distinguish room categories, or which room types are likely to be connected together, a robot can increase its communication abilities but also its performance on complex tasks in indoor environments.

A basic concept of indoor environments is that of a room. Rooms allow humans to segment areas into high-level entities that can be categorized according to their properties and canonical activities performed there, e.g. kitchens are rooms where cornflakes are usually found, a library is identified by the presence of many books, a lecture hall is a place where lessons are given. Since room categories play an important role in understanding and reasoning about indoor human environments this thesis will focus on them.

1.2 Motivation

It is highly desirable that a robot can be deployed in new and unknown environments. In order to achieve that, the robot has to perform its own semantic mapping of the environment instead of relying on available labelled maps. However, the semantic mapping ability of a robot is restricted by its spatial and semantic knowledge. This poses a problem as its unrealistic to assume that in an unknown environment, the robot's knowledge will be complete. Therefore, it becomes important to detect gaps in the robot's knowledge so that the robot can be aware of the limitations of its models and deal with the novel situations. Furthermore, the detection of knowledge gaps can be extremely useful for several other problems in artificial intelligence such as active learning and knowledge maintenance during life-long operation.

1.3 Goals

This thesis aims at developing methods that can be used to detect gaps in spatial semantic knowledge of a robotic agent. To this end, the spatial knowledge representation proposed in [PSA$^+$10] is used as a base for defining the agent's spatial knowledge. Given a concrete environment, a semantic mapping process is applied and a conceptual map representation of the environment it build according to [PJ11]. The conceptual map is represented in terms of a probabilistic graphical model capturing both semantic and structural information of space. This thesis focuses on novelty detection methods from this probabilistic graphical model representation. In particular, emphasis is placed on detection of novel room categories given the observed spatial properties of the environment.

1.4 Thesis Outline

The rest of this thesis is structured as follows:

Chapter 2 introduces fundamental concepts important for understanding the problems addressed in the thesis. In particular section 2.5 introduces *probabilistic graphical models* that lay the foundation for the conceptual map.

Chapter 3 describes the semantic mapping process. It introduces the spatial knowledge representation and describes a system that uses it for performing semantic mapping using using the *probabilistic graphical model*.

Chapter 4 introduces novelty detection from the statistical point of view and shows how to interpret it as a thresholding function. Then, it the main contribution of this thesis: methods for detecting novel semantic categories using graphical models.

Chapter 5 draws conclusions on the developed work and presents possible directions for future works.

Chapter 2

Background

One of the interesting facts about high-level problems is that they often require a broad overview of the whole system and several concepts. For example in the case of this thesis approach to novelty detection on semantic knowledge it becomes first important to understand how classification of already known categories is performed. I.e. understand how can lower-level classifiers be represented, how to obtain those representations from training data, and how to use those them to infer new classifications from those models. It is also important to know how to model several of those classifications with probabilistic relations in a unified model and how to infer information on those.

This chapter tries to briefly introduce the reader to those aspects. The presented material is not complete, it serves only to give directions and cues to the reader on how the several subproblems of visual place classification can be solved. Where appropriate it refers to articles or textbooks that present the introduced techniques.

2.1 Classification

Classification deals with the problem of identifying classes or groups that lie underneath the sensed data. It is expected that the presence of a underlying concept is involved on the generation of the data that is sensed, and the system tries to infer that underlying concept without directly observe it.

Often the sensed data is just to large to be directly handled within a classifier. In those cases classifiers work on a small subset of features extracted from the data. Those features should avoid discard important information and if correctly employed should turn the classification easier.

2.1.1 Recognition and Categorization

Classification can target very different types of classes, as it is not clear on which classes the system is interested in distinguish. For example the system maybe interested in distinguish specific instances or interested in detecting a wide group of instances matching a common category. E.g. distinguish a specific room: room 304 in the 3rd floor, from a generic room category: a library.

Based on the type of desired learning system, the available features and the required generalization specifications a wide range of machine-learning methods can be used. And no single method is expected to handle all the problems with optimal performance. The methods can go from statistical classification, neural networks, nearest neighbours, decision trees, support vector machines to others like graphical models, clustering, Gaussian mixture models, hidden Markov models, principal component analysis [Bis06].

This background chapter does not aims at describing all the classification methods, but at introducing the concept of classifier and to use them to extract information from low-level features. In that sense only Support Vector Machines (SVMs) will be generally described, for more information on others the reader is welcome to check the vast literature in pattern recognition and machine learning such as the standard textbook cited above.

2.1.2 Supervised and Unsupervised Training

It is often impossible to manually specify the information needed to correctly classify samples. There is also interest to make systems flexible and allow them to learn and adapt based solely on the available data. In that sense the option is to train classifiers from available data. The training methods are often separated as:

Supervised Methods assume the existent of a supervisor that is able to give the ground-truth. With it the algorithm tries to learn the best description that matches the given labelled data.

Unsupervised Methods try to learn classes without any extra information. The methods have to figure out how many and which classes seem more reasonable to be modelled.

2.1.3 Discriminative and Generative Models

After constructing a classifier from the information available either in form of available knowledge on the task in hand and from acquired data samples, the agent ends up with a model that represents the knowledge it believes to be suitable to solve the task.

The produced model differs a lot based on the type of classifier but they can be distinguished in two different categories:

Discriminative models are only able to classify samples in the known categories.

Generative models model the full probability relation between sensed features and classes. With that it becomes possible to use the learned model to generate new samples.

2.1.4 Multi-Class Classification

Most classifiers methods are designed as two-class classifier and do not natively support multi-class classification problems. The most common approach is to try to approach multi-class problems by by combining several two-class classifiers and perform a voting scheme or other integrating method based on the confidence of the individual two-class classifiers. For example:

One Against All - in this method c distinct classifiers are trained to distinguish any class of the remaining ones. The output of all those classifiers (distance to the separating hyperplane) is then used to categorize the output. The most common approach is to pick the class with the largest hyperplane distance. Other variations exists as is the case of using the minimal distance to the average classification distance of each class [PC07].

One Against One - in this method $c*(c-1)/2$ classifiers are trained to distinguish between each pair of classes. The final decision is based on the output of all those classifiers being common to use a majority vote strategy.

2.1.5 Support Vector Machines

SVMs where introduced by [CV95] and can be seen as discriminative linear based classifier. They are based on a strong mathematical foundation and have powerful generalization capabilities. In their original form SVM separates two classes of points in an hyper-space with a *maximal margin hyperplane* (Figure 2.1). Later they were extended to deal with noisy data by using *soft margins*. And to handle non-linear spaces as seen on subsection 2.1.6.

They have been used in several classification and recognition problems and are in fact a standard across machine learning techniques. Their efficiency, exact training results and generalization made them suitable for many tasks. Such as text categorization, digit-recognition, spam-classification. They have also been extensively used in visual place classification.

2.1.6 Kernel-Trick

Several classifiers can be modelled by requiring only the concept of inner-product between two samples. That product is often seen as a measure of distance or similarity between the samples on some space. By using kernels it becomes possible to define such

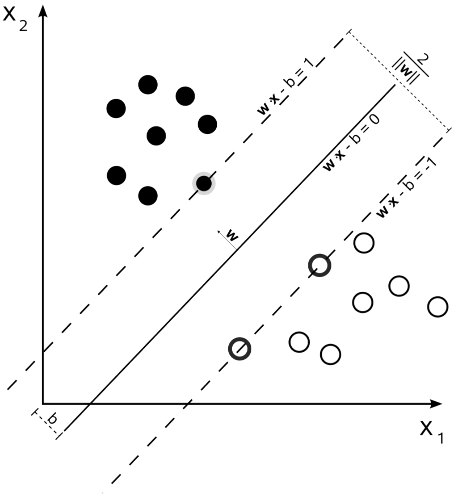


Figure 2.1: SVM separating two class of points by a *maximal margin hyperplane*. The hyperplane can be described by the collection of support vectors and associated weights, marked in the image as sample points with large borders.

space without explicit convert the samples to it. By abstracting the concept of distance on such a kernel function it also becomes possible to use those classification methods on strange data structures such as trees, strings or graphs.

For example SVMs, in their basic form, are only able to handle linear spaces. But the classes are most of the time not linearly separable in the input space. Although there might exist a transformation ϕ from the original space into a space H where the input becomes linearly separable.

The Kernel trick allows to extend the SVM definition to work on such space H without ever performing an implicit transformation between spaces. Being enough for that to have a Kernel function defining an inner-product inside such space: $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$.

Several kernel functions have been proposed being the most commonly used:

Polynomial Kernel - $K(x,y) = (x \cdot y + p)^d$

Radial Basis Function - $K(x,y) = e^{-\gamma||x-y||^2}$

Histogram Intersection - $K(x,y) = e^{-\gamma\chi^2(x,y)}$, where $\chi^2(x,y) = \sum_{i=1}^{N} \frac{(x_i-y_i)}{x_i+y_i}$ introduced by [BOV03] allows to compute histogram similarity.

Matching Kernels - mimic matching similarity and are used when each sample is represented as a set of features [BTB05].

2.2 Features

A feature is a piece of information which is expected to reveal information for solving a specific task. Features are task-dependant and they will yield different performance based on the type of task they are applied to.

A wanted property on features is its repeatability under similar conditions for the problem in hand. This is: they should be stable and invariant across unwanted types of transformations and noise. For example a visual feature for object detection should be present even if the target object was translated, scaled, rotated, the light-conditions have changed or even if the object is partially occluded.

Extracting features with those properties allows to greatly reduce the size of input by removing unwanted noise and useless information from the captured data. Turning the classification problem easier, more reliable and more efficient.

Often several and different types of features need to be extracted. It has been reported by [PMCJ10] that using multiple features provides a great benefit in the context of place classification. And [QT09] has showed that different types have different impact in indoor scene recognition based on the type of scene matching. Specifically it was seen that some room-categories are more likely to be recognized by the presence of some objects and others by it generic appearance.

In the context of robotics, sensors such as cameras, laser scans are used to sense the surrounding environment, and features can be extracted from those. Visual features can be seen as belonging to two categories: *local features* and *global features*.

2.2.1 Local Features

Local features describe fine grain properties of a part of image. For example the existence of specific corner or an edge. Scalar Invariant Feature Transform (SIFT) is an example of such feature and it been proven useful for matching points between images and subsequence extension to object detection.

Interesting Points and SIFT

The detection of interesting points has been studied for several years and is the base of several computer vision problems solution. It allows to perform point matching which can be used in several areas from image stitching, 3D reconstruction, video tracking, object detection, etc…

The most used method was presented by [Low99] and its based on building a feature vector for each image. Each of those features is based on *interesting points* detected by the presence of maxima and minima of difference of Gaussian functions applied in a scale-space. The scale space is used to provide scale invariant detection. Gaussian functions are used as they are the only way to model a linear scale-space. Each interesting point is then described by a container that is rotation invariant.

By seeing an object as a set of features points, index and matching is then performed by a high-dimensional search on a database of know objects. After matching objects can be verified for geometric coherence between features. SVM classifiers can also be trained to detect objects based on this type of local features by using *Matching kernels* (subsection 2.1.6).

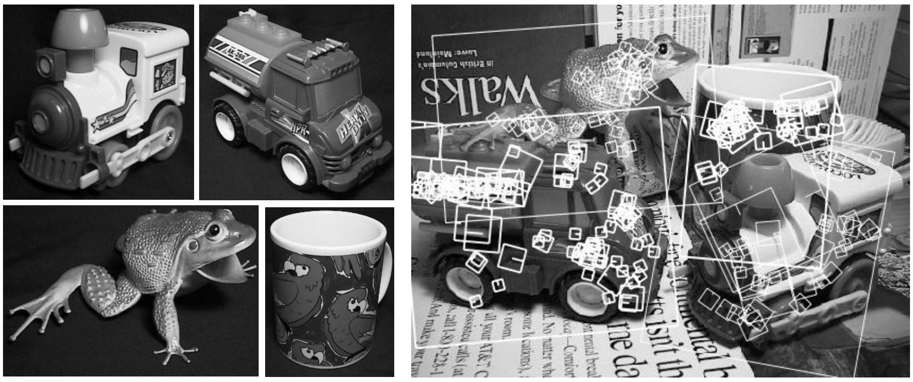


Figure 2.2: SIFT and other local features have been proven useful in object detection.

2.2.2 Global Features

Global features try to describe the whole image. E.g. either by statistical analysis of features over all the image or by a structured distribution of textures findable in the image.

Gist of a Scene

Oliva and Torralba [OT06] argue that fast scene recognition does not need to be built on top of object recognition but can be analyzed by scene-centered mechanisms. They defend that position by pointing out behaviours on human vision: when provided with a glance of a shot a person can identify the meaning of that given shot or "gist of a scene" without remembering specific details.

As seen on Figure 2.3 the gist is able to capture the dominant textural features of the overall image and their coarse spatial layout [MTEF06]. With that, it is expected they serve the purpose of correctly describe the image textures without the need to directly stored the original image.

CRFH - Composed Receptive Field Histograms

Composed Receptive Field Histogram (CRFH) are a multidimensional statistical representation of the occurrence of several image descriptors applied to an image. They can be

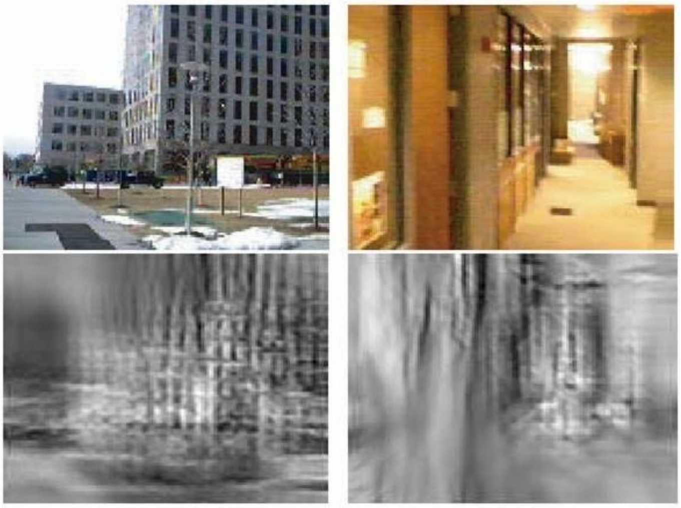


Figure 2.3: An illustration of the gist of an image. Top row: original image I; bottom row: noise image J for which gist(I) = gist(J).

seen as an high-dimension histogram where each cell records how many pixels of the image have the cell response for the applied descriptors. Such high-dimensional histogram is expected to be able to describe global information contained in the image by capturing several properties that co-occur in a part of the image. By using some techniques [LL04] several operations on those high-dimensional histograms can be made computational efficient. This way not only this descriptor discards part of the local information present on the image but also allows to faster computations on similarity measures.

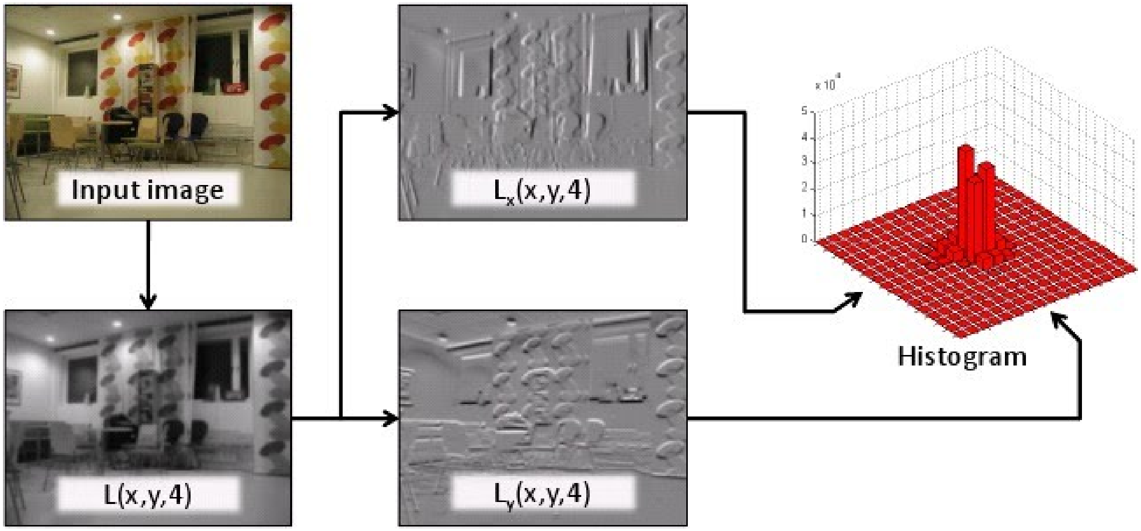


Figure 2.4: A two dimensional histogram of the image built out from two image descriptors: *Lx* and *Ly*. First-order Gaussian derivatives of image luminance in horizontal and vertical direction applied at a scale 4.

Multidimensional histograms have proven to be useful in the context of object recognition [SC96]. And have also been previously used in the context of visual place classification [PMCJ10]. *Histogram intersection kernels* (subsection 2.1.6) can be used to train classifiers based on this feature.

2.3 Probability Theory

An important aspect when making decisions with noisy or missing information is the uncertainty of the received information and of the final conclusion. Probability theory comes to help by providing a framework able to deal with those issues [Bis06].

The key concept on it is that of *probability*, which can be seen as way of express the degree of belief that a certain event occurs. When combined with decision theory it allows to perform optimal decisions with the available information, even if it is ambiguous or incomplete.

A probability $P(x)$ of a given event x is a value between 0 and 1, where 0 means the belief that x will not occur and 1 that it will certainly occur. In cases x does not specifies all the variables states that define the sample space, $P(x)$ is called a marginal probability.

Often it is impossible to exactly describe the probability of a given event, in such cases the concept of likelihood comes to help. The likelihood of an event only has meaning when compared to one of another event. In such case the ratio between both likelihoods will denote on how more likely an event is over another. When dealing with discrete and countable sample spaces any likelihood function can be converted to a probability function by calculating the normalization factor such that the sum over all the sample space is 1.

2.3.1 Conditional Probability

It is also possible to denote the known information when calculating probabilities. For that the notion of conditional probability is used $P(a|b)$. It denotes the probability of a knowing that b has occur. Often conditional probability is also represented by: $P_b(a)$. In this thesis both notations will be used: the former for representing the information on sensed variables and the second for representing information or assumptions on other generic information such as graph structures.

Marginal probabilities can be related with conditional probabilities as described in the equation:

$$P(a|b)P(b) = P(a,b) \tag{2.1}$$

2.4 Principle of Maximum Entropy

The principle of maximum entropy [SJ80] states that when given a set of distributions that are coherent with the acquired knowledge, the one which maximizes entropy should be picked.

This can be used to select a distributions that most correctly describes the obtained knowledge. For example, if all that is known from a distribution is the mean and deviation then the correct approach is to model it with a normal distribution with the known parameters. In case of having no information available, the uniform distribution shall be assumed.

2.5 Probabilistic Graphical Models

Graphical models usage can be tracked back to earlies 1920 but they only become popular in mid-eighties when researchers started to use *Bayesian Networks* to model expert systems [BK02].

They serve as a better tool to model *random variables* (nodes on the graph) and their probabilities as they model the conditional dependence between variables (edges on the graph). Important to note that here *random variables* does not denotes a truly random variable but one that is unknown by the system and is conditioned by other variables/evidences.

This type of graphs provide a generative model where the probability of any given scenario can be determined. This means once a graphical model is learned, it can be used to generate new samples from the learned distribution.

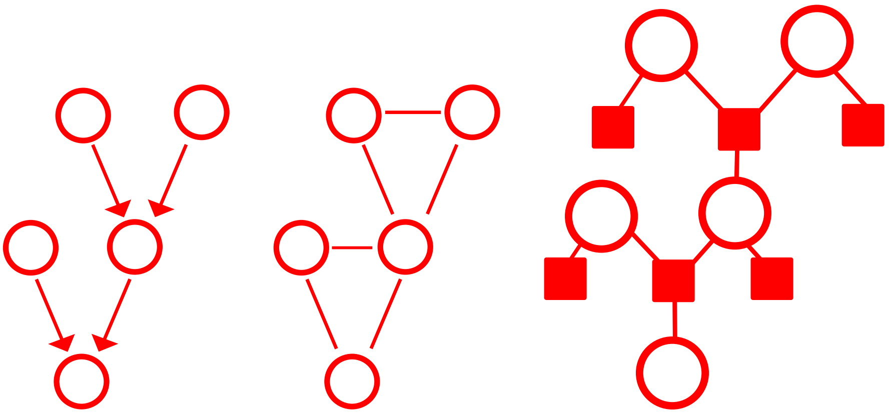
They have been successfully used in several machine-learning task such as: information extraction, speech recognition, computer vision. They are also useful due to their ability to deal with semantic (high-level) features [BLB06] and ability to represent properties of the reality they try to model.

Two main types of graphical models are widely used: Bayesian Networks which model directed edges between variables and Markov Random Fields where variables are connected by a potential but no special direction is given to edges.

An important property of these graphs is the *Markov-blanket* of a node. For a given variable *a*, a *Markov-blanket* is a set of variables in the surroundings of *a* that when given the value of *a* becomes independent of the rest of the graph [Pea88]. On non-directional graphs it is directly determined by the nodes connected to *a*. This allows the usage of graph algorithms such as *min-cut* to quickly determine most likely scenarios. In the case of directed edges a node blanket is also influenced by the direction of the edges and more complex schemes need to be used.

As [LR02] points this two types of models can be represented as a *chain graph* where both directed and undirected edges can co-exist. Though this generalization is hard to implement due to mis-understandings on the concepts the graph-models use.

Another useful interpretation of graphical-models are *factor graphs*. Those are able to handle both *Bayesian Networks* and *Markov Random Fields*. Under this interpretation a graph is seen as a bipartite graph that connects variables with factors that influence them [Bis06]. This gives a very useful framework to develop belief propagation on them by seeing a message-passing mechanism between nodes. Belief propagation is used to calculate marginal-probabilities.



(a) Bayes Networks

(b) Markov Random Field

(c) Factor Graph

2.5.1 Factor Graphs

A *factor graph* [KFL01] is a bipartite graph connecting two sets of nodes X_G and F_G representing random variables and factors. Each factor is described by function ϕ dependent only on the variables x_ϕ to which the factor is connected. Thus, a factor graph can be seen as a description of probability density function obtained by a product of all the factors. In order to represent the probability, a normalization factor Z needs to be introduced, resulting in the following equation:

$$P_G(x) = \frac{1}{Z} \prod_{\phi \in F_G} \phi(x_\phi), \qquad Z = \sum_{X_G} \prod_{\phi \in F_G} \phi(x_\phi) \tag{2.2}$$

2.5.2 Inference

The goal by obtaining a graphical model between all the variables of a system is the ability to efficiently infer the most likely scenarios based on the available information. The sensed information can be used to clamp variables and the model used to calculate probabilities or marginal probabilities.

In that sense a basic operation on a graph is the Maximum a Posteriori (MAP) operation, which allows to calculate which configuration on a variable or subset of variables maximizes a given function.

2.6 Summary

This chapter has introduced concepts used as base during the process of semantic mapping. For that, it introduced classification tasks in a generic way, it explains the difference between supervised and unsupervised approaches, how to obtain multi-class classifiers on top of two class classifications, and how to perform those operations in implicit spaces by using a kernel function. It also presented both local and global visual features that are often used in computer vision for either detecting objects or classifying appearances. Some probability theory bases were presented together with graphical models that allow to perform inference on several variables by creating a model of how variables directly influence each others.

All this concepts were presented and papers and standard text books referenced, with the purpose to make the reader familiar with the tools that are used to extract the higher level features that the semantic mapping process and how does the process infers semantic labels on the detected areas.

Chapter 3

Semantic Mapping

Humans associate concepts to areas that relate to their functionality. That association is either useful by explaining where to find certain objects as well to understand the types of activities to perform on certain areas. Those human-concepts are meant to be meaningful for humans and by incorporating conceptual knowledge about them it is expected to improve the agent ability to interact and communicate with humans and to perform human-like tasks.

In this context semantic mapping is seen as the process of building a representation of the environment which associates spatial entities with a set of defined spatial concepts (in this case human concepts).

In order to build such a semantic map, it is important to define the spatial and conceptual knowledge that allows an agent to map between its machine-friendly representation of the world and the high-level semantics characteristic of the human world. That knowledge needs to define and describe how to segment and identify concepts from the low-level representation. It also needs to define how the concepts relate to each other and attribute meanings to them. In the case of indoor environments this can be seen for example as: rooms are a unit of division of area and are often separated by doors, each room can also be characterized by the objects inside, size and shape, and they can be categorized in meaningful concepts such as: kitchen, library, corridor, etc...

The low-level representation and the knowledge of the agent, will always be to a certain a degree imperfect, incomplete, inaccurate and invalid. Due to that both the concepts and their relations cannot be quantified in a rule base system and ought to be treated in a probabilistic way in order to allow the agent to still reason with some accuracy about the world.

This thesis motivation lies on developing methods for detecting knowledge gaps on the spatial semantic knowledge of a mobile agent. For that, spatial knowledge needs to be defined, as well the methods used to perform the semantic mapping. In that context this chapter presents Dora: The Explorer (Dora), a system that represents its spatial knowledge with the representation proposed by [PSA$^+$10], and performs semantic mapping using inference on a probabilistic graphical model.

3.1 Architecture Overview

Pronobis [Pro11] presents a system architecture working in indoor environments using non-omnidirectional laser and visual sensors for semantic mapping. The system integrates cues such as geometry, object presence and appearances and is able to perform inference across any semantic properties, for example for the purpose of place categorization. The introduced system is built around a spatial knowledge representation [PSA$^+$10] and it has not only been tested on real-scenarios and performance tested across several conditions [PC07] but also shown to be tailored to effectively solve tasks arising on mobile robotics [HGD$^+$11].

The *spatial knowledge representation* has been drawn with an high-focus on probabilistic and uncertain reasoning, human interaction and life-long learning capabilities. For that it was considered an excellent base defining not only the concepts and requirements of an high-level semantic representation but also the information flow between all the layers involved on the agent.

Dora, the system, utilizes its spatial knowledge to perform semantic mapping of the environment using probabilist inferences on top of a graphical model. Due to its nature, it was considered suitable for implementing novelty detection of semantic concepts.

3.2 Spatial Knowledge Representation

[PSA$^+$10] proposes a spatial knowledge representation sub-divided into four layers. Each layer focus on different aspects of the world, abstractions levels of the spatial knowledge and different spatial scales. Each uses different spatial entities and relate to the agent goals in different ways:

In the lowest abstraction level the sensory layer represents the most immediate and short term accurate representation of the world. Above, the place layer discretizes the continuous space into. The categorical layer focus on knowledge of low-level, long-term categorical models of the sensory information. And at the higher level the conceptual map associates concepts to areas such as rooms (see Figure 3.1).

3.2.1 Sensory Layer

The sensory layer keeps a detailed representation of the immediate surrounds of the agent. That representation is based on directly sensed inputs as well on data-fusion for short time-intervals.

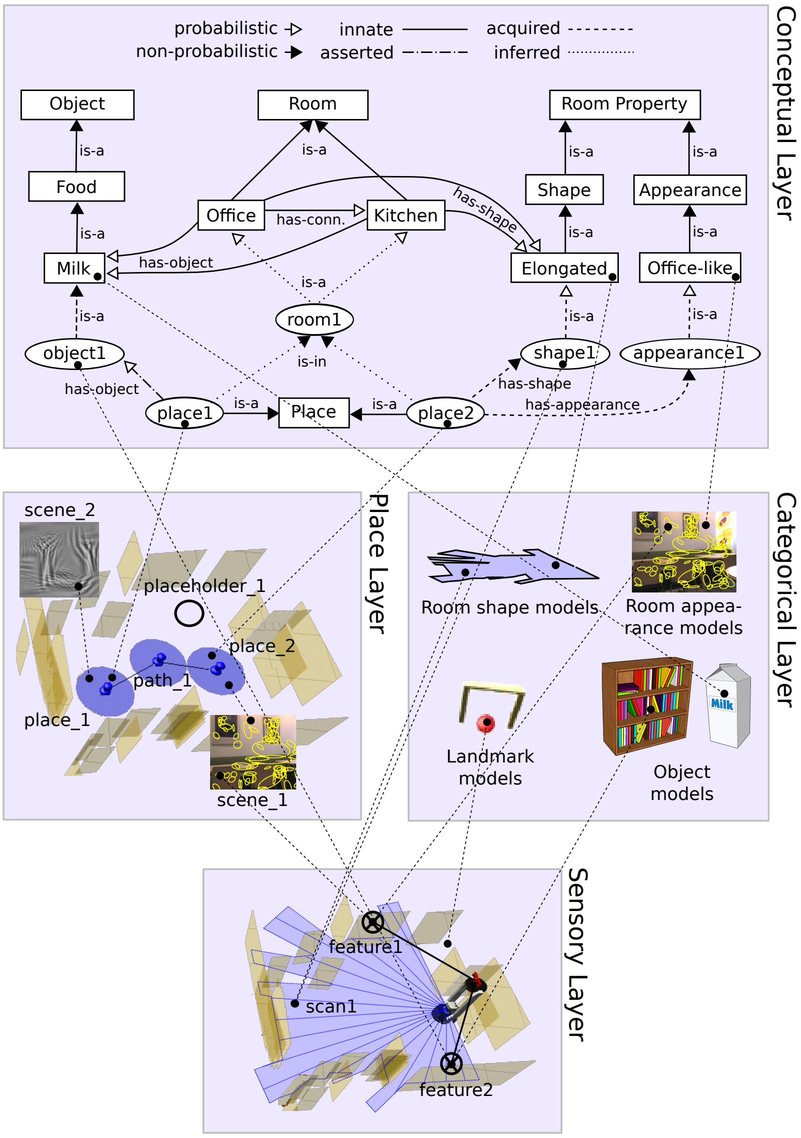


Figure 3.1: Layered structure of the spatial representation.

It is responsible for providing the agent with an exact position and to accurately position low-features and landmarks. The information is stored with associated uncertainty it is highly susceptible to be replaced with newer versions. Also old or distant information is forgotten. By tracking the surrounds for short amount of times it provides an accurate representation that can be used to locate and guide the agent on low-level movements.

3.2.2 Place Layer

The place layer is responsible for maintaining a bottom-up discretization of the continuous space. On it the world is represented as a collection of discrete spatial entities, called places, together with connectivity information.

Each place is defined with the features from the sensory layer and with spatial relations to other places. Connectivity here, stands for ability to travel directly between the places. Additionally places can be created for areas not yet explored allowing to introduce virtual place holders to places that would eventually be uncover if the agent moved there.

The places are also considered stable on long-term and can be used to help the agent performing localization and planning longer distance motion planning. This higher level representation although uncertain allows to model connectivity and relaxed spatial relations.

3.2.3 Categorical Layer

The categorical layer holds the generic long-term, low-level representations of categorical models of the agent's sensory information. The knowledge represented by this layer is not instance specific. It is the knowledge required to determine certain features indicate the presence of an entity of certain category. E.g. the knowledge required to detect objects, landmarks or describe room appearances. Other properties may as well be defined for example shape, color, edges and size properties are all defined in terms of low-level features.

This layer is responsible for representing knowledge on how to extract properties used by the conceptual layer. Although not mandatory, in many cases the categories represented by it will map to human concepts and not to the internal concepts of the agent. The type of properties handled by this layer may require incredible complex models and for that they might be trained on a supervised fashion.

3.2.4 Conceptual Layer

The conceptual layer provides an ontology that represents a taxonomy of the spatial concepts and their relations with the properties represented on the categorical layer. This associates semantic meaning to those properties that is useful for human-agent iteration and provides relations that can be verbalized and explained with the ontology.

It provides also knowledge between the semantic concepts and their instances of those concepts. Including definitions of spatial concepts related to space segmentation as well to semantic categorization of those spatial entities. E.g. rooms are commonly split by doors, rooms exist in a floor, a building has floors, milk is likely to be found on kitchens, etc...

By providing information for segmentation, semantic mapping to concepts and relations between those concepts, the knowledge represented on this layer allows the agent to explain and reason close to the human concepts. The knowledge represented here is considered to be valid through very long periods of time or even life-long.

3.3 System Structure

The Dora system [HGD+11] consists of several co-operating sub-systems (see Figure 3.2), all of which actively use or maintain the *spatial knowledge representation*.

The layer structure of the spatial knowledge representation allow to implement data driven processes that control the flow of knowledge and information between the lower and higher-levels layers. To reduce computation and make it feasible not all updates from lower-levels are treated immediately. Instead the system waits until a substantial change has occurred.

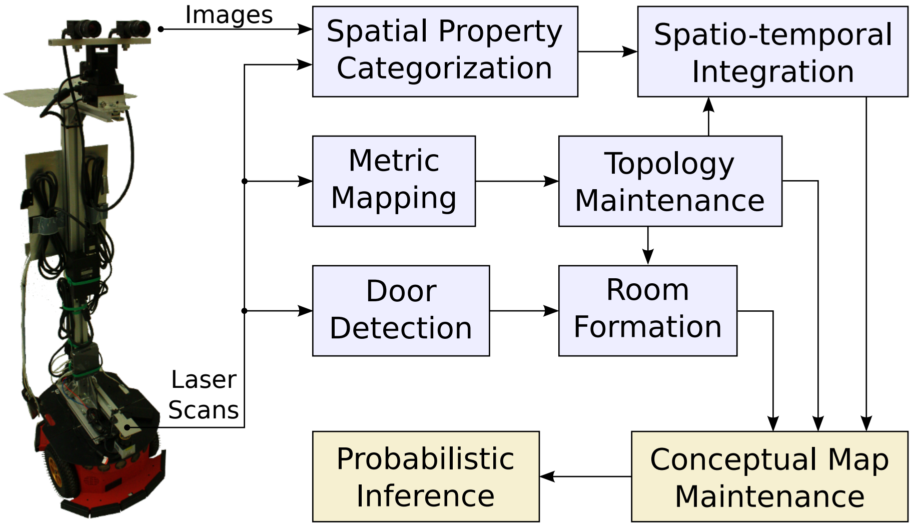


Figure 3.2: Dora the Explorer as well as the elements and the data flow inside the semantic mapping system.

Initially the low-level topology and metric mapping systems use a Simultaneous Localization and Mapping (SLAM) algorithm [FJC07] that is used as representation on the sensory layer. Those two systems are also responsible for creating the place map on the place layer, as that is highly related with the SLAM goals. By using the landmarks (doors) and other available information the system also tries to group places and understand the concept of rooms.

At the same time the spatial property categorization system is responsible for detecting instances of the categorical knowledge by running classifiers previously trained to detect a set of features listed on subsection 3.3.1. For that it uses the data available from a laser range finder and a camera and besides classifying the features it estimates also its confidence on the information. The extracted features together with the estimates is then integrated over time and space in order to increase its robustness.

Finally the conceptual map system utilizes the acquired instance knowledge together with the high-level conceptual knowledge to create a probabilistic graphical model that allows to run inferences. Since this conceptual mapping processing is responsible for creating the probabilistic graphical model it is described in detail in section 3.4.

3.3.1 Features

The *categorical layer* requires classifiers with the ability to categorize the low-level features into categories. For that Dora performs detection on the following categorical entities:

Doorway Detection by using the laser range finder the system tries to detect doors.

Geometric Shape by using features extracted from the laser data and pre-trained classifiers the system tries to classify rooms according to their shapes (e.g. rectangular, square).

Geometric Size is classified once again recurring to laser data. In this case the system tries to categorize the room size according to a set of previously learned concepts (e.g. large, medium, small).

Object Detection is performed by running traditional object recognition classifiers. Those are previously trained on a supervised fashion to later be able to detect objects based on their visual features, e.g. by using SIFT as described in section 2.2.1.

Visual Appearance is extracted from the visual input. For that Dora uses a trained a set of classifiers on global visual features such as CRFH to categorize the visual appearance of a room in categories such as corridor, office, library.

3.4 Conceptual Map

As Dora moves through the environment it builds a *conceptual map*: a structural and probabilistic representation of the space instantiated as a *graphical model*. It includes taxonomy of human-compatible spatial concepts which are linked to the sensed instances of these concepts drawn from lower layers. It is the conceptual layer which contains the information that kitchens commonly contain cereal boxes and have certain general appearance and allows the robot to infer that the cornflakes box in front of the robot makes it more likely that the current room is a kitchen. The conceptual layer is described in terms of a probabilistic ontology defining spatial concepts and linking those concepts to instances of spatial entities (see the example of the ontology in Figure 3.1).

Based on this design, a *chain graph* model is proposed as a representation for performing inferences on the knowledge represented in the conceptual layer. Chain graphs are probabilistic graphical models that combine the properties of both Bayesian Networks and Random Markov Fields. This results in an efficient approach to probabilistic modeling and reasoning about conceptual knowledge.

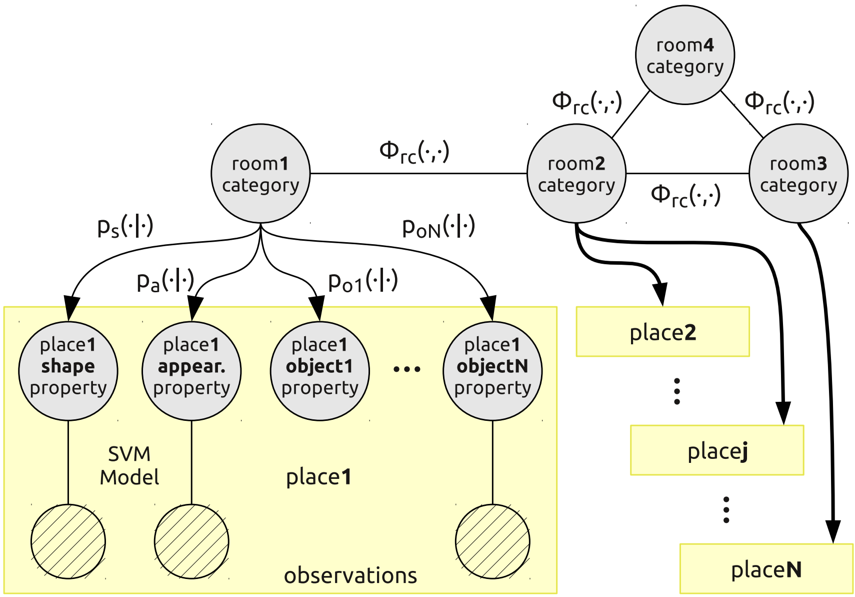


Figure 3.3: Example chain-graph instantiated from the *conceptual layer*.

An exemplary chain graph corresponding to the conceptual map ontology is presented in Figure 3.3. Each discrete place identified in the environment is represented by a set of random variables, one for each class of relation linked to that place. These are each connected to a random variable over the categories of rooms, representing the “is-a” relation between rooms and their categories. Moreover, the room category variables are connected by undirected links to one another according to the topological map. The remaining variables represent: shape and appearance properties of space as observed from each place, and the presence of objects. These are connected to observations of features extracted directly from the sensory input. Finally, the distributions $p_s(\cdot|\cdot)$, $p_a(\cdot|\cdot)$, $p_{o_i}(\cdot|\cdot)$ represent the common sense knowledge about shape, appearance, and object co-occurrence, respectively. They allow for inference about other properties and room categories e.g. that the room is likely to be a kitchen, because you are likely to have observed cornflakes in it.

The use of graphical models to describe distributions of variables has useful properties. First, they permit inference about uncertain conceptual knowledge. At the same time, they are generative models and therefore allow to calculate the probability on any given subset of variables of the graph, allowing the system to work even when some information is missing.

3.4.1 Factor Graphs

Although the conceptual layer works with *chain graphs* [LR02], those can be converted into *factor graphs* [KFL01]. Factor graphs have been introduced in section 2.5 and are used throughout this thesis as they provide an easier manipulation due to factorization.

Moreover, there exist efficient implementations of inference engines operating on factor graph representations [Moo10]. Describing the distribution function in terms of graphs permits the use of those engines to efficently calculate marginals on any given subset of variables by exploiting conditional independence between variables.

3.4.2 Uncertain Sensing and Implicit Factors

In a realistic scenario, classification tasks are, by their nature, uncertain. And although a trained classifier is able to decide which class is more likely, there is interest in handling uncertainty produced by those to make decisions more robust.

By modelling the classifiers as an implicit function $\phi_{classifier}(c,x)$ it becomes possible to model the uncertainty of the decision given the presence of the sensed low-level features x. The presence of those implicit models no longer allow calculation of the normalization factor over the whole graph as there is no explicit model for either the low-level features X (e.g. X can be an image received from a visual sensor) or for $\phi_{classifier}$ (e.g. the classifier can be an SVM).

Nonetheless it is still possible to calculate any probability given that the features X are observed. For that instead of modelling the variables X and explicitly describing the factor $\phi_{classifier}$ it is enough to replace it by an observed factor $\phi_{classifier}(c|x)$ as shown on Figure 3.4.

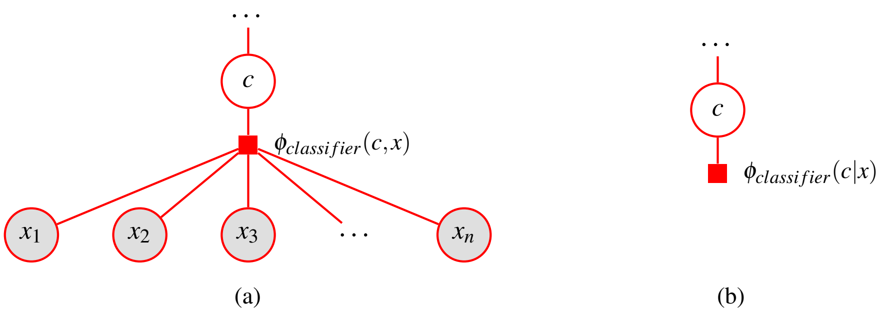


Figure 3.4: By modelling classifiers as implicit factors it is possible to perform inference on the graphical models handling the uncertainty measures produced by those classifiers.

3.5 Summary

This chapter has introduced the semantic mapping process that is used as base platform where to perform detection of novel semantic place categories. It presented the spatial knowledge representation used by it, and shows how the knowledge is used during the mapping process to identify entities and classify places according to a set of human defined semantics.

Chapter 4

Novelty Detection

This chapter presents the main work of this thesis: a method to augment the conceptual map with novelty detection capabilities.

The chapter starts by providing the reader with a brief overview on novelty detection techniques and related works. Then it shows that an optimal novelty detection system can be implemented by thresholding an order relation defined over the inputs. Moreover, an equivalent order relation can be imposed by a ratio between conditional and unconditional probabilities.

Finally, a practical example using semantic data and probabilistic graphical models is presented: it shows how to use the introduced ratio to obtain a novelty detection system and analyses the performance impact by increasing the amount of sensed data and by using an approximation on the unconditional probability.

4.1 Novelty Detection - Review and Related Work

Novelty detection, also known as outlier or anomaly detection, is a classification problem related to identification of new or unknown data patterns that the system is not aware of [MS03].

The ability to identify novel cases is crucial in any autonomous system that is deployed to unknown or to uncontrolled environments, as it gives the system the ability to detect that something is not conforming to its knowledge and therefore should be treated with caution. It has several applications such as fault detection [TNTC99], intrusion detection [FMS$^+$01], detection of masses in mammograms [THCB95] or detection of novel and useful documents [ZCM02].

Any normal classification application is also a good candidate for extending with novelty detection, as the results for samples on which the system has not been trained on can be unreliable [DMO08] e.g. when applied to digit-recognition [TD98] or to detection of novel inputs on a neural network [Bis94].

It is in the nature of unknown environments and in the infeasibility of the system to acquire examples of all possible classes where the complexity of novelty detection lies: it is only possible to obtain samples representing positive examples of known cases and the lack of negative examples renders normal classification methods unusable. In order to become practical in real world applications, novelty detection methods have to overcome a series of obstacles: be able to generalize while still detecting novelty, be resistant to noisy features, scale in feature dimension, deal with multiple classes and perform detection efficiently as many autonomous systems require real-time or close to real-time performance.

4.1.1 Review of Novelty Detection Methods

A common approach is to use density estimation and use the expected probability of a sample to classify the sample as novel. Examples of such techniques are Gaussian Mixture Models and Parzen-window estimators. In order to be effective, those rely on data as close as possible to the input features and use dimensionality-reduction techniques such as Principal Component Analysis (PCA) to make density estimation feasible. Note that as dimension increases an exponential number of data samples would be required to approach the density with the same quality.

[Bis94] uses that approach by employing a Parzen-window to estimate the density of the training data on a given input fed into a neural-network. Using the calculated density, they detect samples that differ from the training data and consider their neural-network output to be unreliable as the samples are distinct from what the network was trained with.

A slightly different approach is applied in case of by one-class SVM approaches that try to distinguish novelty by separating the training set from all the other points in the input space. This is achieved by enclosing the training set with some structure (e.g. an hyper-sphere) [BC00]. Those approaches have been made in line with Vapnik's principle of never to solve a problem which is more general than the one we actually need to solve [SWS$^+$00]. Although having access to a perfect probability distribution of the input would solve the problem, creating such a function is harder than simply creating a boundary between known data and novel data.

Error reconstruction methods have also been used for novelty detection. They use the assumption that the class to be defined lies on a manifold embedded into a sample space of higher dimensionality. By using dimensionality-reduction techniques, a manifold is defined and the distance between the manifold and the new sample is calculated. One of the most common methods used for that purpose is Kernel Principal Component Analysis (K-PCA) [SSM97], which uses the kernel-trick to extend PCA and perform a nonlinear dimensionality reduction of the input. This technique has been successfully used for novelty detection in [Hof07].

[JMG95] also follows a similar approach: *Redundancy Compression and Non-Redundancy Differentiation* which is a process believed to happen in the hippocampus. They introduce an auto-encoder that learns to encode a sample on a considerable smaller description and later reconstructs the original sample from this smaller description. Their suggested system learns how to discard and compress redundant information while still be able to recover the original sample. By training it with samples from a given class, it is expected that it will badly reconstruct a sample from a different one.

[Ran10] presents a system able to perform place recognition and classification from visual clues. It is able to perform segmentation by exploiting time-coherency on video information. The approach is particularly interesting by its ability of keep a fully probabilistic distribution of the place classification and segmentation. In other words, the system never performs a deterministic decision that impacts any future result, allowing it to adjust the segmentation and place classification as more data becomes available. The system is also able to detect novel instances and methods how to adapt it to run on a constant amount of memory and computation are presented.

[BLB06] presents a method to perform scene-classification from low-level region detectors using probabilistic graphical models. The information obtained by the region detectors is exploited together with the spatial relations for the higher level scene-classification task. A scheme using only pairwise relations between regions is shown to have better performance than the correct approach of modelling connectivity of all regions with a large single factor as the pairwise relations can be approximated with modest amounts of training data, where the single factor demands an impractical amount of training data. No approach is made for novelty detection either of the regions or the scene categories.

4.2 Novelty Detection by Thresholding

The objective of a novelty detection system is to classify a given sample x as either *known*: x being generated by a class known to the system, or *novel*: x generated by a class unknown to the system. Based on the ground-truth of a sample four cases are possible:

true positive - when a system correctly flags a sample of an unknown class as *novel*.

false positive - when a system incorrectly flags a sample of a known class as *novel*.

true negative - when a system correctly flags a sample of a known class as *known*.

false negative - when a system incorrectly flags a sample of an unknown class as *known*.

Due to noisy data, unstable features and lack of information, it is impossible to develop a method able to always exactly guess the correct classification of a sample. By modelling the outcomes with probabilities, it becomes possible to handle the uncertainty associated with each decision. The notation $P(novel|x)$ will be used to denote the probability that the sample x is in fact a sample of an unknown class. $P(x)$ will denote the probability that sample x is given to the system to be classified. $\overline{novel}$ is also defined in such a way that $P(\overline{novel}|x)$ measures the probability that the sample x is generated by a known class.

Additionally a decision on the novelty of a sample x performed by a deterministic system will be fully determined by the sample itself. Which implies that any deterministic novelty detection system can be uniquely determined by the set N of samples that are accepted by the classifier as *novel*. This way it is possible to define the probability of a true positive and false positive event for any deterministic novelty detector with the following equations:

$$P(\text{true positive}) = \sum_{x \in N} P(novel|x)P(x) \tag{4.1}$$

$$P(\text{false positive}) = \sum_{x \in N} P(\overline{novel}|x)P(x) \tag{4.2}$$

Note that since probability functions are non-negative, it is impossible to decrease either the true positive or the false positive probability by using a set $N' \supset N$. This describes the base of the *error and rejection tradeoff* [Cho70], which states that a system aiming at increasing the true-positive probability will eventually increase its false-positive error. The true positive probability is related to the goal of detecting as many novel samples as possible. At the same time, the false positive probability is related to the goal of not making too many errors. By fixing one of those, it is possible to define a novelty detection system that achieves the maximum or minimum of the other.

This way an optimal detector can be formulated by achieving the maximum true-positive probability without its false-positive probability increasing beyond a given limit. This is equivalent to a *continuous knapsack problem* which allows a greedy solution by sorting the items with a value per weight function. In the case of detection system that can be defined as:

$$\begin{aligned}
value(x) &= P(\text{true positive}|x) && (4.3)\\
weight(x) &= P(\text{false positive}|x) && (4.4)\\
cost(x) &= value(x)/weight(x) && (4.5)\\
&= \frac{P(novel|x)P(x)}{P(\overline{novel}|x)P(x)} && (4.6)
\end{aligned}$$

Therefore a novelty detection system before classifying a sample a as novel should (greedily) classify any sample b with a smaller cost as that would achieve a higher true positive probability given a fixed false positive one.

$$\frac{P(novel|b)}{P(\overline{novel}|b)} < \frac{P(novel|a)}{P(\overline{novel}|a)} \tag{4.7}$$

This relation between a and b can further be simplified into:

$$P(\overline{novel}|b) < P(\overline{novel}|a) \tag{4.8}$$

Based on this, it can be said that an optimal novelty detection system is interested in defining an order relation on all the possible inputs equivalent to the order defined by the function: $P(\overline{novel}|x)$. And any optimal detector can be described by the largest $P(\overline{novel}|x)$ accepted by it. This can be seen as thresholding.

4.3 Conditional and Unconditional Probability Ratio

In the previous section, it was shown that an optimal novelty detector can be implemented by placing a threshold on the order relation defined by $P(\overline{novel}|x)$ over x. Performing some manipulations with Bayes theorem and assuming a constant $P(\overline{novel})$, a more usable form can be attained:

$$P(\overline{novel}|x) = \frac{P(x|\overline{novel})P(\overline{novel})}{P(x)} \propto \frac{P(x|\overline{novel})}{P(x)} \tag{4.9}$$

Since the interest is only in maintaining the order relation defined by $P(\overline{novel}|x)$, any constant factor can be dropped. Leaving a ratio between a *conditional* and *unconditional probability* suitable for implementing novelty detection by thresholding.

4.3.1 Conditional Probability

The conditional probability $P(x|\overline{novel})$ describes the distribution of the samples given that they are generated by a known class. In case when the labelled data used to learn a concept come from the same distribution from the test samples come, the correct approach is to use it as prior information for modelling the conditional probability.

Note that it is important for the labelled data to be a filtered version of the underlying world distribution (one modeling all the concepts in the world) that does not contain any bias. Otherwise that bias will lead to incorrect modelling of the conditional probability and in consequence wrong ordering of the sample space.

4.3.2 Unconditional Probability

The unconditional probability $P(x)$ plays an important role in obtaining a correct order relation for performing novelty detection. It serves as a normalizing component that allows the system to decide whether conditional probability of a given sample arises from it belonging to the known concept or from the likelihood of being sampled.

On lack of any information about the unconditional probability and conforming to the principle of maximum entropy (section 2.4) a uniform distribution must be chosen. However, by using unlabelled data, it becomes possible to obtain extra information and achieve a better approximation.

Note also that often novelty detection is applied to a fixed set of features together with an assumption of a uniform unconditional probability. In those cases, $P(x)$ becomes a constant and therefore a novelty threshold can be directly applied to $P(x|\overline{novel})$ as is the approach presented in [Bis94]. However, in case where the set of features x is variable, it cannot be discarded. There, $P(x)$ plays a role in levering all the conditional probabilities on different sets of variables into the same units so that a threshold can be obtained.

4.3.3 Assumption about Constant $P(novel)$

The ratio between the conditional and unconditional probabilities of the sensed variables is only directly applicable to novelty detection under the assumption of a constant $P(novel)$.

All the literature focuses on novelty detection within a fixed scenario, where the set of variables used to model the distribution is fixed and defined at the moment the threshold is trained. Due to that, it is possible to drop the constant $P(novel)$ and there is no need to calculate it directly or indirectly. To the best of the knowledge of the author there has been no studies on how to approximate it on dynamic sets and structures of sensed variables and a strong assumption has always been used about the factor being constant through all possible scenarios.

It is questionable whether this assumption holds in realistic scenarios. Note that in order to match the *criterium* of having a constant $P(novel)$, samples needs to be drawn with a method like:

1. Sample a graph structure G (where variable a can be any of the known or unknown classes)
2. Decide novelty of a.
3. Sample the remaining variables according to distribution $P_G(x|a)$.

Alternatively, it is expected that in a realistic scenario, samples are drawn with the following unbiased method:

1. Sample a graph structure G (where variable a can be any of the known or unknown classes)
2. Sample variables x according to distribution $P_G(x)$.

If in reality samples are drawn according to this unbiased method, assuming a constant $P(novel)$ will negatively impact the exactness of the detector. For that reason, the author believes this assumption is very strong and points out that future work should try to develop methods to include structure information about the probabilistic model to allow the system to deal with certain structures being more prone to produce novel samples.

4.4 Novelty Detection on the Conceptual Map

This section presents now how to use graphical models produced by the conceptual map and the novelty detection concepts introduced earlier in this chapter in order to detect room categories the system is not aware of.

For that purpose, two models approximating both the conditional and unconditional probability need to be defined. Then, the ratio between the resulting probabilities is used to define a function over which a threshold is specified.

4.4.1 Approximating the Conditional Probability

The semantic mapping algorithm uses the *conceptual map* introduced in section 3.4 to represent the environment on the high level of abstraction. During the semantic mapping process, the agent instantiates a *chain graph* to model the distribution of the sensed variables according to the known concepts and categories introduced as hidden variables in the graph. Using that graphical model, the system is able to propagate and find the most likely configuration of the represented variables. For instance, the semantic category of a specific room is modelled as a hidden variable with states representing the semantic values. By calculating probabilities of that variable, the system obtains the belief about the room belonging to a specific category.

Since the graphical model produced by the conceptual map models the distribution of the variables assuming the knowledge of the agent holds true, it corresponds to the model of $P(x|\overline{novel})$. It allows to calculate the probability density that the set of features x is sensed given that all the variables and graph structure, including the category of a specific room a are correctly modelled by the knowledge of the agent. This way, the distribution modelled by a factor graph equivalent to the chain graph used by the conceptual map can be used as an approximation for the conditional probability of x.

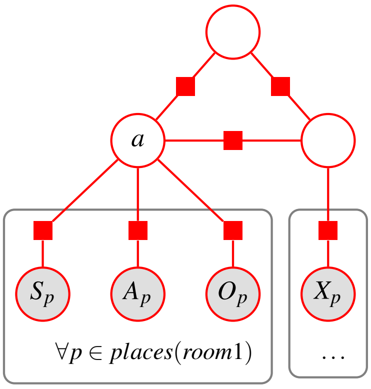


Figure 4.1: Factor graph modelling the conditional probability distribution, case where the room category of room a is known by the system.

4.4.2 Approximating the Unconditional Probability

With no knowledge about the unconditional probability, the correct approach is to assume a uniform distribution (section 2.4), which can be modelled by a factor graph without any factors: Figure 4.2.

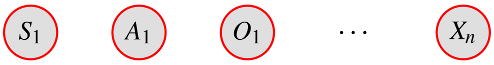


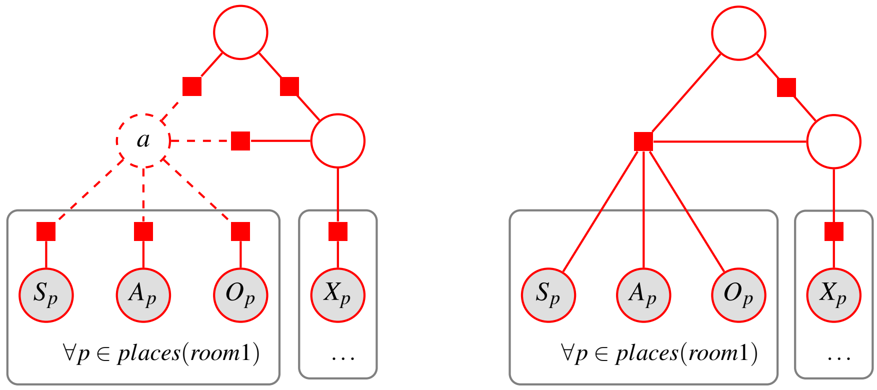
Figure 4.2: A factor graph modelling a uniform distribution over the sensed set of features x.

However, very often additional knowledge can be obtained that helps to model the unconditional distribution. Given that knowledge, more accurate models can be produced. In this case, the system only aims at detecting if a specific variable category is not known. With that in mind, it was assumed that the graph structure is known and that an unknown category can only influence variables using the same structure as the known variables. Therefore, the structural information and all the other hidden variables the system is aware of can be used to more accurately approximate the unconditional distribution.

If the knowledge of the agent is only lacking information about all the states of hidden variable a, it can still be used to approximate the sample distribution by avoiding the need to directly represent a. For that, all the variables that were directly dependent on a become directly dependent between each other introducing a single big factor connecting all of them (Figure 4.3). By approximating this factor, the agent can then obtain a model for the unconditional probability.

In cases when no other information is available, the introduced factor can be considered uniform[1] (see Figure 4.4a). Nonetheless, it may be the case that due to the cost of labelling data, the agent does not has knowledge on all the possible categories of a, but

[1] Uniform factors do not influence the distribution represented by the factor graph and so can be removed from the graph representation



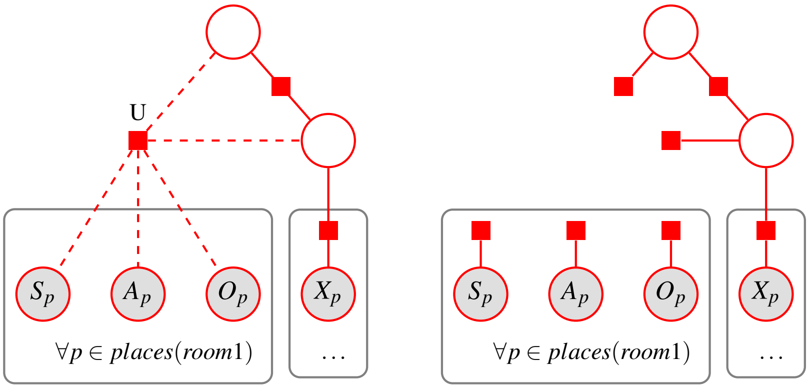
(a) Since variable a can be unknown, the agent has no knowledge about how to model the dashed relations.

(b) Without knowledge about all the states of a, all variables dependent on a are directly dependent on each other, introducing a big single factor.

Figure 4.3: Factor graph modelling distribution of sensed variables when variable a can be unknown.

still has access to unlabelled data. In that case unlabelled samples may be gathered and exploited to achieve a better approximation of the real distribution of variables. However, the factor that needs to be approximated requires handling of all the connected variables and enormous amounts of data may be needed to approximate it.

A simple approach can be used by assuming the variables connected to the factor are independent, but still have a bias towards certain values. This approach is equivalent to connecting single factors to each of variables and approximating those factors using unlabelled data as seen in Figure 4.4b.



(a) Uniform Model

(b) Independent Model

Figure 4.4: Factor graph modelling the conditional probability distribution, case where the room category of room a is known by the system.

Other approaches may be used to approximate the single factor described above and allow modeling the distribution when there is no knowledge about all the states of a specific variable. For example: by training a hidden variable in an unsupervised fashion or by using other factorization schemes for the factor. The presented cases have interesting properties:

uniform model - by assuming the factor to be uniform, it can be considered as not existent in the graphical model.

independent model - due to the use of single connected factors, it can easily be trained from unlabelled data. This helps the system to account for existing bias for each of the variables without the system overfitting to the unlabelled data.

4.4.3 Threshold

After defining a model for the conditional and unconditional distributions, a threshold can be applied to the ratio of those. Under the assumption of a constant $P(novel)$, defining a threshold over the ratio leads to an optimal novelty detector, as showed on section 4.2.

If the training data do not contain any bias, it is expected that by picking a threshold on the training data that achieves certain true-positive and false-positive rate, a system will achieve the same performance on the real distribution. However, the performance of the system is dependent on how well the models approximate the conditional and unconditional distributions and so are dependent on how the graph structure is able to model the real distribution and how well the training data allows the factors to be approximated.

4.5 A Practical Example

In order to summarize the presented concepts related to novelty detection with a threshold function and exemplify how to use graphical models in the context of multi-modality room classification, a synthetic dataset was generated. The dataset was kept simple by only modelling directly sensed features from a room, skipping any structural knowledge (room connectivity) and extra hidden variables.

In this dataset, a room r is seen as a hidden-variable generator of a set of features X that are directly sensed by the agent. All the sensed features x are independent given the room category. In total, there were 11 different room categories and 7 different feature types. Each feature can be sensed more than once (e.g. room shape is extracted from 2D laser scans in more than one position in the room), but all those sensed instances are considered independent given the room category.

The room categories were chosen to mimic as close as possible the real features and categories (i.e. 1 person office, 2 persons office, hallway, robot lab, etc.). A table describing synthetic distribution is given in Table 4.1.

The objective was to design a system that, although only trained with labelled data from 5 of the 11 room categories, was able to detect novel room categories. To this end, 100 labelled samples for the 5 known categories were drawn and 1000 unlabelled samples were drawn from all the room categories for learning the unconditional probability distribution and measure effect of using unlabelled data.

4.5.1 Conditional Probability

Using the labelled samples, 7 factors $\phi_X(r,x)$ were created, one for each feature type, to represent the potential of sensing features x of type X for the room of category r.

$$\phi_X(r,x) = \frac{\#(r,x)+C}{\sum_{i \in X} \#(r,i)+C} \tag{4.10}$$

Where $\#(r,x)$ denotes the number of times a feature x was sensed inside a room category r and C is a smoothing parameter that accounts for fixing the probabilities in case a given sample is never seen. With those, the probability of sensing a set of features x for a room category r known to the agent can be modelled with a factor graph illustrated in Figure 4.5.

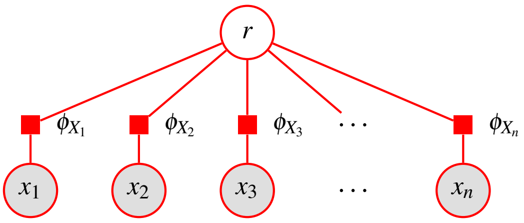


Figure 4.5: A factor graph modelling the conditional probability of sensing a set of features x given that the room category r is one of the known classes.

4.5.2 Unconditional Probability

With no knowledge about the unconditional probability, the correct approach is to assume a uniform distribution. That is represented as a factor graphs without any factors.

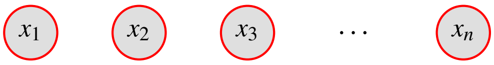


Figure 4.6: A factor graph modelling a uniform distribution over the sensed set of features x.

Very often there is extra knowledge that can be obtained about the distribution of the variables that helps to model the unconditional distribution. In this practical example, the access to unlabelled data is exploited. The availability of unlabeled data is common in practical robotic applications.

Note that the sensed variables are dependent on each other when the room category r is not known. The correct approach would be to train a factor that is able to correlate all the sensed variables as seen in Figure 4.7.

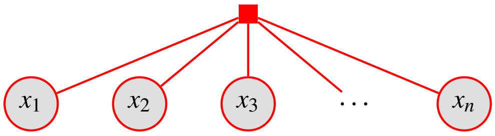


Figure 4.7: A general factor graph able to model any unconditional distribution of the sensed variables requires a factor connecting all of them.

Nonetheless such an approach suffers from the *curse of dimensionality*: as the number of sensed features and feature types increase, exponential amounts of data is needed to describe it. In order to avoid those the issues, an assumption about independent features can be made and then the unconditional probability can be modelled with fully disconnected variables, but with factors that account for existent bias on each single feature (as shown in the graph in Figure 4.8).

In case there is only one sensed feature, the distribution generated by the assumption of independent features correctly models the unconditional probability and it deviates as more features are sensed. The individual factors associated with each variable can be seen as a scaling of each feature which tries to account for a possible existent bias for each of them. Therefore, it is expected to be a better estimate than assuming a uniform distribution.

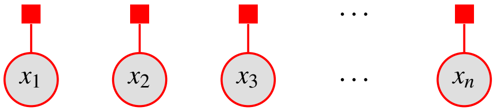


Figure 4.8: A factor graph modelling an independent distribution over the sensed set of features x.

4.5.3 Threshold Functions

Three threshold functions were created using the knowledge on the synthetic distribution and the models learnt from the sample data: G on Figure 4.5, U on Figure 4.6 and I on Figure 4.8.

exact - since the distribution is synthetic, there is access to $P(x)$ and $P(x|concept)$ and the ordering function $P(x|\overline{novel})/P(x)$ could be created to test how far the other presented thresholds are from the optimum.

uniform - by assuming a uniform unconditional distribution, the ordering function is given by $P_G(x)/P_U(x)$.

independent - using the unlabelled data, the unconditional distribution can be approximated. In this case, it was approximated with an independent distribution of the sensed features. The independent threshold was implemented with $P_G(x)/P_I(x)$.

4.5.4 Probability Ratio Comparison

First, the performance of the novelty threshold selection was plotted for a set of 1000 samples taken from the whole distribution (Figure 4.9). The samples where uniformly generated by graphs with 5, 10, 15, 20, 35, 50 features. Additionally the feature types were also uniformly sampled, for that it is possible that in certain samples some feature types were sensed more than once and other were not sensed at all. This was chosen to mimic the dynamic properties expected to see when implemented on a robot.

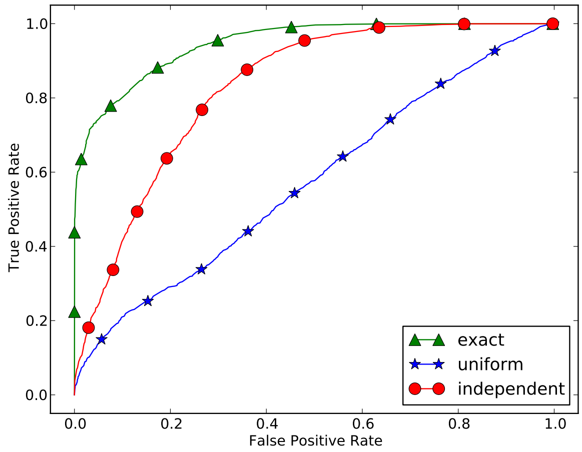


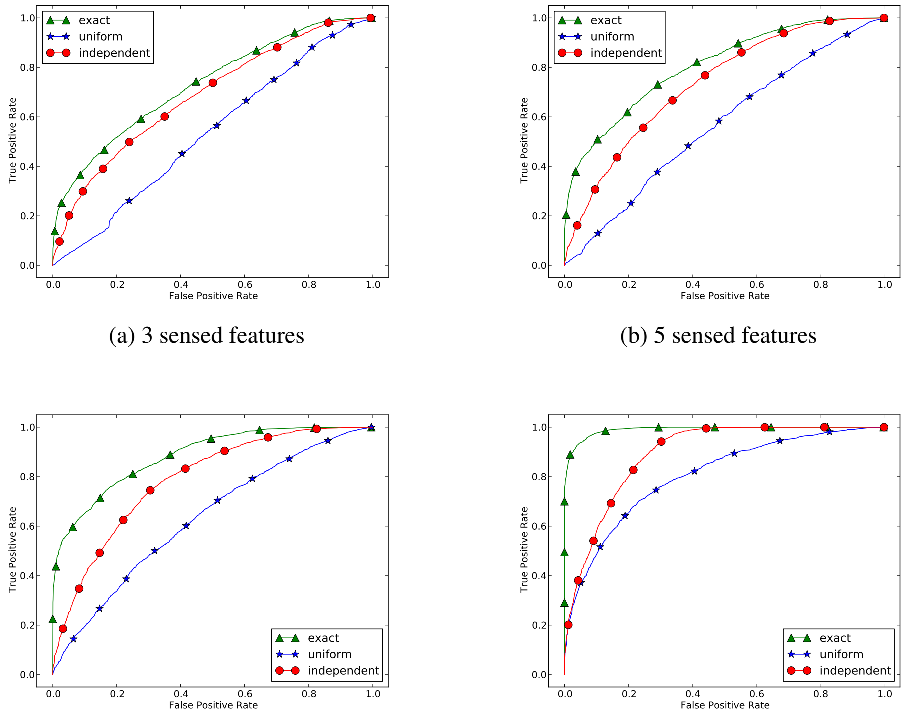
Figure 4.9: ROC curve comparing the novelty detection performance for samples with variable number of sensed properties.

The convex shape of the optimal threshold shows that the ratio between conditional and unconditional probability is indeed a suitable detector for implementing a threshold when the samples are taken from dynamic distributions when $P(novel)$ is constant (e.g. some samples where there is only access to room size versus samples where there is a lot of information about the room properties).

Its also possible to see how important it is to estimate a correct unconditional probability in order to obtain a correct novelty measure on the inputs. The assumption of a uniform unconditional probability has led to very poor results. That is probably explained by the semantic properties being highly biased towards some values. This shows that bias plays an important role in detecting whether a given sensed value is a valuable cue about the room category.

4.5.5 Influence of the Amount of Available Information

In order to measure the performance impact as more semantic information becomes available, ROC curves were plotted for samples grouped by the number of sensed semantic features.



(a) 3 sensed features

(b) 5 sensed features

(c) 10 sensed features

(d) 50 sensed features

Figure 4.10: ROC curves plotted showing performance of the presented novelty detection method for graphs generated for different amount of sensed features.

It is possible to see that as the system gains more semantic information, it becomes easier to detect novelty. The size of the input space increases and allows the existing classes to become more easily distinguished.

The performance of the independent threshold decreases as the number of sensed features increases. This is easily explained by the fact that the graph I is not able to model the existent dependence between the features. This becomes obvious as the number of features increases (e.g. graph I perfectly models $P(x)$ in the case where only 1 feature is sensed).

The uniform threshold shows poor performance especially for samples with small amount of features where it performs almost no better than random. The performance increases as the size of sensed features increases but nonetheless is very small when compared to the optimal threshold.

	appearance property							object book				object cerealbox				object computer				object robot				object stapler				object toiletpaper				shape property			size property		
	anteroom	bathroom	hallway	kitchen	lab	meetingroom	office	0	1	2	3+	0	1	2	3+	0	1	2	3+	0	1	2	3+	0	1	2	3+	0	1	2	3+	elongated	rectangular	square	large	medium	small
anteroom	88.0%	2.0%	2.0%	2.0%	2.0%	2.0%	2.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	10.0%	30.0%	60.0%	10.0%	30.0%	60.0%
bathroom	2.0%	88.0%	2.0%	2.0%	2.0%	2.0%	2.0%	87.5%	11.7%	0.8%	0.0%	90.0%	9.5%	0.5%	0.0%	83.0%	15.5%	1.4%	0.1%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	32.2%	36.5%	20.7%	10.7%	10.0%	30.0%	60.0%	10.0%	30.0%	60.0%
computerlab	2.0%	2.0%	2.0%	2.0%	88.0%	2.0%	2.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	71.0%	24.3%	4.2%	0.5%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	10.0%	30.0%	60.0%	60.0%	30.0%	10.0%
conferencehall	2.0%	2.0%	2.0%	2.0%	2.0%	88.0%	2.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	10.0%	30.0%	60.0%	60.0%	30.0%	10.0%
doubleoffice	2.0%	2.0%	2.0%	2.0%	2.0%	2.0%	88.0%	57.2%	32.0%	8.9%	1.9%	90.0%	9.5%	0.5%	0.0%	12.1%	25.6%	27.0%	35.4%	90.0%	9.5%	0.5%	0.0%	19.6%	32.0%	26.0%	22.4%	90.0%	9.5%	0.5%	0.0%	10.0%	30.0%	60.0%	20.0%	60.0%	20.0%
hallway	2.0%	2.0%	88.0%	2.0%	2.0%	2.0%	2.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	60.0%	30.0%	10.0%	60.0%	30.0%	10.0%
kitchen	2.0%	2.0%	2.0%	88.0%	2.0%	2.0%	2.0%	84.9%	13.9%	1.1%	0.1%	66.4%	27.2%	5.6%	0.8%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	78.1%	19.3%	2.4%	0.2%	61.8%	29.7%	7.2%	1.3%	10.0%	30.0%	60.0%	60.0%	30.0%	10.0%
meetingroom	2.0%	2.0%	2.0%	2.0%	2.0%	88.0%	2.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	10.0%	60.0%	30.0%	20.0%	60.0%	20.0%
professorsoffice	2.0%	2.0%	2.0%	2.0%	2.0%	2.0%	88.0%	57.2%	32.0%	8.9%	1.9%	90.0%	9.5%	0.5%	0.0%	42.1%	36.4%	15.8%	5.7%	90.0%	9.5%	0.5%	0.0%	19.6%	32.0%	26.0%	22.4%	90.0%	9.5%	0.5%	0.0%	10.0%	30.0%	60.0%	20.0%	60.0%	20.0%
robotlab	2.0%	2.0%	2.0%	2.0%	88.0%	2.0%	2.0%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	71.0%	24.3%	4.2%	0.5%	30.0%	36.1%	21.7%	12.1%	90.0%	9.5%	0.5%	0.0%	90.0%	9.5%	0.5%	0.0%	10.0%	30.0%	60.0%	60.0%	30.0%	10.0%
singleoffice	2.0%	2.0%	2.0%	2.0%	2.0%	2.0%	88.0%	57.2%	32.0%	8.9%	1.9%	90.0%	9.5%	0.5%	0.0%	42.1%	36.4%	15.8%	5.7%	90.0%	9.5%	0.5%	0.0%	19.6%	32.0%	26.0%	22.4%	90.0%	9.5%	0.5%	0.0%	10.0%	60.0%	30.0%	20.0%	60.0%	20.0%

Table 4.1: Distribution used for the synthetic data experiment. Each column cell shows $P(feature|class)$

Chapter 5

Conclusions and Future Work

This thesis studied the problem of detecting novel situations in case of which the robot lacks sufficient knowledge to explain the sensory information. It did so in the area of semantic mapping on indoor spaces, and more precisely, for the problem of detecting the situation when none of the known room category models were able to correctly explain the sensed spatial properties.

It reviewed novelty detection and how an optimal detector can be implemented by thresholding. It also showed that with the assumption of constant probability of seeing a novel case, an optimal ordering function for thresholding can be implemented based on the factor between conditional and unconditional probability.

It studied the semantic mapping process proposed in [PJ11] and presented a method to detect novel room categories based on probabilistic graphical models. Using a synthetic dataset, respecting the assumptions, it showed that such a method would be optimal if unconditional probability could be optimally approximated. Since, in realistic conditions, unconditional probability cannot be obtained due to the lack of knowledge about all possible classes, various techniques were used to approximate it with either a uniform assumption or simplified models built from unlabelled data.

The rest of this chapter presents the main results and conclusions. Additionally limitations of the presented methods and directions for future work are discussed.

5.1 Results and Conclusions

After studying the semantic mapping process and novelty detection methods, this thesis proposed modelling the conditional and unconditional probability distribution of sensed data using graphical models. Under the presented assumptions, and assuming both distributions can be exactly known, a ratio between those probabilities would yield an optimal ordering function for implementing a novelty detector. This case was showed to be optimal using a synthetic dataset that simulates a simplified environment for semantic categorization of rooms based on sensed properties. As the next step, this thesis tested methods to approximate the unconditional probability. It tested the usage of a uniform distribution and explored the usage of unlabelled data to produce better models.

Additionally it studied how the performance of the presented methods changed as more data was sensed from the environment. As expected, all the methods increase their performance, but they move further away from what could be obtained with exact information. Additionally, the disadvantage of the uniform assumption compared to using a highly simplified model for the unlabelled data starts reducing as more information is available. Nonetheless the model based on the usage of unlabelled data consistently lead to better detector performance, which is a strong indication that unlabelled data should be used whenever it is available.

As a note, all the results presented in this thesis are directly reproducible from an online[1] repository. That repository besides containing all the code and data for the results also includes tech-reports, presentations, articles and notes that have been produced during this thesis work. Additionally, future research by the author will be correctly linked there, when appropriate.

5.2 Limitations

The presented methods use a strong assumption on a constant $P(novel)$, this is unrealistic and forbids one to exploit graph structural information that could show a given variable is more likely to be unknown to the system because the given graph structure is not easily explained by the current knowledge. As as example of that, consider the following: if all the agent knows are two categories that are very often seen as a bi-colored graph, when presented with a non-bipartite graph, the agent should consider the likelihood of a new category greater than on bipartite graphs.

Additionally, all the presented methods require calculation of $P(x|\overline{novel})$ and $P(x)$. This way a full graphical model is needed and uncertain sensing as described in subsection 3.4.2 cannot be easily incorporated.

These two limitations are a single symptom of the decision on inverting the conditional probability $P(novel|x)$ to obtain a threshold that can be directly modelled by the data. It is expected that by trying to model how the potentials in the factor graphs change between the conditional and unconditional graphical model, both limitations can be surpassed.

5.3 Future Work

Future work will try to study and work around the limitations presented above. At the moment it is not clear how the threshold changes when the assumption about constant

[1] `https://github.com/andresusanopinto/novelty-detection-thesis`

$P(novel)$ cannot be made. Any attempts to bypass the listed limitations will need to consider that problem by developing a method where a threshold can have a more realistic and controlled behaviour for any graph structure.

The following paragraphs describe possible and interesting directions to explore in the context of detecting knowledge gaps in artificial intelligent systems:

Generalized the Framework

The presented method should be generalized by allowing novelty detection to be performed on any variable of the graphical model. Additionally, the novelty information could be incorporated back into the graph allowing the agent to probabilistically reason even when variables are considered unknown. That generalization should aim at being fully probabilistic such as the system presented by [Ran10] and not deterministic by having to make decisions on which variables it considers novel.

Additionally, several methods exists that allow to produce novelty signals from the low-level classifiers. A generalized framework should try to fuse and handle all that information by incorporating it into the graphical model in a similar fashion as how uncertain sensing outcomes are incorporated.

Exploiting Generative Models

An interesting aspect that arises from the use of generative models is the ability to generate new samples from the models. It is expected that this can be exploited to achieve better understanding of what the system is modelling and what explain the character of the novelty back to the user. Moreover, this feature can be used to refine and actively update the models by confronting the captured knowledge with the human understanding of the modeled concepts.

Beyond Novel Semantic Categories: Learning Graph Structures

Although this thesis only touched the problem of detecting novel semantic categories, other knowledge types should also be considered incomplete and are also candidates for novelty detection. More concretely, besides detecting novel semantic categories for given spatial concepts, there is also interest in detecting novel concepts: the presence of hidden variables of a new type, not known to the system: for example detecting rooms or types of environment.

An initial step towards that goal could be probabilistic learning of graph structures to model various possible space segmentations. This would permit the use of a probabilistic approach instead of the currently used deterministic approach where space is segmented based on the presence of doors or other landmarks.

Beyond Detection of Knowledge Gaps

Having methods to detect gaps of knowledge is just one of the first steps towards creating long-term and life-long adaptable systems that are capable of learning over time. After detecting new situations, ways to learn and incorporate the knowledge into the models must be developed.

References

[BC00] K.P. Bennett and C. Campbell. Support vector machines: hype or hallelujah? *ACM SIGKDD Explorations Newsletter*, 2(2):1–13, 2000.

[Bis94] CM Bishop. Novelty detection and neural network validation. *IEE Proc.-Vls. Image Signal Process*, 141(4):217, 1994.

[Bis06] C.M. Bishop. *Pattern recognition and machine learning*, volume 4. Springer New York, 2006.

[BK02] C. Borgelt and R. Kruse. *Graphical models: methods for data analysis and mining*. Wiley, 2002.

[BLB06] M.R. Boutell, J. Luo, and C.M. Brown. Factor Graphs for Region-based Whole-scene Classification. In *Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, page 104. IEEE Computer Society, 2006.

[BOV03] A. Barla, F. Odone, and A. Verri. Histogram intersection kernel for image classification. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 3. IEEE, 2003.

[BTB05] S. Boughorbel, J.P. Tarel, and N. Boujemaa. The intermediate matching kernel for image local features. In *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, volume 2, pages 889–894. IEEE, 2005.

[Cho70] C. Chow. On optimum recognition error and reject tradeoff. *Information Theory, IEEE Transactions on*, 16(1):41–46, 1970.

[CV95] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[DMO08] P.R.R. Devarakota, B. Mirbach, and B. Ottersten. Reliability estimation of a statistical classifier. *Pattern Recognition Letters*, 29(3):243–253, 2008.

[FJC07] J. Folkesson, P. Jensfelt, and H. I. Christensen. The m-space feature representation for SLAM. *IEEE Trans. Robotics*, 23(5):1024–1035, October 2007.

[FMS$^+$01] W. Fan, M. Miller, S.J. Stolfo, W. Lee, and P.K. Chan. Using artificial anomalies to detect unknown and known network intrusions. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 123–130. IEEE, 2001.

[HGD$^+$11] Marc Hanheide, Charles Gretton, Richard W Dearden, Nick A Hawes, Jeremy L Wyatt, Andrzej Pronobis, Alper Aydemir, Moritz Göbelbecker, and Hendrik Zender. Exploiting probabilistic knowledge under uncertain sensing for efficient robot behaviour. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11)*, Barcelona, Spain, July 2011.

[Hof07] Heiko Hoffmann. Kernel pca for novelty detection. *Pattern Recognition*, 40(3):863 – 874, 2007.

[JMG95] N. Japkowicz, C. Myers, and M. Gluck. A novelty detection approach to classification. In *International Joint Conference on Artificial Intelligence*, volume 14, pages 518–523. LAWRENCE ERLBAUM ASSOCIATES LTD, 1995.

[KFL01] F.R. Kschischang, B.J. Frey, and H.A. Loeliger. Factor graphs and the sum-product algorithm. *Information Theory, IEEE Transactions on*, 47(2):498–519, 2001.

[LL04] O. Linde and T. Lindeberg. Object recognition using composed receptive field histograms of higher dimensionality. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, pages 1–6. IEEE, 2004.

[Low99] D.G. Lowe. Object recognition from local scale-invariant features. In *iccv*, page 1150. Published by the IEEE Computer Society, 1999.

[LR02] S.L. Lauritzen and T.S. Richardson. Chain graph models and their causal interpretations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(3):321–348, 2002.

[Moo10] Joris M. Mooij. libDAI: A free and open source C++ library for discrete approximate inference in graphical models. *Journal of Machine Learning Research*, 11:2169–2173, August 2010.

[MS03] M. Markou and S. Singh. Novelty detection: a review–part 1: statistical approaches. *Signal Processing*, 83(12):2481–2497, 2003.

[MTEF06] K. Murphy, A. Torralba, D. Eaton, and W. Freeman. Object detection and localization using local and global features. *Toward Category-Level Object Recognition*, pages 382–400, 2006.

[OT06] A. Oliva and A. Torralba. Building the gist of a scene: The role of global image features in recognition. *Progress in brain research*, 155:23–36, 2006.

[PC07] Andrzej Pronobis and Barbara Caputo. Confidence-based cue integration for visual place recognition. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, pages 2394–2401, San Diego, CA, USA, October 2007.

[Pea88] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.

[PJ11] Andrzej Pronobis and Patric Jensfelt. Understanding the real world: Combining objects, appearance, geometry and topology for semantic mapping. Technical Report TRITA-CSC-CV 2011:1 CVAP319, Kungliga Tekniska Högskolan, CVAP/CAS, May 2011.

[PMCJ10] Andrzej Pronobis, Oscar M. Mozos, Barbara Caputo, and Patric Jensfelt. Multi-modal semantic place classification. *The International Journal of Robotics Research (IJRR), Special Issue on Robotic Vision*, 29(2-3):298–320, February 2010.

[Pro11] Andrzej Pronobis. *Semantic Mapping with Mobile Robots*. PhD thesis, Royal Institute of Technology (KTH), Stockholm, Sweden, June 2011.

[PSA$^+$10] Andrzej Pronobis, Kristoffer Sjöö, Alper Aydemir, Adrian N. Bishop, and Patric Jensfelt. Representing spatial knowledge in mobile cognitive systems. In *11th International Conference on Intelligent Autonomous Systems (IAS-11)*, Ottawa, Canada, August 2010.

[QT09] A. Quattoni and A. Torralba. Recognizing indoor scenes, 2009.

[Ran10] A. Ranganathan. Pliss: Detecting and labeling places using online change-point detection. In *Proceedings of Robotics: Science and Systems*, Zaragoza, Spain, June 2010.

[SC96] B. Schiele and J. Crowley. Object recognition using multidimensional receptive field histograms. *Computer Vision—ECCV'96*, pages 610–619, 1996.

[SJ80] J. Shore and RW Johnson. Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy. *Information Theory, IEEE Transactions on*, 26(1):26–37, 1980.

[SSM97] B. Schölkopf, A. Smola, and K.R. Müller. Kernel principal component analysis. *Artificial Neural Networks—ICANN'97*, pages 583–588, 1997.

[SWS$^+$00] B. Schölkopf, R.C. Williamson, A.J. Smola, J. Shawe-Taylor, and J. Platt. Support vector method for novelty detection. *Advances in neural information processing systems*, 12(3):582–588, 2000.

[TD98] D. Tax and R. Duin. Outlier detection using classifier instability. *Advances in Pattern Recognition*, pages 593–601, 1998.

[THCB95] L. Tarassenko, P. Hayton, N. Cerneaz, and M. Brady. Novelty detection for the identification of masses in mammograms. In *Artificial Neural Networks, 1995., Fourth International Conference on*, pages 442–447. IET, 1995.

[TNTC99] L. Tarassenko, A. Nairac, N. Townsend, and P. Cowley. Novelty detection in jet engines. In *Condition Monitoring: Machinery, External Structures and Health (Ref. No. 1999/034), IEE Colloquium on*, pages 4–1. IET, 1999.

[ZCM02] Y. Zhang, J. Callan, and T. Minka. Novelty and redundancy detection in adaptive filtering. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 81–88. ACM, 2002.