

Towards Generalization of Semi-supervised Place Classification over Generalized Voronoi Graph

Lei Shi, Sarath Kodagoda

Centre for Autonomous Systems (CAS)

The University of Technology, Sydney, Australia

Abstract

With the progress of human-robot interaction (HRI), the ability of a robot to perform high-level tasks in complex environments is fast becoming an essential requirement. To this end, it is desirable for a robot to understand the environment at both geometric and semantic levels. Therefore in recent years, research towards place classification has been gaining in popularity. After the era of heuristic and rule-based approaches, supervised learning algorithms have been extensively used for this purpose, showing satisfactory performance levels. However, most of those approaches have only been trained and tested in the same environments and thus impede a generalized solution. In this paper, we have proposed a semi-supervised place classification over a generalized voronoi graph (SPCoGVG) which is a semi-supervised learning framework comprised of three techniques: support vector machine (SVM), conditional random field (CRF) and generalized voronoi graph (GVG), in order to improve the generalizability. The inherent problem of training CRF with partially observed data has been solved using a novel parameter estimation algorithm. The effectiveness of the proposed algorithm is validated through extensive analysis of data collected at international university environments.

Keywords: place classification, conditional random field, semi-supervised learning, generalized voronoi graph

1. Introduction

Human-robot co-existence is full of challenges as a reasonable mutual understanding may be required. As humans, we are more focused on using semantic labels rather than precise coordinates in our day-to-day language. Therefore, it may be argued that the robots that understand semantic labels of place are better equipped to effectively interact with humans. It allows communication in a human-friendly way. Further, semantic place information has the potential to facilitate other functions such as mapping [1, 2], behavior-based navigation [3], task planning [4] and active object search and rescue [5, 6] in an efficient way. Therefore,

research on place classification has been an important step in the quest for intelligent human robot interactions.

Place classification, also referred to as *scene categorization* or *semantic labeling of places* in some literature, denotes the problem of distinguishing differences between environmental locations, and assigning a label (kitchen, office, corridor, etc) to each location [7, 8]. This is compared with *place recognition*, which refers to the ability to recognize previously seen parts of the environment [9-11]. A commonly held outlook is that *place classification* is a more challenging problem due to the presence of higher intra-class variations which warrants the formation of a conceptual model of the place [12-14].

Place classification has a variety of applications in robotics. At present, semantic mapping is the most fundamental application as the semantic information can be directly added to conventional maps such as metric maps and occupancy grids maps [1, 2]. Other researchers point out that topological maps are more natural forms of semantic mapping results, as a node in the topological map is usually a semantic place unit like a room on a floor plan [7]. We believe that a more forward-looking approach is to develop a hierarchical mapping scheme. To this end, Vasudevan and Siegwart proposed cognitive maps on top of metric and topological maps [13]. The progenitors of the multi-layered conceptual mapping suggested a hierarchical structure that integrates the metric map, navigation map, topological map and conceptual map; and the map building process is to be regarded as a human-like decomposition and categorization of space [15-17]. Similarly, Pronobis and Jensfelt introduced a probabilistic framework combining heterogeneous information, uncertainty and human input for semantic mapping [18].

At present, supervised learning is the most widely adopted approach for place classification. In an earlier work, using principal component analysis (PCA) and distance based matching, Poncela et al. [3] employed the spectral features from sonar data to classify the observations into wall, corridor and door, and the resulting accuracies vary in different environments. Tapus et al. [19] used Bayesian programming to discriminate various corridor shapes and door states achieving above 82% of successful classifications. With an Adaboost classifier and twenty two single valued features from 360° laser range data, Mozos et al. [20] classified the environments into four categories, with accuracies of approximately 92% by training and testing on the same map and approximately 82% on different maps. In a similar setup, we

implemented both binary and multi-class logistic regression based solutions, and have been able to achieve accuracies above 98% [21, 22]. In recent years, SVM as a prominent classifier, has been gaining popularity over other approaches in many applications [23, 24]. Using the multi-class SVM on features extracted from real 3D data, Swadzba and Wachsmuth [23] achieved approximately 80% accuracy, while the work of Mozos et al. [24] showed accuracies above 92% and we have obtained approximately 97% [25]. It is to be noted that the accuracies mentioned here are achieved by different methods, on different data sets and may address completely different tasks. Therefore, a direct comparison of accuracies may not be meaningful.

In general terms, the current place classification approaches face a variety of challenges including changes in appearances and illumination conditions [9, 11], in-class variations due to discrepancies in shape, color and texture across individual samples [23], the influence of choosing training and test data on the overall performance of the system [9], and the interference of dynamic environments and human activities [9, 11, 26]. Many of these problems can be summarized as the reflections of the generalization ability of the system, which may be improved by incorporating contextual information and adopting semi-supervised learning algorithms.

It is generally observed in the literature that contextual information usually leads to better classification accuracies [2, 27, 28]. Typical forms of this information include temporal consistency [7, 29], spatial consistency [1, 18, 20, 30, 31] and place-object relationships [29, 32]. Besides the methods that detect change-point directly [31], many systems adopt probabilistic graphical models like the hidden Markov model (HMM) [20, 30] or the conditional random field (CRF) [1] to incorporate the spatial dependencies between places. For example, a trajectory naturally has a chain-like structure, and other representations of the space, like a navigation graph, can be converted to tree-like or more general structures [17, 28]. However, these algorithms still suffer from poor generalization ability, especially when trained with data from one environment and tested on another.

As widely adopted fully supervised learning has long been criticized for the offline learning process, there has recently been a trend to move towards semi-supervised learning which is believed to be more practical and flexible than the former. For this purpose, a few variations of the original SVM algorithm have been proposed. For example, algorithms like online

independent-SVM (OISVM) and memory-controlled incremental SVM do not require storing all incoming data, and have selection mechanisms to guarantee a bounded memory growth [11, 26, 33]. These approaches focus more on the algorithmic efficiency and can be further improved by considering the spatial context. In this regard, the coSVM algorithm provides a semi-supervised variant of the standard SVM for structured output variables [34]. CRF based semi-supervised solutions have been reported in image processing, sequence segmentation and sequence labeling applications [35, 36]. Attempts to combine both SVM and CRF have also been reported in various applications [37, 38], and the more general kernel methods are regarded as the emerging theme [39]. This has motivated our research towards developing the SPCoGVG framework for the specific application, which exploits the merits of both the generalization properties of SVM and the spatial class dependencies provided by CRF [38]. The work by Brefeld and Scheffer [34] can be regarded as the most closely related work. It considers a co-training framework for semi-supervised learning in structured prediction models focusing on structured SVM. Our proposed method is focused on CRF and the main difference of our work is the incorporation of partial labeling. Although partial labeling in CRF has been examined before [35], to our best of knowledge partially labeled data has not previously been used within the co-training framework. It is also to be noted that the proposed work provides an alternative but mathematically equivalent computational strategy to the approach reported in [35].

The rest of this paper is arranged as follows. Section 2 discusses the underlying learning methods for environment modeling, and the work is extended in Section 3 to introduce the two semi-supervised learning techniques for SPCoGVG including the co-training framework and CRF having partial labeled data (CRF_{PL}). Experimental setup including data sets and the procedure is described in Section 4. Corresponding experimental results are shown in Section 5 with analysis and discussion. Section 6 concludes the paper.

2. Supervised Learning

SVM and CRF are commonly used in fully supervised learning settings. In general, the SVM solution assumes the instances to be independent, whereas the CRF solution incorporates the spatial relationships.

2.1. Support Vector Machine

SVM is a prominent learning algorithm based on a theoretical foundation rooted in statistical learning theory [40]. The basic idea of SVM is to map data into a high dimensional feature

space and find an optimal separating hyper-plane with the maximal margin.

Consider a set of instance-label pairs $\{\mathbf{x}_i, y_i\}_{i=1}^m$, $\mathbf{x}_i \in R^n$ and $y_i \in \{1, -1\}$, where \mathbf{x}_i and y_i are the feature vector and the label respectively of the i^{th} instance, and m is the number of instances. In the training process, instances \mathbf{x} are mapped into a high dimensional feature space F via a nonlinear mapping $\phi: R^n \rightarrow F$, and then the SVM constructs an optimal separating hyper-plane $\mathbf{w}^T \phi(\mathbf{x}) + b = 0$ with a maximum-margin and bounded error by solving the optimization problem:

$$\min_{\mathbf{w}, b} \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^m \xi_i \quad (1)$$

$$s.t. \quad y_i (\mathbf{w}^T \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \dots, m \quad (2)$$

where \mathbf{w} and b denote the weight vector and the bias in the optimal hyper-plane equation respectively. The positive constant C is a penalty parameter used to control the amount of regularization, and ξ_i is a non-negative slack variable accounting for the amount of misclassification.

For implementation, a kernel trick is introduced so that it is not required to find out the specific definition of ϕ . The kernel function is defined as:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \quad (3)$$

and some basic kernel functions are [41]:

$$\text{Linear: } K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$$

$$\text{Polynomial: } K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i^T \mathbf{x}_j + r)^d, \gamma > 0$$

$$\text{Radial Basis Function: } K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2), \gamma > 0$$

$$\text{Sigmoid: } K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma \mathbf{x}_i^T \mathbf{x}_j + r)$$

where γ, r , and d are kernel parameters.

When using a non-linear kernel, the inner-product in equation (1) needs to be computed in the Hilbert-space corresponding to the mapping.

Given that the SVM is inherently a binary classifier, there exist various multi-class solutions. In this paper, we utilize the multi-class implementation of the C-support vector classification scheme included in the LIBSVM package [42].

2.2. Conditional Random Field

2.2.1. Pairwise CRF

Probabilistic graphical models capture both the uncertainty and logical structure to compactly represent complex real-world phenomena and to effectively learn and perform inference in large networks [43]. Unlike generative models such as the hidden markov model (HMM) or the markov random field (MRF), CRF is a discriminative model estimating the conditional probability distribution $p(\mathbf{y}|\mathbf{x})$ directly, where \mathbf{y} and \mathbf{x} represent labels and instances (feature vectors) respectively.

Specifically, an implementation of a CRF with pairwise potentials by Schmidt et al. [44] was employed in the work presented in this paper as it supports our framework well. The conditional distribution of pairwise CRF is defined as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{\langle ij \rangle} \psi_{ij}(y_i, y_j, \mathbf{x}) \prod_i \psi_i(y_i, \mathbf{x}) \quad (4)$$

where ψ_i and ψ_{ij} are node and edge potentials respectively, and $Z(\mathbf{x})$ is the normalizing partition function. The node potential is a function of node features \mathbf{x}_i and parameter matrix \mathbf{w} , and the edge potential is a function of edge features \mathbf{x}_{ij} and parameter matrix \mathbf{v} . As an option of the algorithm proposed in [44], we choose $\mathbf{x}_i = [1, \mathbf{f}_i]$ and $\mathbf{x}_{ij} = [1, \mathbf{f}_i, \mathbf{f}_j]$ as the forms of the node and edge feature sets respectively, where \mathbf{f}_i is the local feature vector associated with node i . To be specific, the edge feature set is comprised of the node features from both end nodes of an edge.

In order to reduce the risk of over parameterization and improve the generalization ability of the model, we apply the same set of parameter matrices on all nodes and edges, and set the node and edge potentials in the following forms:

$$\psi_i(\cdot, \mathbf{x}) = \left(e^{\mathbf{w}_1^T \mathbf{x}_i}, e^{\mathbf{w}_2^T \mathbf{x}_i}, \dots, e^{\mathbf{w}_{n-1}^T \mathbf{x}_i}, 1 \right) \quad (5)$$

$$\psi_{ij}(\cdot, \cdot, \mathbf{x}) = \begin{pmatrix} e^{v_{11}^T \mathbf{x}_{ij}} & 1 & \dots & 1 & 1 \\ 1 & e^{v_{22}^T \mathbf{x}_{ij}} & \dots & 1 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \dots & e^{v_{n-1,n-1}^T \mathbf{x}_{ij}} & 1 \\ 1 & 1 & \dots & 1 & 1 \end{pmatrix} \quad (6)$$

where n is the number of classes, $\mathbf{w} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{n-1}]$, and $\mathbf{v} = [\mathbf{v}_{11}, \mathbf{v}_{22}, \dots, \mathbf{v}_{n-1,n-1}]$.

For the convenience of further analysis, equation (4) can be written in another way:

$$p(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta}) = \exp(\langle \phi(\mathbf{x}, \mathbf{y}), \boldsymbol{\theta} \rangle - z(\boldsymbol{\theta} | \mathbf{x})) \quad (7)$$

$$\text{where } z(\boldsymbol{\theta} | \mathbf{x}) = \ln \sum_{\mathbf{y}} \exp(\langle \phi(\mathbf{x}, \mathbf{y}), \boldsymbol{\theta} \rangle)$$

The parameter $\boldsymbol{\theta} = [\mathbf{w}, \mathbf{v}]$ is the concatenation of \mathbf{w} and \mathbf{v} , $\langle \cdot, \cdot \rangle$ denotes the inner product, and $\phi(\mathbf{x}, \mathbf{y})$ is called sufficient statistics:

$$\phi(\mathbf{x}, \mathbf{y}) = \left(\sum_{\langle ij \rangle} \psi_{ij}(y_i, y_j, \mathbf{x}), \sum_i \psi_i(y_i, \mathbf{x}) \right) \quad (8)$$

By applying the clique decomposition, $\phi(\mathbf{x}, \mathbf{y})$ can be calculated by summing the clique potentials over all nodes and edges [45]. It also requires that the parameters be shared across nodes and edges.

2.2.2. Parameter Estimation and Inference

Given a training set with m instances, $\{X, Y\} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^m$, the distribution $P(Y | X; \boldsymbol{\theta})$ can be written as:

$$P(Y | X; \boldsymbol{\theta}) = \prod_{i=1}^m p(\mathbf{y}_i | \mathbf{x}_i; \boldsymbol{\theta}) \quad (9)$$

To estimate the parameter $\boldsymbol{\theta}$, the maximum conditional likelihood estimation is adopted, i.e.

$\boldsymbol{\theta}^* = \arg \max_{\boldsymbol{\theta}} P(Y | X; \boldsymbol{\theta})$, which is equivalent to minimizing the negative log-likelihood

$\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta}} nll(\boldsymbol{\theta})$. As per the common practice, a $\lambda \|\boldsymbol{\theta}\|^2$ term can be added to the

conditional likelihood function to improve cross-validation results, known as L2-

regularization. After parameter estimation, the prediction process consists of calculating

$\mathbf{y}^* = \arg \max_{\mathbf{y}} p(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta}^*)$ for any new observation \mathbf{x} .

By putting equation (7) and (9) together, the negative log-likelihood function can be written as:

$$nll(\boldsymbol{\theta}) = -\sum_{i=1}^m \left[\langle \phi(\mathbf{x}_i, \mathbf{y}_i), \boldsymbol{\theta} \rangle - z(\boldsymbol{\theta} | \mathbf{x}_i) \right] \quad (10)$$

and the gradient of the negative log-likelihood is also required:

$$\nabla nll(\boldsymbol{\theta}) = -\sum_{i=1}^m \left[\phi(\mathbf{x}_i, \mathbf{y}_i) - E_{p(\mathbf{y}|\mathbf{x}_i;\boldsymbol{\theta})} [\phi(\mathbf{x}_i, \mathbf{y})] \right] \quad (11)$$

$$\text{where } E_{p(\mathbf{y}|\mathbf{x};\boldsymbol{\theta})} [\phi(\mathbf{x}, \mathbf{y})] = \sum_{\mathbf{y}} p(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta}) \phi(\mathbf{x}, \mathbf{y})$$

Various methods are used to perform the parameter estimation and inference for CRF, and the methods chosen depend on the particular situation. In this paper, the loopy belief propagation (LBP), which is the generalization of the forwards-backwards message passing algorithm to loopy graphs [46], has been adopted for parameter estimation and inference.

2.3. Conditional Random Field over Generalized Voronoi Graph

The generalized voronoi graph (GVG) is a topological map proposed by Choset and Burdick [47], and it has quite a few roles in navigation, localization and mapping [47, 48]. GVG is usually represented by meet-points (locations of three-way or more equidistance to obstacles) and edges (feasible paths of two-way equidistance to obstacles between meet-points) [49], which can be abstracted to a graph with nodes and edges. GVG can be built with different resolutions according to the requirements, and in this article the finest available scale is adopted for future applications such as object mapping. An example of GVG is shown in Fig. 1, and a conditional random field over GVG (CRFoGVG) is shown in Fig. 2.

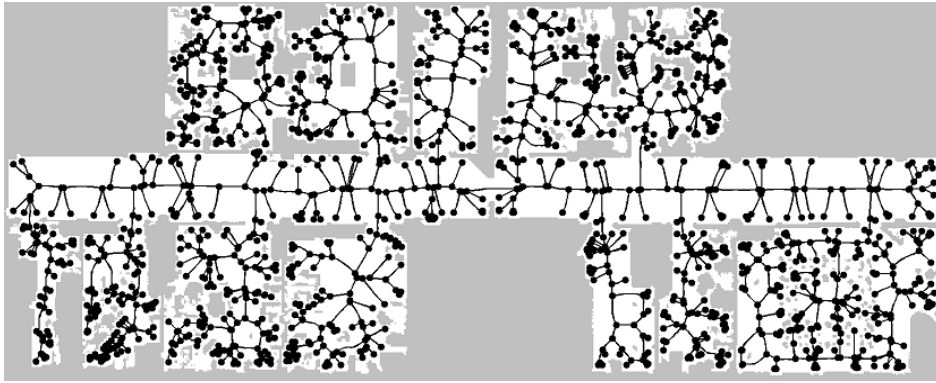


Fig. 1. An example of GVG on the grid map

By combining GVG and CRF, Friedman et al. proposed the voronoi random field (VRF) for multi-class place labeling, with the experimental results on leave-one-out training showing an

average accuracy of 91.7% on four maps [28]. With the motivation of improving the system’s generalization ability, we extended this work to semi-supervised learning so that it was capable of dealing with limited training data. The name VRF is somewhat misleading because the GVG simply provides the graph structure for CRF. Throughout this article we use the name CRFoGVG for ease of understanding.

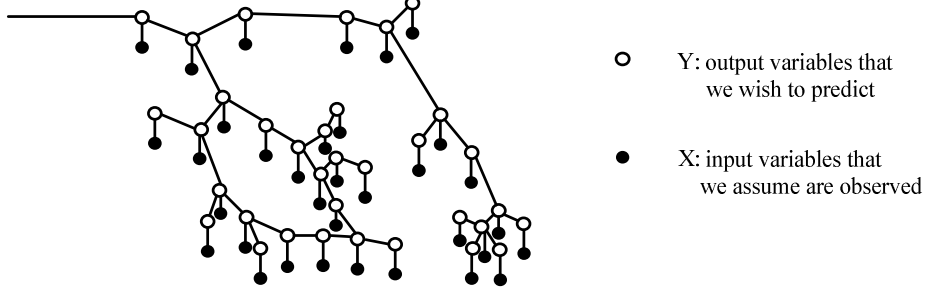


Fig. 2. An example of CRF over GVG

3. Semi-supervised Learning

Semi-supervised learning algorithms automatically exploit the unlabeled data in addition to the labeled data to improve learning performance, with the underlying belief that the unlabeled data contains some helpful information about the real data distribution [50, 51]. The learning framework reported in the present work includes two techniques which are co-training and graph-based semi-supervised learning. The former technique uses two classifiers to teach each other and the latter technique takes advantage of the spatial consistence of instances. It is to be noted that none of the current semi-supervised learning approaches are guaranteed to be superior, and sometimes the exploitation of unlabeled data may lead to performance degeneration [50].

3.1. Co-training-like Semi-supervised Learning

The co-training algorithm proposed by A. Blum and T. Mitchell [51] splits the features into two redundantly sufficient sets, and trains two classifiers on each feature set extracted from the labeled data. With the unlabeled data, these two classifiers iteratively examine new examples and add the most confidently labeled ones to the training set to improve the models [51, 52]. In this paper, we borrow the idea of co-training and propose a similar learning procedure according to our setup.

It is generally accepted that SVM has the prominent generalization ability and CRF has the advantage of using spatial connectivity. Although there are some attempts to combine their excellence by using the output scores of classifiers for independent and identically distributed

(i.i.d) data as the input of graphical models [28], these fully supervised approaches do not guarantee better performances in the challenging scenarios such as the one described in this paper. In addition, some emerging semi-supervised techniques emphasizing both the generalization ability and consistency have been reported to be successful in specific applications [37-39], and are worth exploring in the future, but the implementation involves complicated modifications for the current task. Therefore, we propose a co-training-like semi-supervised learning framework SPCoGVG by assuming that the agreed decisions of both classifiers are more likely to be correct, and this will allow us to learn from both the training data and the agreed test data to improve the classification accuracy. As shown in Fig. 3 and the following pseudo-code, the proposed algorithm iteratively picks up the test instances believed to be correctly identified, mixes them with the original training data and improves each model, until the agreement rate of SVM and CRFoGVG on test data cease to increase.

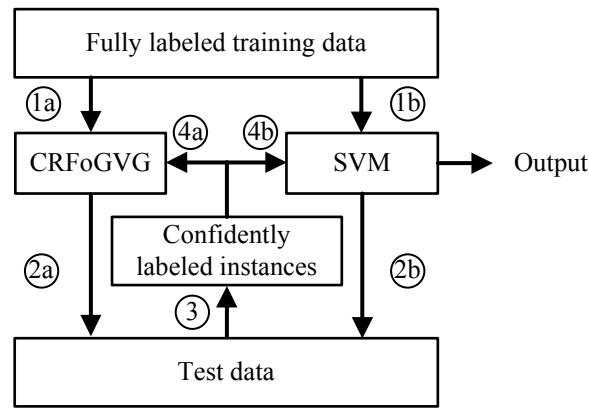


Fig. 3. Schematic flow chart of the proposed semi-supervised learning framework SPCoGVG.

Input:	V_{trn}, V_{tst}	The voronoi structures of the training and test maps
	X_{trn}, X_{tst}	The original training and test data collected from the GVG nodes
	Y_{trn}	Labels of training instances
Output:	Y_{tst}	Labels of test instances

Algorithm:

- 1: (1a) Generate CRFoGVG model M_{CRF} from X_{trn}, V_{trn} and Y_{trn}
- 2: (1b) Generate SVM model M_{SVM} from X_{trn} and Y_{trn}
- 3: (2a, 2b) Calculate the CRFoGVG prediction Y_{CRF} , the SVM prediction Y_{SVM} , and their agreement rate R
- 4: **Repeat**
- 5: Update R if necessary
- 6: (3, 4a) Take those agreed by both SVM and CRFoGVG predictions X_{extra}, Y_{extra} , train a new CRFoGVG model M_{CRF_new} from $X_{trn}, V_{trn}, Y_{trn}$ and $X_{extra}, V_{tst}, Y_{extra}$ (with partially labeled data)
- 7: (3, 4b) Mix X_{extra}, Y_{extra} with the original training data X_{trn}, Y_{trn} and generate a new SVM model M_{SVM_new}
- 8: (2a) Calculate a new CRFoGVG prediction Y_{CRF_new} on X_{tst}, V_{tst}
- 9: (2b) Calculate a new SVM prediction Y_{SVM_new} on X_{tst}
- 10: Calculate a new agreement rate R_{new} from Y_{SVM_new} and Y_{CRF_new}
- 11: **Until** $R_{new} \leq R$
- 12: Y_{tst} equals to previous Y_{CRF_new} or previous Y_{SVM_new}

3.2. CRF with Partial Labeled Data

The introduction of semi-supervised learning leads to a change in the graph structure, resulting in CRF having partially labeled data (CRF_{PL}) as shown in Fig. 4. As is the standard approach, ignoring fully-unlabeled data will probably break the graph structure and cause us to miss out on useful information [36]. Therefore, modeling the new structure requires an extension of the conventional CRF parameter estimation algorithm to handle partially labeled data by marginalizing out the unlabeled data, so that it can work on more realistic scenarios where not all the training data is properly labeled. Although the issue has been addressed in other literature on chain-like structures [53] or for scene segmentation applications [35], an

alternative computational strategy is adopted for this task considering the original implementation of CRF.

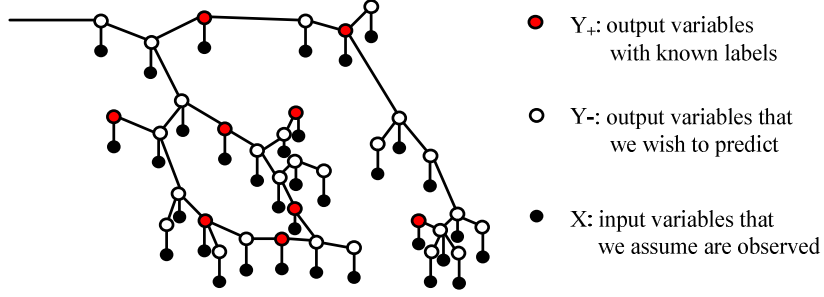


Fig. 4. An example of CRF_{PL}

By introducing the unlabeled data to the equation (7) in Section 2.2.1, \mathbf{y} is split into $[\mathbf{y}_K, \mathbf{y}_U]$ where \mathbf{y}_K are known labels and \mathbf{y}_U are unknown labels. \mathbf{y}_K is further defined as containing \mathbf{y}_{Ka} and \mathbf{y}_{Kb} , where \mathbf{y}_{Ka} corresponds to the nodes whose direct neighbors are all labeled, and \mathbf{y}_{Kb} are for the nodes whose direct neighbors contain unlabeled node(s).

Now, we can write:

$$p(\mathbf{y}_K | \mathbf{x}; \boldsymbol{\theta}) = \sum_{\mathbf{y}_U} p(\mathbf{y}_K, \mathbf{y}_U | \mathbf{x}; \boldsymbol{\theta}) \quad (12)$$

$$= \sum_{\mathbf{y}_U} \exp(\langle \phi(\mathbf{x}, \mathbf{y}), \boldsymbol{\theta} \rangle - z(\boldsymbol{\theta} | \mathbf{x})) \quad (13)$$

The parameter estimation problem becomes:

$$\boldsymbol{\theta} = \arg \max_{\boldsymbol{\theta}} P(Y_K | X; \boldsymbol{\theta}) \quad (14)$$

$$= \arg \max_{\boldsymbol{\theta}} \prod_{i=1}^m p(\mathbf{y}_{i,K} | \mathbf{x}_i; \boldsymbol{\theta}) \quad (15)$$

As discussed before, a $\lambda \|\boldsymbol{\theta}\|^2$ term is added to $P(Y_K | X; \boldsymbol{\theta})$ known as L2-regularization in the implementation. The negative log-posterior can be written as:

$$nll(\boldsymbol{\theta}) = -\sum_{i=1}^m \ln \sum_{\mathbf{y}_{i,U}} \exp(\langle \phi(\mathbf{x}_i, \mathbf{y}_i), \boldsymbol{\theta} \rangle - z(\boldsymbol{\theta} | \mathbf{x}_i)) \quad (16)$$

$$\begin{aligned} &= -\sum_{i=1}^m \ln \sum_{\mathbf{y}_{i,U}} \exp(\langle \phi(\mathbf{x}_i, \mathbf{y}_{i,K}), \boldsymbol{\theta} \rangle + \langle \phi(\mathbf{x}_i, \mathbf{y}_{i,U}, \mathbf{y}_{i,Kb}), \boldsymbol{\theta} \rangle - z(\boldsymbol{\theta} | \mathbf{x}_i)) \\ &= -\sum_{i=1}^m \langle \phi(\mathbf{x}_i, \mathbf{y}_{i,K}), \boldsymbol{\theta} \rangle - \sum_{i=1}^m \ln \sum_{\mathbf{y}_{i,U}} \exp(\langle \phi(\mathbf{x}_i, \mathbf{y}_{i,U}, \mathbf{y}_{i,Kb}), \boldsymbol{\theta} \rangle) + \sum_{i=1}^m z(\boldsymbol{\theta} | \mathbf{x}_i) \end{aligned}$$

$$\text{By defining } z_U(\boldsymbol{\theta} | \mathbf{x}) = \ln \sum_{\mathbf{y}_U} \exp(\langle \phi(\mathbf{x}, \mathbf{y}_U, \mathbf{y}_{Kb}), \boldsymbol{\theta} \rangle) \quad (17)$$

$$nll(\boldsymbol{\theta}) = -\sum_{i=1}^m \left[\langle \phi(\mathbf{x}_i, \mathbf{y}_{i,K}), \boldsymbol{\theta} \rangle + z_U(\boldsymbol{\theta} | \mathbf{x}_i) - z(\boldsymbol{\theta} | \mathbf{x}_i) \right] \quad (18)$$

Now, the gradient of the negative log-likelihood function is:

$$\nabla nll(\boldsymbol{\theta}) = -\sum_{i=1}^m \left[\phi(\mathbf{x}_i, \mathbf{y}_{i,K}) + E_{p(\mathbf{y}_U | \mathbf{x}_i; \boldsymbol{\theta})} [\phi(\mathbf{x}_i, \mathbf{y}_U, \mathbf{y}_{Kb})] - E_{p(\mathbf{y} | \mathbf{x}_i; \boldsymbol{\theta})} [\phi(\mathbf{x}_i, \mathbf{y})] \right] \quad (19)$$

where $E_{p(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta})} [\phi(\mathbf{x}, \mathbf{y})] = \sum_{\mathbf{y}} p(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta}) \phi(\mathbf{x}, \mathbf{y})$

and $E_{p(\mathbf{y}_U | \mathbf{x}; \boldsymbol{\theta})} [\phi(\mathbf{x}, \mathbf{y}_U, \mathbf{y}_{Kb})] = \sum_{\mathbf{y}_U} p(\mathbf{y}_U | \mathbf{x}; \boldsymbol{\theta}) \phi(\mathbf{x}, \mathbf{y}_U, \mathbf{y}_{Kb})$

It is to be noted that the presence of $z_U(\boldsymbol{\theta} | \mathbf{x})$ makes the problem non-convex and theoretically the LBP algorithm may trap in a local minima.

3.3. Labels and Features

3.3.1. Labels

In the research of place classification, although many researchers agree that the robot's space representation must at least partially overlap with human spatial concepts [17, 54], there remains controversy over the selection of target class labels.

Based on observation from human spatial cognition, some researchers believe that space should be categorized not only geometrically but also functionally [17, 26]. For example, Martinez-Gomez and Caputo suggested a subdivision of rooms in terms of their appearance, the activities people usually perform in them, and the objects they contain [33]. These systems usually require rich sensory modalities and hierarchical concept modeling, so that a robot can integrate its understanding about distinct topological areas with its knowledge about the presence of certain objects [17]. On the contrary, systems without multiple sensory modalities tend to simplify the classification task and only provide basic discrimination.

Another problem lies with the treatment of a door/doorway, which indicates the transition between different spatial regions, and is claimed to be the place where most errors occur [7, 9]. In addition, there are cases where the door/doorway is not described by an obvious separator, or does not even exist (e.g. cubicle-corridor transition)[55]. Therefore, in some applications, door/doorway are detected separately or inferred from topological analysis [17].

3.3.2. Simple Geometric Features

In machine learning tasks, the feature construction is of great importance because it can impact on the generalization ability and overheads of the system [21]. Good features represent the target concepts well, and at the same time minimize the within-class variability and maximize the between-class variability [9]. Various types of features from the 2D laser range data have been used in the literature including spectral features [3] and single-valued features which capture statistical and geometric information [20]. For this study, the data collected from the laser range finder is a non-negative beam sequence $\mathbf{B} = \{b_1, b_2, \dots, b_n\}$ corresponding to a constant angle interval, which represents a 2D point set $\mathbf{P} = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ in the Euclidean Plane. The following twenty four features, including three features originally derived by Hjorth to describe the time domain signal [56] and a subset of simple geometric features, have been adopted.

- The area (f_1), perimeter (f_2) and the normalized circularity (f_3) of the polygon specified by the observed point set \mathbf{P} [57]; the quotient (f_4) of the area and the perimeter of the above-mentioned polygon.

$$f_1 = \frac{1}{2} \left| \sum_{i=1}^n (x_i y_{i+1} - x_{i+1} y_i) \right| \quad (20)$$

Where $x_{n+1} = x_1$ and $y_{n+1} = y_1$

$$f_2 = \sum_{i=1}^n \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2} \quad (21)$$

Where $x_{n+1} = x_1$ and $y_{n+1} = y_1$

$$f_3 = \frac{4 \cdot \pi \cdot f_1}{f_2^2} \quad (22)$$

$$f_4 = \frac{f_1}{f_2} \quad (23)$$

- The average (f_5) and the standard deviation (f_6) of the beam length:

$$f_5 = \frac{1}{n} \sum_{i=1}^n b_i \quad (24)$$

$$f_6 = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (b_i - f_5)^2} \quad (25)$$

- The average (f_7) and the standard deviation (f_8) of the normalized beam length:

$$f_7 = \frac{1}{n} \sum_{i=1}^n \frac{b_i}{b_{\max}} \quad (26)$$

$$f_8 = \sqrt{\frac{1}{n-1} \sum_{i=1}^n \left(\frac{b_i}{b_{\max}} - f_7 \right)^2} \quad (27)$$

- The average (f_9), the standard deviation (f_{10}) and the normalized average (f_{11}) of the difference between the length of consecutive beams:

We define $S = \{s_1, s_2, \dots, s_n\}$, where $s_i = b_{i+1} - b_i$, and $b_{n+1} = b_1$

$$f_9 = \frac{1}{n} \sum_{i=1}^n s_i \quad (28)$$

$$f_{10} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (s_i - f_9)^2} \quad (29)$$

f_{11} is the same as f_9 in terms of expression but uses preprocessed beam ranges which applies a maximum range threshold as the upper limit on the original beam ranges.

- The average (f_{12}) and the standard deviation (f_{13}) of the relation between the length of consecutive beams:

$$f_{12} = \frac{1}{n} \sum_{i=1}^n \frac{b_i}{b_{i+1}} \quad (30)$$

$$f_{13} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n \left(\frac{b_i}{b_{i+1}} - f_{12} \right)^2} \quad (31)$$

Where $b_{n+1} = b_1$

- The average (f_{14}), the normalized average (f_{15}), the standard deviation (f_{16}) and the normalized standard deviation (f_{17}) of the distances between the centroid and the shape boundary:

Given $\mathbf{c} = (c_x, c_y)$ being the center of mass (centroid) of the polygon [58], we define:

$$\mathbf{D} = \{d_1, d_2, \dots, d_n\}, \text{ where } d_i = \sqrt{(x_i - c_x)^2 + (y_i - c_y)^2},$$

$$f_{14} = \frac{1}{n} \sum_{i=1}^n d_i \quad (32)$$

$$f_{15} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (d_i - f_{14})^2} \quad (33)$$

$$f_{16} = \frac{1}{n} \sum_{i=1}^n \frac{d_i}{d_{\max}} \quad (34)$$

$$f_{17} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n \left(\frac{d_i}{d_{\max}} - f_{16} \right)^2} \quad (35)$$

- The lengths of the major axis (f_{18}) and minor axis (f_{19}) of the ellipse that approximates the \mathbf{P} ; the quotient of the lengths of the above-mentioned major and minor axis (f_{20}).

Given z_v being the Cartesian Fourier descriptor of the boundary curve, z_1 and z_{-1} describe an ellipse which is believed to be an approximation of the observed point set \mathbf{P} [59, 60]:

$$f_{18} = |z_1| + |z_{-1}| \quad (36)$$

$$f_{19} = \left| |z_1| - |z_{-1}| \right| \quad (37)$$

$$f_{20} = \frac{f_{18}}{f_{19}} \quad (38)$$

- Kurtosis (f_{21}) of the beam sequence \mathbf{B} , reflecting the degree of peakedness of a distribution [61, 62]:

$$f_{21} = \frac{\frac{1}{n} \sum_{i=1}^n (b_i - f_5)^4}{f_6^4} - 3 \quad (39)$$

- The activity (f_{22}), mobility (f_{23}) and complexity (f_{24}) of the beam sequence. Given m_i being the i^{th} order spectral moment of a signal as defined in the literature [56]:

$$f_{22} = m_0 \quad (40)$$

$$f_{23} = \sqrt{\frac{m_2}{m_0}} \quad (41)$$

$$f_{24} = \sqrt{\frac{m_4}{m_0}} \quad (42)$$

We have shown that not all these twenty four features are required for a high system accuracy because the classification relies strongly on a subset of them [21]. However, compared with the dimensions of raw data, the dimensions of the current features would neither cause over fitting problems nor significantly affect the performance of the classifier. Therefore, we leave it to the classification algorithm to weight features through the parameter estimation.

3.3.3. Node and Edge Features

CRFoGVG modeling uses features associated with both the nodes and the edges. As per the common practice, it is assumed that the edge features share the node features of two end nodes [44]. In addition, the following four GVG edge-length-based connectivity features have been added to the node features, and are also shared as edge features.

- The maximum value, minimum value, average value and the standard deviation of the Euclidean distances to the neighbors (GVG edge lengths).

Furthermore, we introduce another set of features called centrality, which is a family of functions assigning numerical values to each node of a graph. As a common descriptor of the importance of an individual node, the concept of centrality has prevailing popularity in the social network and biological network analysis [63]. There are quite a few types of centralities available in different applications [63-65]; however, in this paper, we opt to choose the four most common centralities namely: degree centrality; eigenvector centrality; closeness centrality; and modified betweenness centrality.

Mathematically, a graph can be represented by a symmetric adjacency matrix $A = [a_{ij}]_{N \times N}$, where N is the number of nodes. The element of A is defined as:

$$a_{ij} = \begin{cases} 1 & \text{if there is an edge between nodes } i \text{ and } j \\ 0 & \text{otherwise} \end{cases} \quad (43)$$

- Degree centrality, the number of edges attached to a node, is defined as:

$$C_{deg}(i) = \sum_{j=1}^N a_{ij} \quad (44)$$

- Eigenvector centrality, an extension of the degree centrality by considering the number of connections both of a node and of its neighbors, is defined as:

$$C_{eiv}(i) = \frac{1}{\lambda} \sum_{j=1}^N a_{ij} \cdot C_{eiv}(j) \quad (45)$$

where λ is a constant.

Equation (45) can be rewritten as:

$$\lambda C_{eiv} = A C_{eiv} \quad (46)$$

Therefore, C_{eiv} is an eigenvector of the adjacency matrix A with eigenvalue λ .

- Closeness centrality, the reciprocal of the sum of all i -related geodesic distances within the network [63], is defined as:

$$C_{clo}(i) = \frac{1}{\sum_{j=1}^N dist(i, j)} \quad (47)$$

where $dist(i, j)$ is the geodesic distance defined as the shortest path in terms of number of edges traversed between a specified pair of nodes [64].

- Shortest-path between centrality, the fraction of geodesic paths between other nodes that the evaluated node falls on [64], is defined as:

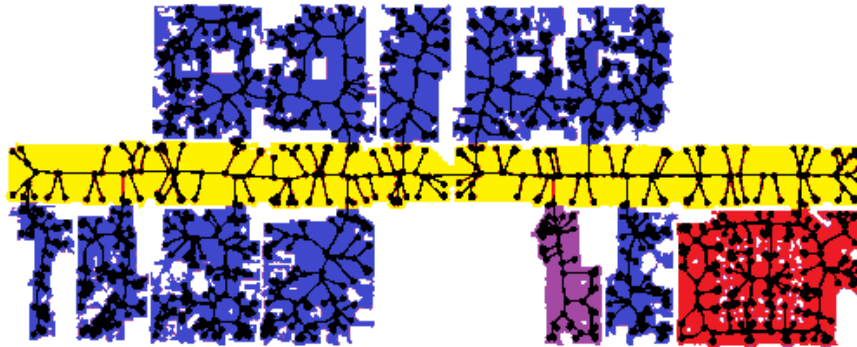
$$C_{spb}(i) = \sum_{m \in V \wedge m \neq i} \sum_{n \in V \wedge n \neq i} \frac{\sigma_{mn}(i)}{\sigma_{mn}} \quad (48)$$

where σ_{mn} is the number of shortest paths between nodes m and n , and $\sigma_{mn}(i)$ is the number of shortest paths between nodes m and n using node i as an interior node.

In the current implementation, considering the properties of the GVG, we modify the above definition by assigning the maximum between centralities of a node's 2-nearest neighbors to the node being evaluated.

4. Experimental Setup

The data sets used in the following experiments were collected in six real-world grid maps of different international university indoor environments (Fig. 5). A simulated robot equipped with a virtual on-board 2D laser range finder which has a maximum range of 30 meters and a horizontal field of view of 360° navigated through the grid maps while capturing data. We opted to generate the simulated laser data on real grid maps due to the infeasibility of visiting all the international sites for data collection. We have assessed the feasibility of using publically available data sets as an option; however, this met with minimum success due to the unavailability of observations made at estimated GVG nodes.



Bldg. 79, Uni-Freiburg (Fr79)

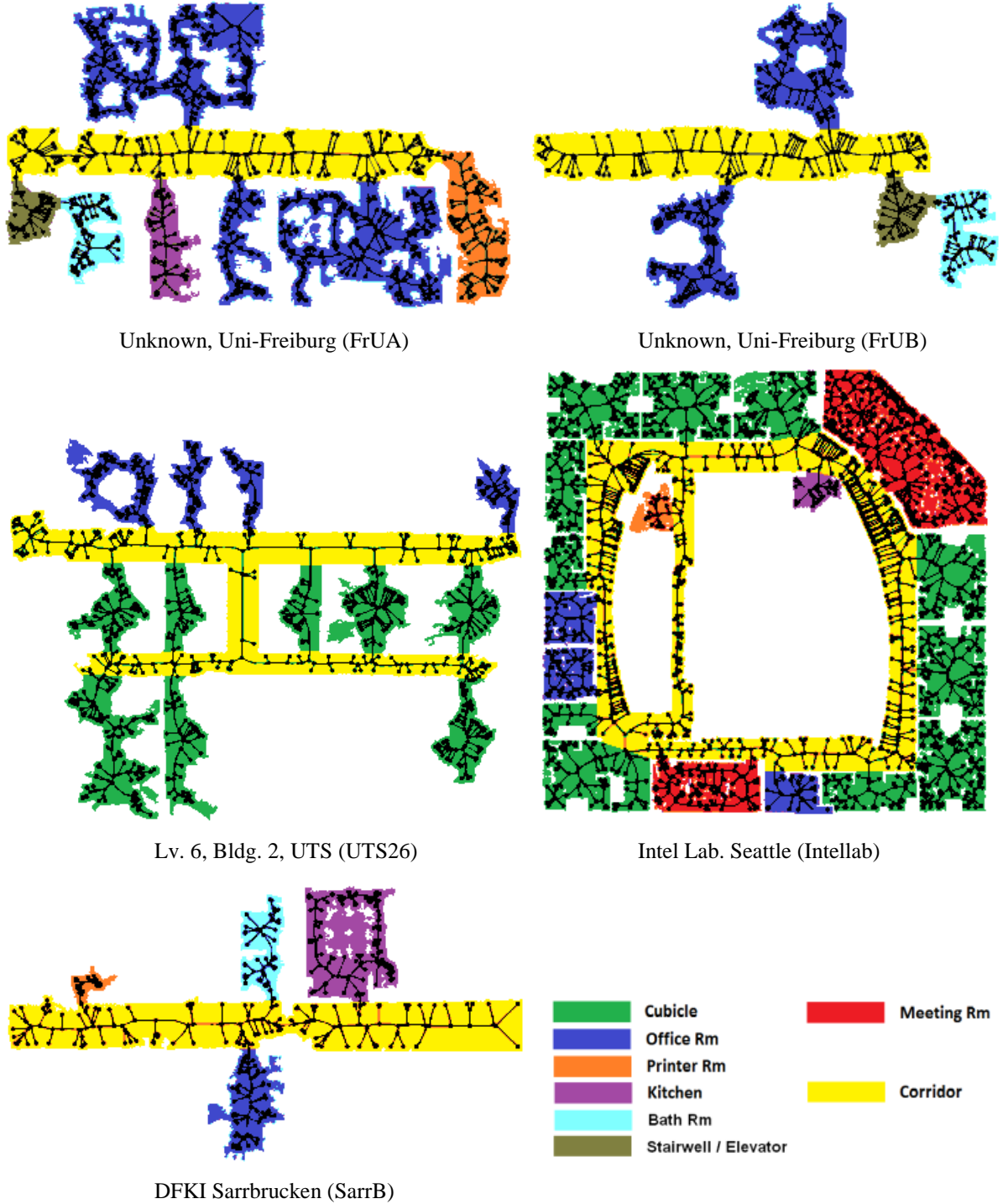


Fig. 5. Real-world grid maps from different international universities. These images are not to scale and unconfirmed parts are removed. Areas filled with different colours (other than black and white) represent free space belonging to a variety of space types according to the legend in the bottom right corner. The GVG of the individual map is drawn in black where a dot and line represents node and edge respectively. (better viewed in colour)

The human defined spaces given in Fig. 5 are of high complexity and require substantial information to classify. The sensor used in this work is a two dimensional laser measurement system, and hence does not contain enough discriminative information to segregate all the

given categories, even with the human eye. Therefore, we have redefined the target classes as: *Class 1* - space designed for small number of individuals including cubicle, office, printer room, kitchen, bathroom, stairwell and elevator; *Class 2* - space for group activities including meeting room and laboratory; and *Class 3* - corresponds only to corridor. The utilization of three classes in the current implementation is justified as the focus of this paper is targeted at implementing a proof of concept for a generalized solution rather than a complex semantic classifier. However, it could be noted that the framework proposed in this paper will be extended to many classes observed with complex sensors like the ones that will produce RGB-D data in our future work. Among the six environments, two of them contain spaces covering all three classes (referred to as complete maps) and the other four contain parts of these classes (referred to as incomplete maps).

5. Results and Discussion

The results and discussion section is divided into subsections for the convenience of the reader. First, a benchmark is established by training and testing on the same data. Then a practically feasible fully supervised leave-one-out (LOO) training is performed and tested. Thereafter, a more general and challenging leave-many-out (LMO) training is performed with promising results.

5.1. Fully Supervised Comprehensive Training

In this scenario, the data in each map was used for training the SVM and CRFoGVG classifiers and tested on the same data. The method is hereinafter referred to as fully supervised comprehensive training (FSCT). Although there is no significant meaning to the results in terms of practical deployability, this serves as our ceiling or benchmark. It can be seen in Table 1 that CRFoGVG in general outperforms SVM through correctly modeling contextual relationships.

Table 1: FSCT results from SVM and CRFoGVG models

Map	Overall accuracy (SVM)	Overall accuracy (CRFoGVG)
UTS26	97.02%	97.90%
Fr79*	94.94%	99.51%
FrUA	98.18%	99.45%
FrUB	100.00%	99.19%
Intellab*	92.43%	98.43%
SarrB	100.00%	100.00%
Average	97.10%	99.08%

5.2. Fully Supervised Leave-one-out Training

This part of the experiment is based on fully supervised leave-one-out training. Five of the total six data sets were selected as training data, and the remaining data set was used for testing. The results using SVM and CRFoGVG are summarized in Table 2. As expected, it could be noted that both SVM and CRFoGVG have an overall reduction in accuracies compared with those presented in Table 1. In addition, a further investigation on the low accuracies in both cases shown on the map *Intellab* reveals that the existence of furniture resulting in high similarity of the 2D laser range/bearing observations between office rooms and meeting rooms, and neither the narrow corridors nor the small meeting rooms in *Intellab* have similar attributes in other maps.

Table 2: Performance of **leave-one-out** training using **SVM** and **CRFoGVG**

Map	Overall accuracy (SVM)	Overall accuracy (CRFoGVG)
UTS26	93.35%	95.53%
Fr79*	92.23%	99.38%
FrUA	97.47%	99.13%
FrUB	99.19%	98.87%
Intellab*	85.47%	76.78%
SarrB	93.71%	92.84%
Average	93.57%	93.76%

5.3. Fully Supervised Leave-many-out Training

The idea of our research is to improve the generalization ability under the constraint of limited training data. Therefore, in this experiment we evaluate the performance of the original SVM and CRFoGVG based solutions with leave-many-out training data. As there are two complete maps, without loss of generality, they are used in turn as training data, and the remaining five data sets serve as test data. Results from the SVM based approach, as shown in Table 3 and Table 4, indicate that the model generated by training with *Fr79* does not generalize the target concepts well, while the model generated by training with *Intellab* data generates a better model. On the contrary, as shown in Table 5 and Table 6, the CRFoGVG model generated from the map *Fr79* has significantly better results probably due to the more general GVG structure of the map *Fr79*. However, CRFoGVG still has sporadic poor accuracies when tested on different maps.

Table 3: Performance of SVM trained on *FR79* with **leave-many-out** training

Map	Overall accuracy	Class-specific accuracy		
		<i>Class 1</i>	<i>Class 2</i>	<i>Class 3</i>
UTS26	83.54%	100.00%	---	38.36%
FrUA	92.72%	99.53%	---	58.57%
FrUB	80.74%	100.00%	---	33.52%
Intellab*	79.89%	85.46%	73.57%	74.51%
SarrB	82.43%	100.00%	---	52.91%
Average	83.86%			

Table 4: Performance of SVM trained on *Intellab* with **leave-many-out** training

Map	Overall accuracy	Class-specific accuracy		
		<i>Class 1</i>	<i>Class 2</i>	<i>Class 3</i>
UTS26	87.74%	94.27%	---	69.84%
Fr79*	88.34%	89.96%	80.09%	88.58%
FrUA	96.04%	98.67%	---	82.86%
FrUB	97.25%	99.54%	---	91.62%
SarrB	85.68%	97.92%	---	65.12%
Average	91.01%			

Table 5: Performance of CRFoGVG trained on *FR79* with **leave-many-out** training

Map	Overall accuracy	Class-specific accuracy		
		<i>Class 1</i>	<i>Class 2</i>	<i>Class 3</i>
UTS26	96.15%	98.09%	---	90.82%
FrUA	99.29%	99.34%	---	99.05%
FrUB	98.87%	98.86%	---	98.88%
Intellab*	81.12%	97.06%	76.26%	54.76%
SarrB	99.78%	100.00%	---	99.42%
Average	95.04%			

Table 6: Performance of CRFoGVG trained on *Intellab* with **leave-many-out** training

Map	Overall accuracy	Class-specific accuracy		
		<i>Class 1</i>	<i>Class 2</i>	<i>Class 3</i>
UTS26	86.43%	100.00%	---	49.18%
Fr79*	97.66%	99.82%	100.00%	85.83%
FrUA	97.31%	99.62%	---	85.71%
FrUB	96.60%	96.36%	---	97.21%
SarrB	59.00%	46.02%	---	80.81%
Average	87.40%			

This shows that even though the leave-many-out strategy is closer to the practical application, it failed to deliver higher classification accuracies while used in the fully supervised framework. It has a very high variation in accuracies. This leads us to the semi-supervised strategy.

5.4. Semi-supervised Leave-many-out Training

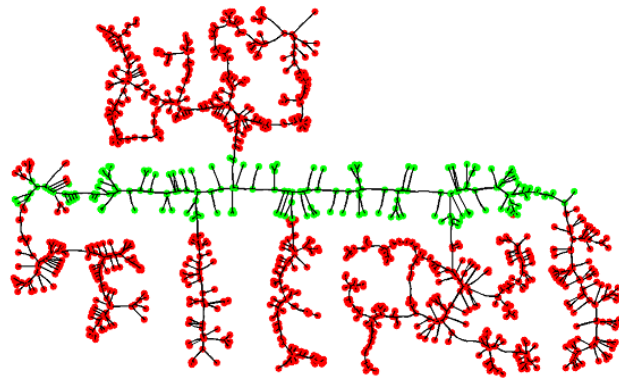
The generalization ability of the SPCoGVG framework proposed in Section 3 is analyzed here. It uses SVM and CRF_{PL} in a co-training framework. Similar to the previous experiment, one complete map at a time was used as training data and the remaining maps served as test data. The learning process does not stop until the useful information from the test data is believed to be fully exploited. The classification results are presented in Table 7 and Table 8. Comparing those with the results from the fully supervised leave-many-out training (Table 3 to Table 6), results from the semi-supervised leave-many-out training show convincing improvements in many aspects. Fig. 6 visualizes the classification results given in Table 7 with satisfactory high classification accuracies, although there still exist some errors in *Class 3* (corridor) of the maps *UTS26* and *Intellab* due to their uniqueness. However, the main point to note is the lower variation in accuracies than that of the fully supervised accuracies, which will be discussed next.

Table 7: Performance of SPCoGVG trained on *FR79*

Map	Overall accuracy	Class-specific accuracy		
		<i>Class 1</i>	<i>Class 2</i>	<i>Class 3</i>
UTS26	89.84%	100.00%	---	61.97%
FrUA	97.71%	99.34%	---	89.52%
FrUB	99.19%	100.00%	---	97.21%
Intellab*	86.89%	95.01%	84.01%	73.81%
SarrB	93.71%	100.00%	---	83.14%
Average	93.47%			

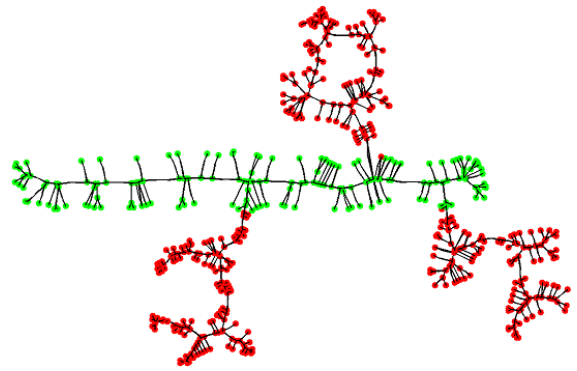
Table 8: Performance of SPCoGVG trained on *Intellab*

Map	Overall accuracy	Class-specific accuracy		
		<i>Class 1</i>	<i>Class 2</i>	<i>Class 3</i>
UTS26	90.72%	99.76%	---	66.89%
Fr79*	92.04%	94.98%	78.35%	91.34%
FrUA	96.52%	99.15%	---	83.33%
FrUB	98.71%	100.00%	---	95.53%
SarrB	88.72%	99.31%	---	70.93%
Average	93.39%			



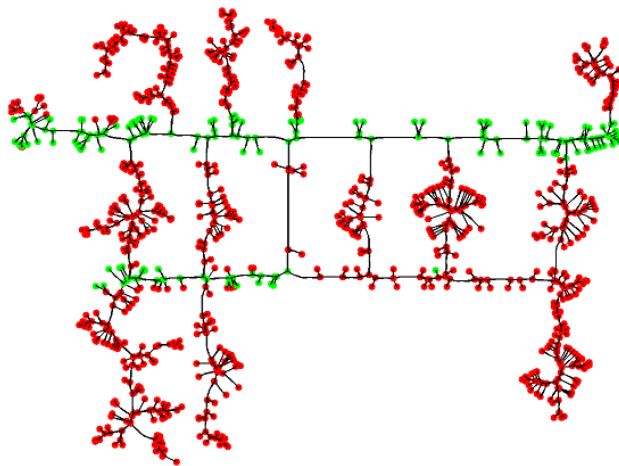
Unknown, Uni-Freiburg (FrUA)

Accuracy 97.71%



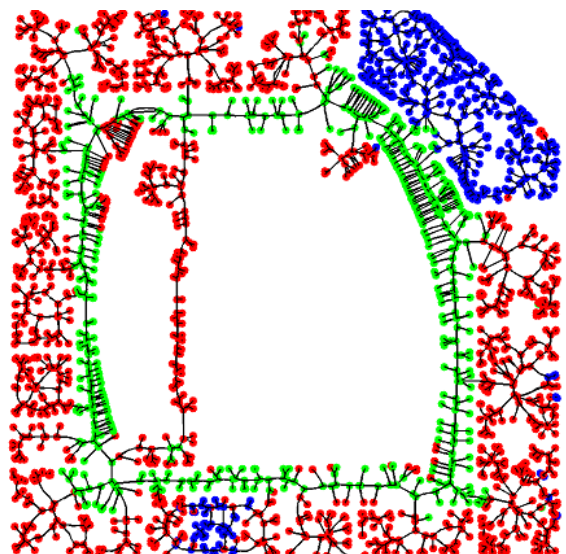
Unknown, Uni-Freiburg (FrUB)

Accuracy 99.19%



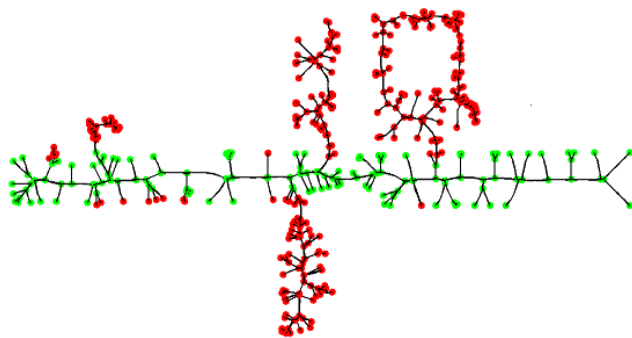
Lv. 6, Bldg. 2, UTS (UTS26)

Accuracy 89.84%



Intel Lab. Seattle (Intellab)

Accuracy 86.89%



DFKI Sarrbrucken (SarrB)

Accuracy 93.71%



Fig. 6. Test results corresponding to Table 7. The GVG nodes of the individual map are labeled as belonging to three semantic classes (see the bottom right corner for labels).

Fig. 7 shows the overall comparisons of the results in the methods reported in Section 5.1 to 5.4 in the box-and-whisker diagram. It could be noted that the fully supervised comprehensive training (trained and tested on same data) gave rise to the highest accuracies with the lowest variation in accuracies (SVM: $97.10\% \pm 2.98\%$, CRFoGVG: $99.08\% \pm 0.77\%$). Leave-one-out fully supervised training also provides reasonably accurate results (SVM: $93.57\% \pm 4.78\%$, CRFoGVG: $93.76\% \pm 8.70\%$). Averaging over all cases, leave-many-out fully supervised training demonstrates the worst classification accuracies and the largest variations (SVM: $87.44\% \pm 6.18\%$, CRFoGVG: $91.22\% \pm 12.87\%$), while the proposed SPCoGVG has improved classification accuracies with reduced variations ($93.43\% \pm 4.41\%$). This is in fact a convincing fact towards generalization. It also demonstrates that the proposed leave-many-out SPCoGVG is competitive with the fully supervised leave-one-out training scheme.

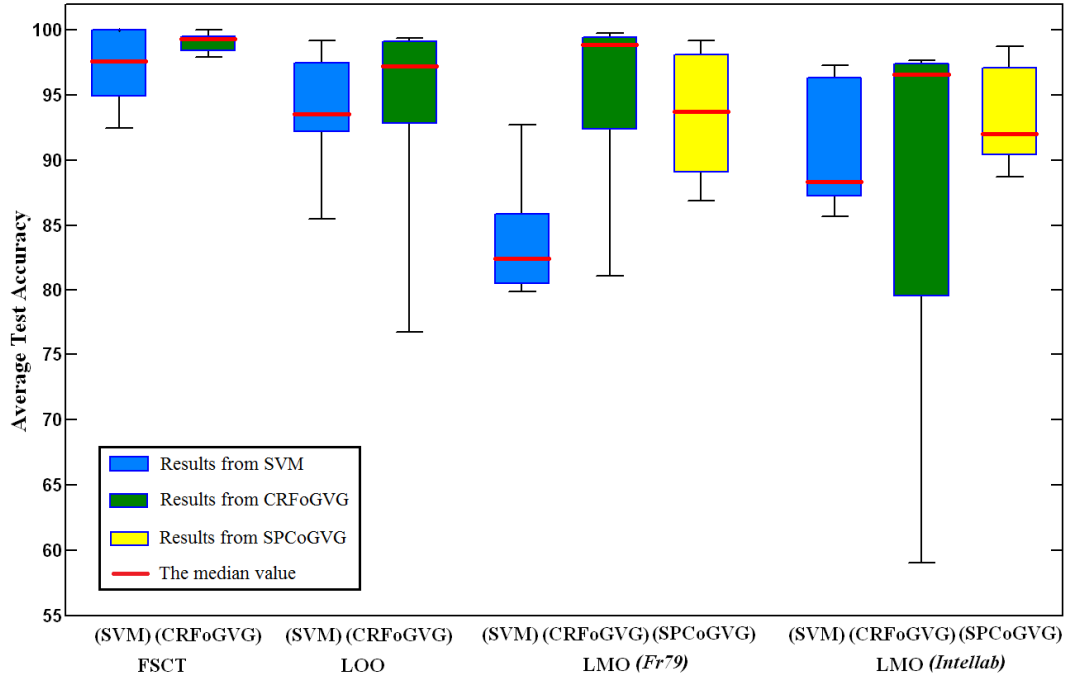


Fig. 7. Overall comparisons of experimental results shown in boxplot (better viewed in colour)

The results of a further comparison in leave-many-out scenario only, as shown in Table 9, also supports our argument that extra learning “on-the-spot” improves the generalizability and stability of the system.

Table 9: Performance of proposed method in comparison to other classifiers in leave-many-out scenario

	SVM	CRFoGVG	SPCoGVG
Overall accuracy (over 10 entries)	$87.44\% \pm 6.18\%$	$91.22\% \pm 12.87\%$	$93.43\% \pm 4.41\%$
Overall class-specific accuracy (over 22 entries)	$80.68\% \pm 20.30\%$	$88.78\% \pm 17.23\%$	$89.25\% \pm 12.12\%$

6. Conclusion

This paper described and proposed a semi-supervised place classification framework (SPCoGVG) with the objective of improving the generalizability of a learning system. It was our intention to train with one data set collected at one university environment and test it at other international university environments (leave-many-out training). We have proposed the SPCoGVG based on the SVM and CRFoGVG because both SVM and CRFoGVG have good inherent capabilities and they compensate each other. The problem of training CRF with partially labeled data has been successfully solved using the CRF parameter estimation with the maximum conditional likelihood estimation marginalizing the unknown labels.

Experimental results showed that with abundant and diversified training data, both the SVM and CRFoGVG based approaches generalized well on test data. However, with leave-many-out training they often gave rise to poor accuracies. The proposed co-training-like semi-supervised learning algorithm SPCoGVG has proven to be having comparable results with those of the leave-one-out training schemes showing the improved generalizability. The generalizability of the proposed algorithm was further reinforced by the lower variations in the testing accuracies in different environments.

In this work, we opted to assume that the environments consist of three general classes. We are in the process of expanding it to many classes based on additional sensors which can provide informative features.

References

- [1] A. Ranganathan, L. Jongwoo, Visual place categorization in maps, in: Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), San Francisco, 2011, pp. 3982-3989.
- [2] L. Shi, S. Kodagoda, G. Dissanayake, Semantic Grid Map Building, in: Proc. Australasian Conf. on Robotics and Automation (ACRA), Australasian Conference on Robotics and Automation, Brisbane, 2010.
- [3] A. Poncela, C. Urdiales, B. Fernandez-Espejo, F. Sandoval, Place characterization for navigation via behaviour merging for an autonomous mobile robot, in: Proc. 14th IEEE Mediterranean Electrotechnical Conf. (MELECON), Ajaccio, 2008, pp. 350-355.
- [4] C. Galindo, J.-A. Fernandez-Madrigal, J. Gonzalez, A. Saffiotti, Robot task planning using semantic maps, *Robotics and Autonomous Systems*, 56 (2008) 955-966.
- [5] F. Ramos, B. Upcroft, S. Kumar, H. Durrant-Whyte, A Bayesian approach for place recognition, *Robotics and Autonomous Systems*, 60 (2012) 487-497.
- [6] A. Aydemir, K. Sjoo, J. Folkesson, A. Pronobis, P. Jensfelt, Search in the real world: Active visual object search based on spatial relations, in: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), Shanghai, 2011, pp. 2818-2824.

- [7] L. Yuan, K.C. Chan, C.S.G. Lee, Robust Semantic Place Recognition with Vocabulary Tree and Landmark Detection, in: Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS) workshop, San Francisco, 2011.
- [8] A. Pronobis, P. Jensfelt, Hierarchical multi-modal place categorization, in: Proc. European Conf. on Mobile Robots (ECMR), Orebro, 2011.
- [9] A. Pronobis, B. Caputo, P. Jensfelt, H.I. Christensen, A realistic benchmark for visual indoor place recognition, *Robotics and Autonomous Systems*, 58 (2010) 81-96.
- [10] B. Steder, M. Ruhnke, S. Grzonka, W. Burgard, Place recognition in 3D scans using a combination of bag of words and point feature based relative pose estimation, in: Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), IEEE, San Francisco, 2011, pp. 1249-1255.
- [11] F. Orabona, C. Castellini, B. Caputo, J. Luo, G. Sandini, Indoor place recognition using online independent support vector machines, in: Proc. British Machine Vision Conf. (BMVC), Coventry, 2007, pp. 1090-1099.
- [12] E. Fazl-Ersi, J. Elder, J. Tsotsos, Hierarchical Classifiers for Robust Topological Robot Localization, *Journal of Intelligent & Robotic Systems*, (2012) 1-17.
- [13] S. Vasudevan, S. Gachter, V. Nguyen, R. Siegwart, Cognitive maps for mobile robots - an object based approach, *Robotics and Autonomous Systems*, 55 (2007) 359-371.
- [14] S. Vasudevan, R. Siegwart, Bayesian space conceptualization and place classification for semantic maps in mobile robotics, *Robotics and Autonomous Systems*, 56 (2008) 522-537.
- [15] O.M. Mozos, P. Jensfelt, H. Zender, G.J.M. Kruijff, W. Burgard, From labels to semantics: An integrated system for conceptual spatial representations of indoor environments for mobile robots, in: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA) workshop, Rome, 2007, pp. 33-40.
- [16] A. Pronobis, P. Jensfelt, K. Sjoo, H. Zender, G.J.M. Kruijff, O.M. Mozos, W. Burgard, Semantic Modelling of Space, in: H.I. Christensen, G.-J.M. Kruijff, J.L. Wyatt (Eds.) *Cognitive Systems*, Springer Berlin Heidelberg, 2010, pp. 165-221.
- [17] H. Zender, O.M. Mozos, P. Jensfelt, G.J.M. Kruijff, W. Burgard, Conceptual spatial representations for indoor mobile robots, *Robotics and Autonomous Systems*, 56 (2008) 493-502.
- [18] A. Pronobis, P. Jensfelt, Large-scale semantic mapping and reasoning with heterogeneous modalities, in: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), Saint Paul, 2012, pp. 3515-3522.
- [19] A. Tapus, G. Ramel, L. Dobler, R. Siegwart, Topology learning and recognition using Bayesian programming for mobile robot navigation, in: Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), Sendai, 2004, pp. 3139-3144 vol.3134.
- [20] O.M. Mozos, C. Stachniss, W. Burgard, Supervised Learning of Places from Range Data using AdaBoost, in: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), Barcelona, 2005, pp. 1730-1735.
- [21] L. Shi, S. Kodagoda, G. Dissanayake, Laser range data based semantic labeling of places, in: Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), Taipei, 2010, pp. 5941-5946.
- [22] L. Shi, S. Kodagoda, G. Dissanayake, Multi-class classification for semantic labeling of places, in: Proc. Int. Conf. on Control Automation Robotics & Vision (ICARCV), Singapore, 2010, pp. 2307-2312.
- [23] A. Swadzba, S. Wachsmuth, Categorizing Perceptions of Indoor Rooms Using 3D Features, in: Proc. Joint IAPR Int. Workshop on Structural, Syntactic, and Statistical Pattern Recognition (SSPR & SPR), Springer, Orlando, 2008, pp. 734-744.
- [24] O.M. Mozos, H. Mizutani, R. Kurazume, T. Hasegawa, Categorization of Indoor Places Using the Kinect Sensor, *Sensors*, 12 (2012) 6695-6711.
- [25] L. Shi, S. Kodagoda, R. Ranasinghe, Fast Indoor Scene Classification Using 3D Point Clouds, in: Proc. Australasian Conf. on Robotics and Automation (ACRA), Melbourne, 2011.
- [26] A. Pronobis, L. Jie, B. Caputo, The more you learn, the less you store: Memory-controlled incremental SVM for visual place recognition, *Image and Vision Computing*, 28 (2010) 1080-1097.
- [27] C. Sutton, A. McCallum, An Introduction to Conditional Random Fields for Relational Learning, in: L. Getoor, B. Taskar (Eds.) *Introduction to statistical relational learning*, The MIT Press, 2007, pp. 93.

- [28] S. Friedman, H. Pasula, D. Fox, Voronoi random fields: Extracting the topological structure of indoor environments via place labeling, in: Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI), Hyderabad, 2007, pp. 2109-2114.
- [29] S. Kim, I.S. Kweon, Simultaneous place and object recognition using collaborative context information, Image and Vision Computing, 27 (2009) 824-833.
- [30] O.M. Mozos, R. Triebel, P. Jensfelt, A. Rottmann, W. Burgard, Supervised semantic labeling of places using information extracted from sensor data, Robotics and Autonomous Systems, 55 (2007) 391-402.
- [31] A. Ranganathan, Pliss: Detecting and labeling places using online change-point detection, in: Proc. Robotics: Science and Systems (RSS), Zaragoza, 2010.
- [32] R. Luo, S. Piao, H. Min, Simultaneous Place and Object Recognition with Mobile Robot Using Pose Encoded Contextual Information, in: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), Shanghai, 2011, pp. 2792-2797.
- [33] J. Martinez-Gomez, B. Caputo, Towards semi-supervised learning of semantic spatial concepts, in: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), Shanghai, 2011, pp. 1936-1943.
- [34] U. Brefeld, T. Scheffer, Semi-supervised learning for structured output variables, in: Proc. of the 23rd Int. Conf. on Machine learning, ACM, Pittsburgh, Pennsylvania, 2006, pp. 145-152.
- [35] J. Verbeek, B. Triggs, Scene Segmentation with CRFs Learned from Partially Labeled Images, in: J.C. Platt, D. Koller, Y. Singer, S. Roweis (Eds.) Proc. Advances in Neural Information Processing Systems, MIT Press, Cambridge, MA, 2008, pp. 1553-1560.
- [36] F. Jiao, S. Wang, C.-H. Lee, R. Greiner, D. Schuurmans, Semi-supervised conditional random fields for improved sequence segmentation and labeling, in: Proc. of the 21st Int. Conf. on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Sydney, Australia, 2006, pp. 209-216.
- [37] B.T.C.G.D. Roller, Max-margin Markov networks, in: Proc. Advances in Neural Information Processing Systems, MIT Press, 2004, pp. 25.
- [38] C.-H. Lee, R. Greiner, M. Schmidt, Support Vector Random Fields for Spatial Classification, in: A. Jorge, L. Torgo, P. Brazdil, R. Camacho, J. Gama (Eds.) Knowledge Discovery in Databases: PKDD 2005, Springer Berlin Heidelberg, 2005, pp. 121-132.
- [39] J. Lafferty, X. Zhu, Y. Liu, Kernel conditional random fields: representation and clique selection, in: Proceedings of the twenty-first international conference on Machine learning, ACM, Banff, Alberta, Canada, 2004, pp. 64.
- [40] H. Xue, Q. Yang, S. Chen, SVM: Support Vector Machines, in: X. Wu, V. Kumar (Eds.) The top ten algorithms in data mining, CRC Press, Boca Raton, 2010, pp. 37-47.
- [41] C.W. Hsu, C.C. Chang, C.J. Lin, A practical guide to support vector classification, in, Department of Computer Science, National Taiwan University, Taipei, Tech. Rep. , 2003.
- [42] C. Chang, C. Lin, LIBSVM: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology, 2 (2011) 1-27.
- [43] D. Koller, N. Friedman, L. Getoor, B. Taskar, Graphical Models in a Nutshell, in: L. Getoor, B. Taskar (Eds.) Introduction to statistical relational learning, The MIT Press, 2007, pp. 13.
- [44] M. Schmidt, K. Murphy, G. Fung, R. Rosales, Structure learning in random fields for heart motion abnormality detection, in: Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), Anchorage, 2008.
- [45] S.V.N. Vishwanathan, N.N. Schraudolph, M.W. Schmidt, K.P. Murphy, Accelerated training of conditional random fields with stochastic gradient methods, in: Proc. Int. Conf. on Machine Learning (ICML), ACM, Pittsburgh, 2006, pp. 969-976.
- [46] M. Louw, Department of Electrical Engineering, University of Cape Town, 2009.
- [47] H. Choset, J. Burdick, Sensor based planning. I. The generalized Voronoi graph, in: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), Nagoya, 1995, pp. 1649-1655.
- [48] J.O. Wallgrun, Hierarchical voronoi graphs: spatial representation and reasoning for mobile robots, Springer, Heidelberg ; London, 2010.

- [49] T. Tao, S. Tully, G. Kantor, H. Choset, Incremental Construction of the Saturated-GVG for Multi-Hypothesis Topological SLAM, in: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), Shanghai, 2011, pp. 3072-3077.
- [50] Z. Zhou, M. Li, Semi-supervised learning by disagreement, Knowledge and Information Systems, 24 (2010) 415-439.
- [51] A. Blum, T. Mitchell, Combining labeled and unlabeled data with co-training, in: Proc. Int. Conf. on Learning Theory (COLT), ACM, Madison, 1998, pp. 92-100.
- [52] S. Kiritchenko, S. Matwin, Email classification with co-training, in: Proc. Conf. Centre for Advanced Studies on Collaborative Research, IBM Press, Toronto, 2001.
- [53] K. Bellare, A. McCallum, Learning extractors from unlabeled text using relevant databases, in: Proc. of Information Integration on the Web (IIWeb) workshop, Vancouver, 2007.
- [54] M.M. Ullah, F. Orabona, B. Caputo, You live, you learn, you forget: Continuous learning of visual places with a forgetting mechanism, in: Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), St. Louis, 2009, pp. 3154-3161.
- [55] E.A. Topp, H.I. Christensen, Detecting Region Transitions for Human-Augmented Mapping, IEEE Transactions on Robotics, 26 (2010) 715-720.
- [56] B. Hjorth, EEG analysis based on time domain properties, Electroencephalography and Clinical Neurophysiology, 29 (1970) 306-310.
- [57] S. Bockman, Generalizing the formula for areas of polygons to moments, American Mathematical Monthly, 96 (1989) 131-132.
- [58] P. Bourke, Calculating the area and centroid of a polygon, in, 1988.
- [59] B. Jahne, Digital image processing, 6th ed., Springer, Berlin ; New York, 2005.
- [60] Ó.M. Mozos, Semantic labeling of places with mobile robots, Springer, Berlin, 2010.
- [61] M.R. Spiegel, L.J. Stephens, ebrary Inc., Schaum's outline of theory and problems of statistics, 4th ed., McGraw Hill, New York, 2008.
- [62] B.S. Chissom, Interpretation of the Kurtosis Statistic, The American Statistician, 24 (1970) 19-22.
- [63] B.H. Junker, F. Schreiber, Analysis of biological networks, John Wiley & Sons, New York, 2008.
- [64] M.E.J. Newman, The mathematics of networks, in: L.E. Blume, S.N. Durlauf (Eds.) The New Palgrave Encyclopedia of Economics, Palgrave Macmillan, Basingstoke, 2008.
- [65] E. Estrada, J.A. Rodriguez-Velazquez, Subgraph centrality in complex networks, Physical Review E, 71 (2005).