Robotics: Science and Systems 2008
Zurich, CH, June 25-28, 2008

Laser and Vision Based Outdoor Object Mapping

Bertrand Douillard
ARC Centre of Excellence
for Autonomous Systems
Australian Centre for Field Robotics
University of Sydney, Australia
Email: b.douillard@cas.edu.au

Dieter Fox
Dept. of Computer Science & Engineering
University of Washington
Seattle, WA, USA
Email: fox@cs.washington.edu

Fabio Ramos
ARC Centre of Excellence
for Autonomous Systems
Australian Centre for Field Robotics
University of Sydney, Australia
Email: f.ramos@cas.edu.au

***Abstract*— Generating rich representations of environments can significantly improve the autonomy of mobile robotics. In this paper we introduce a novel approach to building object-type maps of outdoor environments. Our approach uses conditional random fields (CRF) to jointly classify laser returns in a 2D scan map into seven object types (car, wall, tree trunk, foliage, person, grass, and other). The spatial connectivity of the CRF model is determined via Delaunay triangulation of the laser map. Our model incorporates laser shape features, visual appearance features, structural information extracted from clusters of laser returns, and visual object detectors trained on image data sets available on the internet. The parameters of the CRF are trained from partially labeled laser and camera data collected by a car moving through an urban environment. Our approach achieves 91% accuracy in classifying objects observed along a 3 kilometer trajectory.**

I. Introduction

Generating rich representations of environments can bring another level of autonomy to mobile robotics. Over the last decade, much of the research in map building has focused on the simultaneous localization and mapping (SLAM) problem, *i.e.*, the problem of estimating the joint posterior distribution over the robot's location and the map of the environment. Research in this topic has produced various techniques that are able to build spatially consistent maps of large scale, cyclic environments [22].

More recently, several research groups extended SLAM approaches to generate maps that describe environments in terms of object types and places. Such representations can be extremely valuable, since they enable robots to perform high-level reasoning about their environments and the objects therein. For instance, in search and rescue tasks, a mobile robot that can reason about objects such as doors, and places such as rooms is able to coordinate with first responders in a much more natural way, being able to accept commands such as "Search the room behind the third door on the right of this hallway", and conveying information such as "There is a wounded person behind the desk in that room" [11]. As another example, consider autonomous vehicles navigating in urban areas. While the recent success of the DARPA Urban Challenge [5] demonstrates that it is possible to develop autonomous vehicles that can navigate safely in constrained settings, successful operation in more realistic, populated urban areas requires the ability to distinguish between objects such as cars, people, buildings, trees, and traffic lights.

In this paper we introduce a novel approach to building object type maps of outdoor environments. Our approach applies standard scan matching techniques to align 2D laser scans collected by a vehicle driving through urban environments. We use conditional random fields (CRF) to classify each laser return into the seven object types: car, wall, tree trunk, foliage, person, grass, and other. In contrast to previous work on outdoor object mapping [18], our model performs joint classification of the laser returns. This is done by connecting the nodes of the CRF based on a Delaunay triangulation of the laser data. An important aspect of CRFs is their ability to incorporate many features with arbitrary dependencies. Our model takes advantage of this ability by incorporating large sets of laser shape features and visual appearance features extracted from camera data. The parameters of our models are learned from partially labeled laser and camera data. We show that classification can be further improved by explicitly modeling within a CRF the information contained in the arrangement of clusters of returns. We also present results on the incorporation of visual object detectors trained on publicly available image data sets such as the LabelMe set [2].

We evaluate our technique on laser and camera data collected by a vehicle navigating through an urban environment. Tested using ten-fold cross validation, objects observed along a 3 kilometer long trajectory are identified with an accuracy of 91%.

This paper is organized as follows. Related work is discussed first, in Section II. In Section III, we introduce the probabilistic models underlying our mapping approach, followed by a description of features used for classification. Experimental results are presented in Section V. Finally, we conclude in Section VI.

II. Related Work

Object recognition is a long-standing problem in robotics and computer vision. Most of the approaches in computer vision aim at recognizing objects from single images. Classifiers are trained on labeled data and used to either classify images as containing or not an instance of the object, or to segment the object in the image. Examples are [8, 23, 25]. In robotics, the problem is different. Recognition can be performed in a sequence of images, in many cases combined with other sensor

modalities. Alternatively, object recognition can be required on a full map, as addressed in this paper.

Within the robotics community, recent developments have created representations of the environment integrating more than one sensor modality. In [17], a 3D laser scanner and loop closure detection based on photometric information are brought together into the Simultaneous Localization and Mapping (SLAM) framework. This approach does not generate a semantic representation of the environment which can be obtained from the same multi-modal data using the approach proposed here.

In [20], a robust landmark representation is created by probabilistic compression of high-dimensional vectors containing laser and camera information. This representation is used in a SLAM system and updated on-line when a landmark is re-observed. However, it does not reason about landmark classes and therefore does not support the higher-level object detection described in this work.

Object recognition based on laser and video data has been demonstrated in [15]. Using a sum rule, this approach combines the outputs of two classifiers, each of them being assigned to the processing of one type of data. More recently, Posner and colleagues combine 3D laser range data with camera information to classify surface types such as brick, concrete, grass, or pavement in outdoor environments [18, 19]. The authors classify each laser scan return independently which can disregard important neighborhood information. As other researchers have shown, classification results can be improved by jointly classifying laser beams using techniques such as associative Markov networks [24] or conditional random fields [7].

In [3], a Markov Random Field is used to segment objects from 3D laser scans. The model is trained discriminatively using a max-margin objective function. The features used were simple geometric features capturing plane properties of groups of points. The authors considered four classes: ground, building, tree and shrubbery. Friedman and colleagues introduced Voronoi Random Fields, which generate semantic place maps of indoor environments by labeling the points on a Voronoi graph of a laser map using conditional random fields [10].

The key contribution of this paper is a methodology to build maps of objects in which accurate classification is achieved by exploiting the ability of CRFs to represent spatial correlations and to model the structural information contained in clusters of laser returns.

III. Mapping In Conditional Random Fields

To augment geometric maps with semantic information, we have developed three approaches corresponding to three different models. All these models are based on the framework provided by conditional random fields. Before describing how these models are built from laser and camera data, we provide background on learning and inference in conditional random fields.

A. Conditional Random Fields

Conditional random fields (CRF) are undirected graphical models developed for labeling sequence data [12]. CRFs directly model $p(\mathbf{x}|\mathbf{z})$, the *conditional* distribution over the hidden variables $\mathbf{x}$ given observations $\mathbf{z}$. In our framework, $\mathbf{x}$ is the set of object types to be estimated, a hidden state being instantiated for each laser return. The observations $\mathbf{z}$ correspond to shape and appearance features extracted from laser and vision data, respectively. A CRF can be formulated as follows:

$$p(\mathbf{x}|\mathbf{z}) = \frac{1}{\mathbf{Z}} \exp\left(w_A \sum_i A(x_i, \mathbf{z}) + w_I \sum_e I(x_e, x_{e'}, \mathbf{z}) \right) \quad (1)$$

Here, the term $1/\mathbf{Z}$ is a normalization factor. The functions A and I are the association and interaction potentials, respectively. In our framework, an association potential A is instantiated as a logitboost classifier [9] and estimates the object type of node x_i using the set of observations $\mathbf{z}$ but does not take into account information contained in the structure of the neighborhood. An interaction potential I is a function associated to each edge e of the CRF graph, where x_e and $x_{e'}$ are the nodes connected by edge e. Intuitively, interaction potentials measure the compatibility between neighboring nodes and act as smoothers by correlating the estimation across the network.

In our system, the first step of the CRF training is learning the logitboost classifier A which is performed as in [7]. The second step of the learning consists in finding optimal values for the set of weights w_A and w_I based on a labeled data set. Depending on the connectivity structure of the network to be trained, the system uses exact or approximate learning techniques. For non-cyclic networks, the systems uses a Maximum Likelihood approach since inference can be performed exactly. For networks containing cycles, the system uses the approximate version of this technique which is known as Maximum Pseudo-Likelihood learning [4].

Since the values of the local potential function A are obtained as the output of a logitboost classifier, our approach for training can be seen as an extension of boosting to structured classification tasks. As a result, this approach is very flexible and powerful. It not only learns the weights of the potentials, but also selects the subset of dimensions in the observation vectors $\mathbf{z}$ which are useful for classification [10, 13].

In this work, the maximum pseudo-likelihood learning is slightly extended in such a way that the labels of neighbor nodes are not required, allowing training to be performed on partially labeled data. This is achieved by optimizing the pseudo-likelihood written as:

$$\begin{aligned} pl(\mathbf{x}|\mathbf{z}) &= \prod_{i=1}^{N} p(x_i|\mathrm{MB(x_i)}, \mathrm{z}) \propto \prod_{\mathrm{i=1}}^{\mathrm{N}} \exp(\mathrm{w_A A(x_i, \mathbf{z})}) \\ &\quad \prod_{k \in \mathrm{MB(x_i)}} \exp(w_I I(x_i, x_k, \mathbf{z}) + w_A A(x_k, \mathbf{z})) \end{aligned}$$

where the last equation is obtained by breaking the exponential in Eq. 1 into two terms (the full derivation is not given

here due to space constraints). N refers to the number of nodes in the network and $MB(x_i)$ is the Markov blanket of node x_i. The parameters to be adjusted to find the maximum value of the pseudo-likelihood are w_A and w_I. In this formulation, the usually required neighbor labels are replaced by the estimated distribution over the neighbor's label: $\exp(w_A A(x_k, \mathbf{z}))$. Via this formulation, the learning algorithm can use the unlabeled nodes in the neighborhood of each labeled node and be performed on partially labeled data.

Inference in CRFs estimates either the marginal distribution of each hidden variable $\mathbf{x}_i$ or the most likely configuration of all hidden variables $\mathbf{x}$ (*i.e.*, MAP estimation), based on their joint conditional probability (Eq. 1). We solve both tasks using belief propagation (BP) for non-cyclic networks. For cyclic networks, we use the approximate version of BP called Loopy Belief Propagation (loopy BP) [16].

B. From Laser Scans to Conditional Random Field

The input to our system is a collection of spatially aligned laser scans obtained by performing scan matching with the iterative closest point (ICP) algorithm [27] [1]. In this section, we present three types of CRFs which will be compared in order to better understand how to model the spatial correlations in a semantic map. We show how the three different models can be instantiated from aligned laser data and indicate which learning and inference techniques are used in each case. For these three networks, the hidden state for each node ranges over the seven object types: car, trunk, foliage, people, wall, grass, and other (any other object type).

1) Delaunay CRF: In this first type of network, each laser return is instantiated as one node in the CRF. The connections between the nodes are found using the Delaunay triangulation procedure [6] which efficiently finds a triangulation with non-overlapping edges. The system then removes links which are longer than a pre-defined threshold (50 cm in our application) since distant nodes are not likely to be strongly correlated. The resulting network is displayed as a set of blue edges in Fig. 2.

Since a Delaunay CRF contains cycles, training and inference are performed with maximum pseudo-likelihood and loopy BP, respectively.

2) Delaunay CRF with link selection: Generally speaking, structured classification as performed by CRFs is expected to improve on local classification since independence is not assumed, *i.e.*, neighborhood information is modelled through interaction potentials. However, as illustrated by the experimental results, the first type of CRF previously described does not improve on local classification. A too coarse modelling of the spatial correlations is responsible for this result. The term $\exp(w_I I(x_i, x_k, \mathbf{z}))$ of Eq. 1 is learnt in this first type of network as a constant matrix instantiated at each of the links. This gives the network a smoothing effect on top of the local classification. Since all the links are represented with the same matrix, only one type of node-to-node relationship is encoded, for example: neighbor nodes should have the same label. Such links are appropriate in very structured parts of the environment but may over-smooth in areas where the density of objects increases.

In order to model more than one type of node-to-node relationships, the network is augmented with an additional node T for every pair of nodes $\{x_i, x_j\}$ as displayed in Fig. 1. The state of this node specifies which type of link is instantiated. For this second type of network, we consider two types of links encoding the following node-to-node relationships: (1) neighbor nodes have the same label, (2) neighbor nodes have a different label. Node T receives an observation S which is the output of a logitboost classifier learned to estimate whether node x_i and x_j are similar based on their respective local observation z_i and z_j. The observation S is a direct observation of the state of node T.

Since this second type of network contains loops, training and inference are also performed with maximum pseudo-likelihood and loopy BP, respectively.

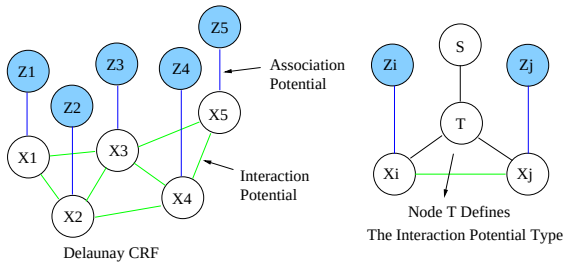

Fig. 1. Representation of the additional infrastructure required in a Delaunay CRF to perform link selection.

3) Tree based CRF: The previous two types of network contain cycles, which implies the use of approximate learning and inference algorithms. We now present a third type of network which is cycle free and does not require the use of approximate techniques. To design non-cyclic networks we start from the following observation: laser returns in a scan map are naturally organized into clusters. These clusters can be identified by analysising the connectivity of the Delaunay graph and finding its disconnected sub-components. Disconnected components appear when removing longer links of the original triangulation. In Fig. 2, the extracted clusters are indicated by green rectangles.

Once the clusters are identified, the nodes of a particular cluster are connected by a tree of depth one. A root node is instantiated for each cluster and each node in the cluster becomes a leaf node. The trees associated to the clusters in Fig. 2 are represented by green volumes. A tree-based CRF does not encode node-to-node smoothing but rather performs smoothing based on the identified clusters of laser returns.

The root node does not have an explicit state. It allows the instantiation of a network which does not contains cycles enabling learning and inference to be performed exactly. With this third type of network, the system uses a maximum likelihood approach for learning and belief propagation for

[1] In spatially more complex data sets containing loops, consistently aligned scans can be generated using various existing SLAM techniques [22]

inference. The possibility of using exact learning and inference is a strong advantage compared to the absence of theoretical results in terms of convergence of maximum pseudo-likelihood learning and loopy belief propagation.

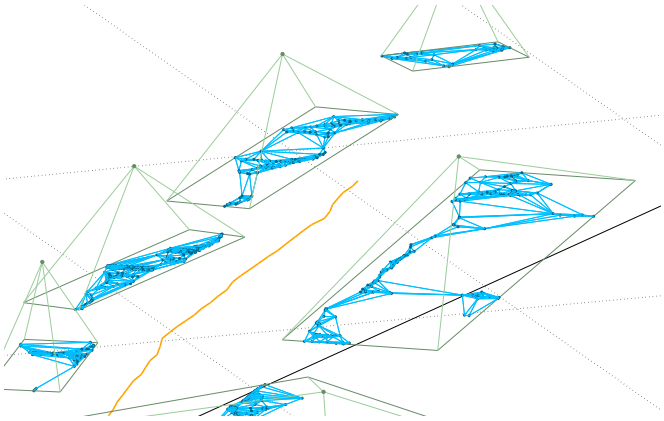

Fig. 2. Representation of a Tree based CRF in one region of a graph generated from data. The trajectory of the vehicle is displayed in orange. Laser returns are instantiated as nodes in the network and connected using the Delaunay triangulation. Nodes and edges are plotted in dark and light blue, respectively. Identified clusters are indicated by the green rectangles while root nodes are plotted in green. Root nodes are connected to all nodes in the cluster but for clarity this is represented by a rectangle enclosing the cluster.

IV. Features For Object Mapping

As formulated in Eq. 1, the computation of the posterior probability requires the set of observations $\mathbf{z}$. In this work, $\mathbf{z}$ consists of high-dimensional feature vectors $\mathbf{f}$ computed for each scan return. $\mathbf{f}$ results from the concatenation of three types of features, geometric features, visual features and features extracted from on-line datasets:

$$\mathbf{f} = [\mathbf{f}_{\text{geo}}, \mathbf{f}_{\text{visu}}, \mathbf{f}_{\text{www}}], \tag{2}$$

Geometric and visual features are first described. We then show how on-line labeled datasets freely available on the internet can provide additional binary features.

A. Geometric Features

Geometric features capture geometric properties of the objects in the laser returns. The feature vector computed for one scan return has a dimensionality of 231 and results from the concatenation of 38 different multi dimensional features. Due to limited space we only present a subset of these features below:

$$\mathbf{f}_{\text{geo}}(i, z_A) = [\mathbf{f}_{\text{dist}}, \mathbf{f}_{\text{angle}}, \mathbf{f}_{\text{oor}}, \mathbf{f}_{\text{cluster}}, \ldots], \tag{3}$$

where i indexes one of the returns in scan z_A.

$\mathbf{f}_{\text{dist}}$ or distance features are computed for each return $z_{A,i}$ in scan A as its distance to other points in scan A:

$$\mathbf{f}_{\text{dist}}(i, k, z_A) = \|z_{A,i} - z_{A,i+k}\|, \tag{4}$$

where k varies from -10 to $+10$.

$\mathbf{f}_{\text{angle}}$ or angle features are computed as angles formed by various configurations of neighbor returns:

$$\mathbf{f}_{\text{angle}}(i, k, l, z_A) = \|\angle(\overline{z_{A,i-k}z_{A,i}}, \overline{z_{A,i}z_{A,i+l}})\|. \tag{5}$$

where k and l vary from -10 to $+10$. These two first types of features provide information about the local shape of the scan around return i.

$\mathbf{f}_{\text{oor}}$ or out of range features count the number of "out of range" beams between pairs of successive returns. These features allow the representation of open areas between valid beams of the laser scan.

$\mathbf{f}_{\text{cluster}}$ consists of various features computed to describe a cluster of laser returns. Cluster of returns within a single scan are extracted based on a simple distance criteria and characterized through the following quantities: geodesic length of the cluster, length of its two principal components, error generated by the fit of a spline to the cluster points. Note that two returns in the same cluster have the same $\mathbf{f}_{\text{cluster}}$ vector. The aim of the $\mathbf{f}_{\text{cluster}}$ features is to capture the organization of objects at the scale of one laser scan.

B. Visual Features

In addition to laser range scans, our system incorporates visual appearance by projecting the laser returns into camera images collected by a calibrated camera mounted on the vehicle, similar approach to [7, 19].

The CRF learned with a logitboost based algorithm can not only integrate geometric information but also any other type of data and, in particular, visual features extracted from monocular color images. As a consequence, the system extracts features in a region of interest (ROI) defined around the projection of each return into the corresponding image. The parameters required to carry out the projection are defined through the camera laser calibration procedure developed in [26]. The size of the ROI is changed depending on the range of the return. This provides a mechanism to deal with changes in scales across images. It was verified that the use of a size varying ROI improves classification accuracy by 4%.

The visual feature vector associated to each return has a dimensionality of 1239 and results from the concatenation of 51 multi-dimensional features computed in the ROI. Due to limited space, we only describe the most important of these features:

$$\mathbf{f}_{\text{visu}}(i) = [\mathbf{f}_{\text{pyr}}, \mathbf{f}_{\text{rgb}}, \mathbf{f}_{\text{hsv}}, \mathbf{f}_{\text{haar}}, \mathbf{f}_{\text{edges}}, \mathbf{f}_{\text{lines}}, \mathbf{f}_{\text{sift}}, \ldots], \tag{6}$$

where index i refers to the ROI associated to return i.

$\mathbf{f}_{\text{pyr}}$ returns texture information encoded as a vector containing the steerable pyramid [21] coefficients of ROI i as well as the minimum and the maximum of these coefficients. These extrema are useful to classify cars which from most point of views have a relatively low texture maxima due to their smooth surface.

$\mathbf{f}_{\text{rgb}}$ and $\mathbf{f}_{\text{hsv}}$ return a 3D histogram of the RGB and HSV data in ROI i.

$\mathbf{f}_{\text{haar}}$ returns Haar features of ROI i computed using the integral image approach proposed in [25].

$\mathbf{f}_{\text{edges}}$ uses a Canny edge detector to extract the number of pixels within ROI i recognized as belonging to an edge.

$\mathbf{f}_{\text{lines}}$ processes the whole image with the line detector [1] and extracts the number of lines intersecting ROI i as well as the maximum length of this subset of lines.

$\mathbf{f}_{\text{sift}}$ counts the number of Sift features [14] found in ROI i.

C. Using On-line Datasets

In our datasets, some of the classes such as the class people have no more than one hundred training samples. This can be detrimental to the accuracy of the classifier. To compensate for the lack of training data, we have used binary features computed with classifiers trained on on-line datasets. Across the web, large labeled datasets such as the LabelMe dataset [2] can be used to learn binary classifiers on large amount of training data. We used the LabelMe data to train binary object detectors for each of the ten classes: car, tree trunk, foliage, pedestrian, building, grass, road, pole, fence and road; and applied these detectors to our data to generate an additional binary feature vector $\mathbf{f}_{\text{www}}$ of dimensionality 10.

In addition to an algorithm which can be trained with partially labeled data, the use of on-line labeled data sets decrease the labelling effort. The results reported in Sec. V-B.4 with respect to the $\mathbf{f}_{\text{www}}$ features show the right trend while no significant improvement has been obtained yet. This part of the work is preliminary and aims at introducing the idea of generating additional features as output of classifiers trained on on-line datasets. We believe that understanding the requirements for features to be portable from standard datasets to a given robotics application is crucial for large-scale autonomy and this paper opens up this direction of research.

V. Experimental Results

A. Experimental Setup

Experiments were performed using outdoor data collected with a modified car traveling at 0 to 40 km/h along a 3km long trajectory. The car drove in a university campus which has structured areas with buildings, walls and cars, and unstructured areas with bush, trees and lawn fields. The overall dataset contains 4500 images representing 20 minutes of logging. Laser and vision data was acquired at a frequency of 4Hz. The laser sensor used belongs to the family of SICK devices and the camera was a high-resolution wide angle Hanvision camera.

The evaluation of the classifier was performed on a ten-fold cross validation setup which involves training each classifier on nine tenth of the trajectory and testing it on the remaining one tenth. These two operations are repeated ten times by changing the testing and training sets accordingly. The results presented below are averaged over the cross validation runs.

Each set of scans was converted into a probabilistic network as described in Sec. III-B. Training and testing sets were partly hand labeled to provide labels to the learning algorithm and a ground truth to evaluate classification accuracy.

The properties of the training and testing sets averaged over the ten tests are provided in Table I.

	Length vehicle trajectory	# scans total labeled	# nodes total labeled
Training set	2.6 km	3843 72	67612 5168
Testing set	290 m	427 8	7511 574

TABLE I

Properties of the Training and Testing sets

B. Classification Performance

This section presents the classification performances obtained with the three models presented in Sec. III-B. Results for local classification are first presented in order to provide a baseline for comparison.

1) Local Classification: A seven-class logitboost classifier is learned and instantiated at each node of the network as the association potential A (Eq. 1). Local classification, *i.e.*, classification which does not take neighborhood information into account is performed with the confusion matrix presented in Table II. This confusion matrix displays a strong diagonal which corresponds to an accuracy of 90.4%. A compact characterization of the confusion matrix is given by precision and recall values. These are presented in Table III. Averaged over the seven classes, the classifier achieves a precision of 89.0% and a recall of 98.1%.

Truth \ Inferred	Car	Trunk	Foliage	People	Wall	Grass	Other
Car	1967	1	7	10	3	0	48
Trunk	4	165	18	0	4	0	11
Foliage	25	18	1451	0	24	0	71
People	6	2	2	145	0	0	6
Wall	6	6	21	0	513	1	39
Grass	0	0	1	1	1	146	4
Other	54	5	123	3	24	0	811

TABLE II

Local Classification: Confusion Matrix

In %	Car	Trunk	Foliage	People	Wall	Grass	Other
Precision	96.6	81.7	91.3	90.1	87.5	95.4	79.5
Recall	97.9	99.3	96.4	99.7	98.5	99.9	95.4

TABLE III

Local Classification: Precision and Recall

2) Delaunay CRF classification:

a) CRF without built-in link selection: the accuracy achieved by this first type of network is 90.3% providing no improvements on local classification. As developed in Sec. III-B.2, the modelling of the spatial correlation is too coarse since it contains only one type of link which cannot accurately model the relationships between neighbor nodes. As a consequence, the links end up representing the predominant relationship in the data. In our application the predominant neighborhood relationships are of the type "neighbor nodes possessing the same label". The resulting learned links enforce this "same-to-same" relationship across the network leading to over smooth estimates and explaining why this class of networks fails to improve on local classification. To verify that

a better modelling of the CRF links improves the classification performance, we now presents results generated by the second proposed type of CRF, characterized by a built-in link selection process.

b) CRF with built-in link selection: the accuracy achieved by this second type of network is 91.4% which corresponds to 1.0% improvement in accuracy. Since the local accuracy is already high, the improvement brought by the network may be better appreciated when expressed as a reduction of the error rate of 10.4%. This result validates the claim that a set of link types encoding a variety of node-to-node relationships is required to exploit the spatial correlations in the laser map.

3) Tree based CRF classification: The two types of networks evaluated in the previous section contain cycles and require the use of approximate learning and inference techniques. The tree based CRFs presented in Sec. III-B.3 avoid these issues by allowing the use of exact learning and inference procedures.

This third type of network achieves an accuracy of 91.1% which is slightly below the accuracy given by a CRF with link selection while still improving on the CRF without link selection. However, the major improvement brought by this third type of network is in terms of computational time. Since the network has the complexity of a tree of depth one, learning and inference, in addition to being exact, can be implemented very efficiently. As displayed in Table IV, a tree based CRF is 80% faster at training and 90% faster at testing than a Delaunay CRF. Since both network types use as their association potential the seven classes logitboost classifier, they require the same features extracted from a scan and its associated image in 1.2 secs on average. As shown in Table I, the test set contains 7511 nodes on average which suggests that the tree based CRF approach is in its current state is very close to real time, feature extraction being the main bottleneck.

	Feature Extraction (per scan)	Learning (training set)	Inference (test set)
Delaunay CRF (with link selection)	1.2 secs	6.7 mins	1.5 mins
Tree based CRF	1.2 secs	1.5 mins	10.0 secs

TABLE IV
COMPUTATION TIMES

4) Using on-line data sets for training: Based on the LabelMe set, 10 binary object detectors are trained using the logitboost algorithm. The 10 classes considered are: car, tree trunk, foliage, pedestrian, building, grass, road, pole, fence and road. Since the LabelMe dataset contains vision data only, these binary classifiers are vision based detectors and, in order to use their output as additional features, we run them on the ROIs selected in each image of our urban dataset (the selection of these ROIs is performed as described in Sec. IV-B).

Within our urban dataset as well as within the LabelMe dataset, the size of the selected ROIs are not constant which requires designing the various vision features in such a way that the dimensionality of the vector $\mathbf{f}_{\mathrm{visu}}$ is independent of the ROI size. Our approach consist in using features which are distributions (e.g. an histogram with a fixed number of bins) and whose dimensionality is constant (e.g. equal to the number of bins in the histogram). A larger ROI leads to a better sampled distribution (e.g. a larger number of samples in the histogram) and the actual feature dimensionality remains invariant.

The use of these additional $\mathbf{f}_{\mathrm{www}}$ features slightly improves the local classification accuracy from 90.4% to 90.6%. We believe that there is no further increase in accuracy due to the fact that the lighting conditions in the two datasets differ significantly (our urban dataset contains images which are on average much darker than the ones in the LabelMe dataset). In the context of preliminary investigations, these results are encouraging and future tests will involve datasets with more similar lighting conditions.

C. Map of Objects

This section presents a visualization of the mapping results. It follows the lay out of Figure 3 in which the vehicle was travelling from right to left.

At the location of the first inset, the vehicle was going up a straight road with a fence on its left and right, and, from the foreground to the background, another fence, a car, a parking meter and bush. All these objects were correctly classified with the fences and the parking meter identified as other.

In the second inset, the vehicle was coming into a curve facing a parking lot and bush on the side of the road. Four returns misclassified as other can be seen in the background of the image. The class other regularly generated false positives which is possibly caused by the dominating number of training samples in this class. Various ways of re-weighting the training samples or balancing the training set were tried without significant improvements.

While reaching the third inset, a car driving in the opposite direction came into the field of view of our vehicle's sensors. The trace let by this car in the map appears in the magnified inset as a set of blue dots along side our vehicle's trajectory. Dynamic objects are not explicitly considered within this work. They are assumed to move at a speed which does not prevent ICP from performing accurate registration. In the campus areas where the data was obtained, this assumption has proven to be valid. In spite of a few miss-classifications in the bush on the left side of the road, the pedestrians on the side walk are correctly identified and the wall of the building is recognised.

Entering the fourth inset, our vehicle was facing a second car, scene which appears in the map as a blue trace intersecting our vehicle's trajectory. Apart from one miss-classified return on one of the pedestrians, and one miss-classified return on the tree in the right of the image, the inferred labels are accurate. Note that the first right return is correctly classified illustrating the accuracy of the model at the border between objects.

VI. CONCLUSIONS

This paper introduces a novel approach for object mapping in outdoor environments. Our technique applies conditional

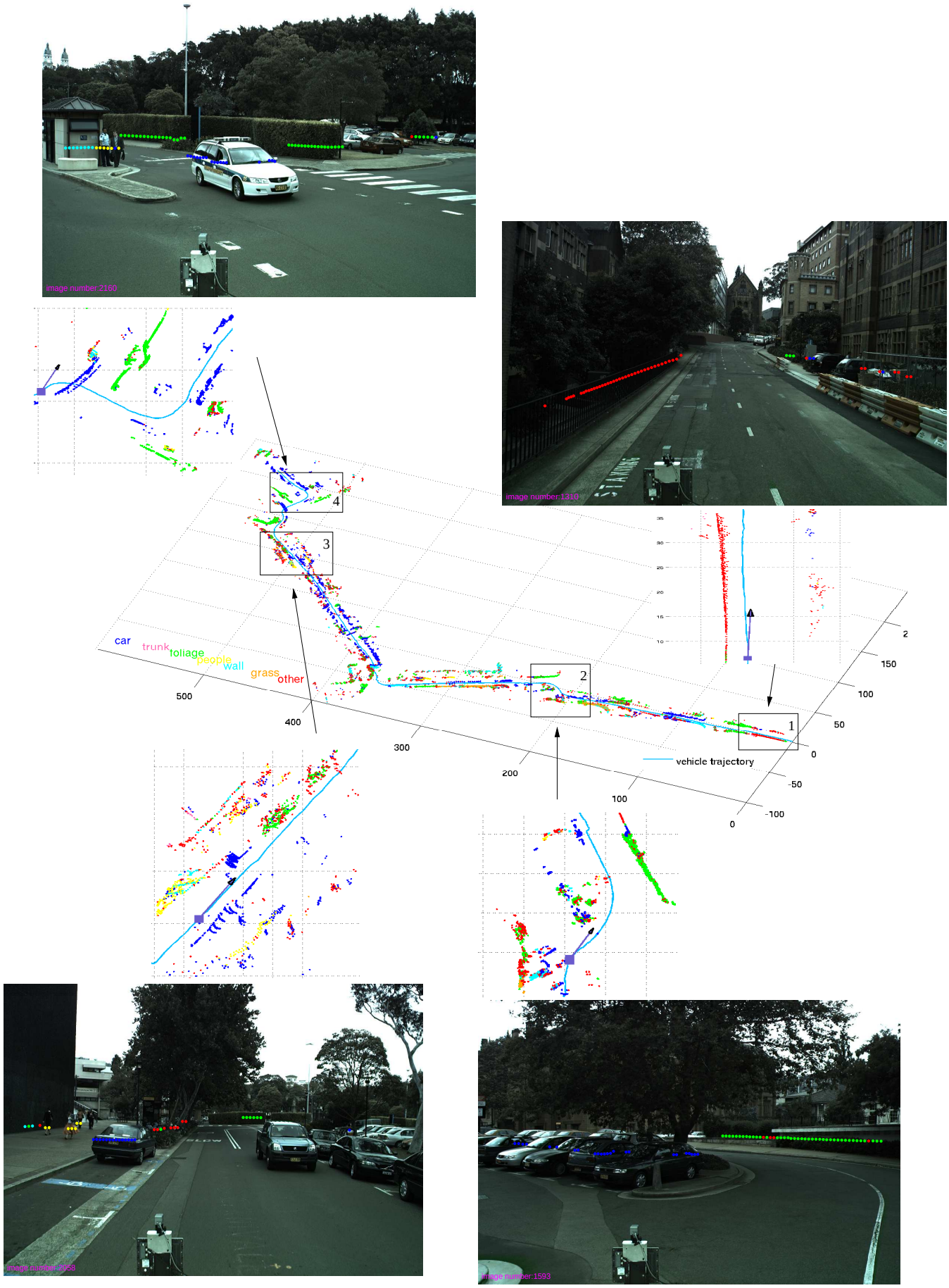

Fig. 3. Visualization of 750 meters long portion of the estimated map of objects with total length of 3km. The map was generated using the tree based CRF model. The legend is indicated in the bottom left part of the 2D plane. The color of the vehicle's trajectory is specified in the bottom right part of the same plane. The coordinate in the plane of the map are in meters. Each inset is magnified and associated to an image displayed with the inferred labels projected back onto the original returns. The location of the vehicle is shown in each magnified patch with a square and its orientation indicated by the arrow attached to it. The laser scanner mounted on the vehicle can be seen in the bottom part of each image.

random fields to label individual points in a 2D laser map annotated with camera data. We take advantage of CRFs' ability to handle dependent features by incorporating large sets of shape and appearance information extracted from laser scans and cameras. Spatial dependencies are modeled by connecting nodes in the CRF based on a Delaunay triangulation of the laser data. Label smoothing on the object level is achieved by three different graph structures based on a spatial segmentation of the laser data. Our approach learns both feature functions and model parameters using a combination of maximum likelihood and logitboost training on partially labeled data.

Experiments conducted on data collected along a 3km trajectory through an urban area indicate that our system achieves very good classification rates for object types such as car, trunk, foliage, people, wall, grass, and other. The approach achieves a reduction of the classification error of 10.4% with respect to a local approach solely integrating standard shape and appearance features. We also show how on-line datasets can be integrated by incorporating object detectors as additional features.

These results are extremely encouraging and the following aspects are promising directions for future work. The accuracy of our current system suffers from lack of training data, especially for more sparsely observed objects such as tree trunks and people. While this can be overcome by collecting and labeling more data, our experiments indicate that leveraging the large number of labeled (and unlabeled) vision data resources on the web is a more scalable technique. The CRFs underlying our system are able to incorporate many externally learned classifiers, and an interesting question is how to best combine these classifiers with the shape information provided by the laser data.

While the current mapping system is designed to run off-line, the efficiency of feature extraction and inference makes it possible to generate object maps on the fly, additionally labeling objects as moving or not.

Finally, the most important limitation of our current system is the reliance on 2D laser range data. However, we believe that our approach can also be applied to 3D laser data, which should greatly improve the accuracy and richness of the generated maps.

VII. ACKNOWLEDGMENTS

The authors would like to thank Roman Katz for numerous useful discussions, Juan Nieto, Jose Guivant, and Oliver Frank for helping with the dataset acquisition. This work is supported by the ARC Center of Excellence programme, the Australian Research Council (ARC), the New South Wales (NSW) State Government, the University of Sydney Visiting Collaborative Research Fellowship Scheme, and DARPA's ASSIST and CALO Programmes (contract numbers: NBCH-C-05-0137, SRI subcontract 27-000968).

REFERENCES

[1] Finding long, straight lines code. `http://www.cs.uiuc.edu/homes/dhoiem/`.

[2] Labelme data set. `http://labelme.csail.mit.edu/`.

[3] D. Anguelov, D. Koller, E. Parker, and S. Thrun. Detecting and modeling doors with mobile robots. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 2004.

[4] J. Besag. Statistical analysis of non-lattice data. *The Statistician*, 24, 1975.

[5] DARPA Urban Challenge. `http://www.darpa.mil/grandchallenge/index.asp`.

[6] M. De Berg, M. Van Kreveld, M. Overmars, and O. Schwarzkopf. Springer-Verlag, 2000. 2nd rev. ISBN: 3-540-65620-0.

[7] B. Douillard, D. Fox, and F. Ramos. A spatio-temporal probabilistic model for multi-sensor multi-class object recognition. In *Proc. of the International Symposium of Robotics Research (ISRR)*, 2007.

[8] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[9] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, 28(2), 2000.

[10] S. Friedman, D. Fox, and H. Pasula. Voronoi random fields: Extracting the topological structure of indoor environments via place labeling. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2007.

[11] V. Kumar, D. Rus, and S. Singh. Robot and sensor networks for first responders. *IEEE Pervasive Computing*, 3(4), 2004. Special Issue on Pervasive Computing for First Response.

[12] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of the International Conference on Machine Learning (ICML)*, 2001.

[13] L. Liao, T. Choudhury, D. Fox, and H. Kautz. Training conditional random fields using virtual evidence boosting. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2007.

[14] D. Lowe. Discriminative image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 2004.

[15] G. Monteiro, C. Premebida, P. Peixoto, and U. Nunes. Tracking and classification of dynamic obstacles using laser range finder and vision. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2006.

[16] K. Murphy, Y. Weiss, and M. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 1999.

[17] P. Newman, D. Cole, and K. Ho. Outdoor SLAM using visual appearance and laser ranging. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, Orlando, USA, 2006.

[18] I. Posner, D. Schroeter, and P. M. Newman. Describing composite urban workspaces. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 2007.

[19] I. Posner, D. Schroeter, and P. M. Newman. Using scene similarity for place labeling. In *Proc. of the International Symposium on Experimental Robotics (ISER)*, 2007.

[20] F. Ramos, J. Nieto, and H.F. Durrant-Whyte. Recognising and modelling landmarks to close loops in outdoor slam. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 2007.

[21] E. Simoncelli and W. Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *Proc. of the International Conference on Image Processing*, 1995.

[22] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, Cambridge, MA, September 2005. ISBN 0-262-20162-3.

[23] A. Torralba, K. Murphy, and W. Freeman. Contextual models for object detection using boosted random fields. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.

[24] R. Triebel, K. Kersting, and W. Burgard. Robust 3D scan point classification using associative Markov networks. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 2006.

[25] P. Viola and M. Jones. Robust real-time ob ject detection. In *International Journal of Computer Vision*, volume 57, page 2, 2004.

[26] Q. Zhang and R. Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japan, 2004.

[27] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, 1994.