A Spatio-Temporal Probabilistic Model for Multi-Sensor Object Recognition

Bertrand Douillard* Dieter Fox† Fabio Ramos*

* *ARC Centre of Excellence for Autonomous Systems*
Australian Centre for Field Robotics
University of Sydney
Sydney, NSW, Australia

† *Dept. of Computer Science & Engineering*
University of Washington
Seattle, WA, USA

***Abstract*— This paper presents a general framework for multi-sensor object recognition through a discriminative probabilistic approach modelling spatial and temporal correlations. The algorithm is developed in the context of Conditional Random Fields (CRFs) trained with virtual evidence boosting. The resulting system is able to integrate arbitrary sensor information and incorporate features extracted from the data. The spatial relationships captured by are further integrated into a smoothing algorithm to improve recognition over time. We demonstrate the benefits of modelling spatial and temporal relationships for the problem of detecting cars using laser and vision data in outdoor environments.**

I. INTRODUCTION

Reliable object recognition is an important step for enabling robots to reason and act in the real world. A high-level perception model able to integrate multiple sensors can significantly increase the capabilities of robots. Tasks such as obstacle avoidance, mapping, and tracking can all benefit from a fast and general object detector able to be trained to recognise specific objects of interest.

Although object recognition has been a major research topic in the computer vision community, direct application of the algorithms to robotics problems is not always feasible. There are three main reasons for this. First, robotics applications require real-time object recognition. Although real-time algorithms for face detection do exist [22], real-time recognition of general objects is still under development. Second, robots can be equipped with different types of sensors including ranging and visual. The integration of these sensors for object recognition can complement the visual information by providing additional geometric properties of observed objects. Multi-sensor fusion for object recognition is thus a desirable feature to be considered in robotics perception. Third, when navigating, robots observe the same objects from different locations and in different periods of time. This is conceptually different from most object recognition algorithms in computer vision where observations are considered independent. Algorithms able to integrate observations at different times and positions are expected to perform more robustly in complex outdoor environments with variable illumination and multi-scale observations.

We present an algorithm to address these issues in the context of Conditional Random Fields (CRFs). CRFs are discriminative models for classification of sequential (dependent) data, directly modelling the conditional probability $p(\mathbf{x}|\mathbf{z})$ of hidden states given observations [6]. CRFs have been applied with substantial success to recognition problems in robotics, including object mapping and semantic place labeling [9], [4]. By building on the recently developed Virtual Evidence Boosting (VEB) algorithm for CRF training [7], the model described here is able to automatically select features during the learning phase. Expert knowledge about the problem can be encoded as a selection of features capturing particular properties of the data such as geometry, colour and texture. Given a training set, the algorithm automatically selects and weighs each of these features according to their importance in discriminating the data.

The proposed framework uses CRFs as a unifying methodology to learn spatial and temporal relationships from observations obtained with a laser range-finder and a camera. The model is trained to recognise cars in an urban environment from a moving vehicle. In outdoor environments the problem of recognising objects is more difficult. Illumination can change significantly as the robot enters covered areas. Terrain irregularities can contribute to blurring and occlusions occur quite often. Our experiments reveal that temporal integration can significantly improve the accuracy of object detection. By observing the same objects at different points in time, detection becomes more robust and able to cope with the complexity of the environment. As the vehicle approaches an object, the scale of the object increases in both laser and camera data. The additional information can be integrated with past observations through temporal links included in the CRF model. Temporal features are pairwise relationships linking laser points at different times. The Iterative Closest Point algorithm (ICP) [24] is used to define which points are connected by temporal features.

This paper is organised as follows. After discussing related work in Section II, background on conditional random fields and virtual evidence boosting is provided in Section III. The application to spatio-temporal object recognition is described in Section IV, followed by the experimental evaluation. We conclude in Section VI.

II. RELATED WORK

CRFs have been applied to recognition of cars from side and rear views in [14]. The proposed model is trained by optimising the maximum likelihood in a part-based approach. The imagery used was, however, quite limited. Cars were centred in the image and only observed from the rear or from the side. Changes in illumination and scale were not substantial compared to real situations in urban environments.

To deal with changes in objects perceived scale, [1] proposed a sub-sampling method which modifies its sub-sampling interval based on the size of the object in imaging data. Representative samples are deterministically stored in a data base and compared against in-coming images at run time to perform object recognition. We address the problem of variations in scale using laser returns to define Regions Of Interest (ROI) in the image.

Ramos et al [17] have demonstrated recognition and segmentation of objects in unstructured environments using camera images and generative models. Mixtures are generatively trained through Variational Bayesian Expectation Maximisation (VBEM) for models representing the background and the object. Although this algorithm is fast and provides segmentation, no spatial or temporal dependencies are taken into account.

Within the robotics community, researchers have recently developed representations of the environment integrating more than one modality. In [12], a 3D laser scanner and loop closure detection based on photometric information are brought together in 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 [16], 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 readily allow the inference of landmarks' class which could contribute to higher level reasoning.

Object recognition based on laser and video data has been demonstrated in [10]. 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. In contrast, we learn a CRF classifier with the Virtual Evidence Boosting algorithm which performs feature selection in both datasets in order to minimize the classification error on training data. The VEB algorithm can, as it is, learn a classifier given as many data types as available and is not restricted to laser and vision inputs. For instance, VEB has been applied in the context of activity recognition to a data set containing inputs from thermometers, barometers, accelerometers, microphones, phototransitors and GPS units [7], [20].

The key contribution of this work is to present a probabilistic model for object recognition which integrates spatial and temporal correlations and can be learnt given any types of labeled data.

III. CONDITIONAL RANDOM FIELDS

This section provides a brief description of conditional random fields (CRF) and virtual evidence boosting (VEB), an extremely efficient way of learning CRF parameters for arbitrary feature functions (see [21] and [7] for more information).

A. Model Description

Conditional random fields are undirected graphical models developed for labeling sequence data [6]. CRFs directly model $p(\mathbf{x}|\mathbf{z})$, the *conditional* distribution over the hidden variables $\mathbf{x}$ given observations $\mathbf{z}$. This is in contrast to generative models such as Hidden Markov Models or Markov Random Fields, which apply Bayes rule to infer hidden states [15]. Due to this structure, CRFs can handle arbitrary dependencies between the observations $\mathbf{z}$, which gives them substantial flexibility in using high-dimensional feature vectors.

The nodes in a CRF represent hidden states, denoted $\mathbf{x} = \langle \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n \rangle$, and data, denoted $\mathbf{z}$. The nodes $\mathbf{x}_i$, along with the connectivity structure represented by the undirected edges between them, define the conditional distribution $p(\mathbf{x}|\mathbf{z})$ over the hidden states $\mathbf{x}$. Let $\mathcal{C}$ be the set of cliques in the graph of a CRF. Then, a CRF factorizes the conditional distribution into a product of *clique potentials* $\phi_c(\mathbf{z}, \mathbf{x}_c)$, where every $c \in \mathcal{C}$ is a clique in the graph and $\mathbf{z}$ and $\mathbf{x}_c$ are the observed data and the hidden nodes in the clique c, respectively. Clique potentials are functions that map variable configurations to non-negative numbers. Intuitively, a potential captures the "compatibility" among the variables in the clique: the larger the potential value, the more likely the configuration. Using clique potentials, the conditional distribution over the hidden states is written as

$$p(\mathbf{x} \mid \mathbf{z}) = \frac{1}{Z(\mathbf{z})} \prod_{c \in \mathcal{C}} \phi_c(\mathbf{z}, \mathbf{x}_c), \tag{1}$$

where $Z(\mathbf{z}) = \sum_{\mathbf{x}} \prod_{c \in \mathcal{C}} \phi_c(\mathbf{z}, \mathbf{x}_c)$ is the normalizing partition function. The computation of this partition function can be exponential in the size of $\mathbf{x}$. Hence, exact inference is possible for a limited class of CRF models only.

Potentials $\phi_c(\mathbf{z}, \mathbf{x}_c)$ are described by log-linear combinations of *feature functions* $\mathbf{f}_c()$, *i.e.*,

$$\phi_c(\mathbf{z}, \mathbf{x}_c) = \exp\left(\mathbf{w}_c^T \cdot \mathbf{f}_c(\mathbf{z}, \mathbf{x}_c)\right), \tag{2}$$

where $\mathbf{w}_c^T$ is a weight vector, and $\mathbf{f}_c(\mathbf{z}, \mathbf{x}_c)$ is a function that extracts a vector of features from the variable values. Using feature functions, we rewrite the conditional distribution (1) as

$$p(\mathbf{x} \mid \mathbf{z}) \quad = \quad \frac{1}{Z(\mathbf{z})} \exp\left\{ \sum_{c \in \mathcal{C}} \mathbf{w}_c^T \cdot \mathbf{f}_c(\mathbf{z}, \mathbf{x}_c) \right\} \tag{3}$$

B. Inference

Inference in CRFs can estimate 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), as defined in (3). Both tasks can be solved using

belief propagation (BP) [13], which works by sending local messages through the graph structure of the model. Each node sends messages to its neighbours based on messages it receives and the clique potentials, which are defined via the observations and the neighborhood relation in the CRF. A message a particular node k sends to its neighbour i is defined as:

$$\mu_{k\to i}(\mathbf{x}_i) = \alpha \sum_{\mathbf{x}_k} \phi(\mathbf{x}_k, \mathbf{x}_i)\, \phi(\mathbf{x}_k, \mathbf{z}_k) \prod_{j\in n(\mathbf{x}_k), j\neq i} \mu_{j\to k}(\mathbf{x}_k) \quad (4)$$

where $n(\mathbf{x}_k)$ denotes the neighbours of a node k. Here, the first potential corresponds to neighborhood potential between $\mathbf{x}_k$ and $\mathbf{x}_i$, and the second potential measures the consistency between the state $\mathbf{x}_k$ and the observation $\mathbf{z}_k$. Messages are propagated until convergence or until a maximum number of iterations is reached.

BP provides exact results in graphs with no loops, such as trees or polytrees. However, since the models used in our approach contain various loops due to temporal relationships, we apply loopy belief propagation, an approximate inference algorithm that is not guaranteed to converge to the correct probability distribution [11]. Fortunately, in our experiments, this approximation turned out to be reasonably accurate even when loopy BP failed to converge (the maximum number of iterations is reached).

C. Learning via Virtual Evidence Boosting

Learning a CRF model involves determining the weights used in the clique potentials (2) that determine the probabilistic relationships of the model. CRFs are trained discriminatively by maximizing the conditional likelihood (3) of labeled training data. This optimization is typically done by gradient-based techniques such as L-BFGS, where gradients are computed using inference in the CRF model. In order to avoid computationally complex inference for gradient computation, several researchers applied pseudo-likelihood training, which can be performed without running inference [21], [8].

While CRFs can handle extremely high-dimensional continuous and discrete features, the integration of continuous features is not straightforward. This is due to the fact that the incorporation of raw, continuous features in CRFs is similar to uni-modal Gaussian likelihood models in generative approaches such as hidden Markov models. Obviously, such simple likelihoods are not well suited to model more complex, multi-modal features and sensor data. Recently, researchers have applied boosting in order to discretize continuous features into binary threshold functions, called decision stumps [4]. The thresholds are learned by minimizing an exponential loss function of the training data [2]. The decision stumps are then used as binary features in a CRF, and the weights for these features are learned using regular CRF training [4].

More recently, Liao and colleagues introduced virtual evidence boosting (VEB), which incorporates feature discretization into CRF training [7]. VEB jointly learns an appropriate discretization of continuous features, the weights of these features, and the weights of neighborhood potentials of the CRF. In essence, this is done by performing boosting on both the features and the neighborhood potentials of the CRF. VEB has demonstrated superior performance on both synthetic and real data. Furthermore, the automatic feature discretization makes VEB extremely flexible and allows the incorporation of arbitrary, continuous and discrete features. Since modelling flexibility is crucial in the context of our object recognition task, we chose to use VEB for learning the parameters of our CRFs.

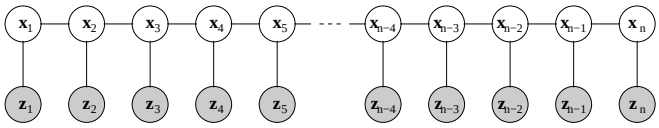

Fig. 1. Graphical model of a linear chain CRF for one time slice object recognition. Each hidden node $\mathbf{x}_i$ represents one beam in a laser scan. The nodes $\mathbf{z}_i$ correspond to spatial features extracted from the laser scan and local visual features extracted from a camera image.

IV. CRFs FOR OBJECT RECOGNITION

This section describes the deployment of the CRF framework to perform object recognition. This work focuses on the problem of detecting cars in an outdoor urban environment given laser data and monocular colour images. Fig. 4 shows examples of laser scans projected into the corresponding image according to the procedure described in [23]. These images illustrate the typical variety in terms of classes of objects, scales and lighting conditions encountered in outdoor urban environments.

The CRF framework is applied to this data by converting each scan into a linear chain CRF such as the one displayed in Fig. 1. Each node of this CRF represents a laser return. The hidden variable to be estimated is the class of the return, i.e class "car" or class "other". The parametrization of such a CRF model of a laser scan is now described. We then explain how this model is further incorporated into a more elaborated representation which takes temporal relationships into account.

A. One time slice model

To jointly estimate all the labels of a laser scan, observations are first passed to each node via local feature functions $\mathbf{f}_{\text{local}}()$. Each node performs local estimation and then propagates its local estimate across the network via a second type of feature function, $\mathbf{f}_{\text{compatibility}}()$, which encodes the neighborhood relationships amongst adjacent nodes. The compatibility functions are learnt by the VEB algorithm in a form of a 2×2 matrix. This matrix correlates the distribution over classes computed by two neighbour nodes. In the experiments two types of local feature functions are used: geometric feature functions and visual feature functions. We now detail each of them.

Geometric laser features: These features capture geometric properties of the objects detected in the laser scan. While local shape can be captured by various types of features, we chose to implement simple shape features measuring

distance, angle, and number of out of range returns between two beams. The resulting feature function has the form

$$\mathbf{f}_{\mathrm{geo}}(i, z_A) = \mathrm{concat}\left(\mathbf{f}_{\mathrm{dist}}(\mathrm{i}, \mathrm{z_A}), \mathbf{f}_{\mathrm{angle}}(\mathrm{i}, \mathrm{z_A}), \mathbf{f}_{\mathrm{oor}}(\mathrm{i}, \mathrm{z_A})\right), \quad (5)$$

where i indexes one of the returns in scan z_A. The concat function performs a concatenation operation, and the resulting function $\mathbf{f}_{\mathrm{geo}}(i, z_A)$ returns a vector of dimensionality 213, as specified next.

To generate distance features $\mathbf{f}_{\mathrm{dist}}$, we compute for each point $z_{A,i}$ in scan A its distance to other points in scan A. These other points are chosen based on their relative indices in the scan. With k being an index offset, the distance feature corresponding to points $z_{A,i}$ is computed as follows:

$$\mathbf{f}_{\mathrm{dist}}\left(i, k, z_A\right) = \frac{\left\|z_{A,i} - z_{A,i+k}\right\|^2}{\sigma^2}. \quad (6)$$

In our implementation this feature is computed for index offsets k varying from -10 to $+10$.

Another way to consider local shape is by computing the angles of points w.r.t their neighbours. The angle of a point $z_{A,i}$ is defined as the angle between the segments connecting point i to its neighbours:

$$\mathbf{f}_{\mathrm{angle}}\left(i, k, z_A\right) = \frac{\left\|\angle\left(\overline{z_{A,i-k} z_{A,i}}, \overline{z_{A,i} z_{A,i+k}}\right)\right\|^2}{\sigma^2}. \quad (7)$$

Again, we vary the index offset k from -10 to $+10$.

The out of range feature $\mathbf{f}_{\mathrm{oor}}$ counts the number of "out of range" returns between pairs of non "out of range" returns. The idea is to encode open areas in the laser scan.

Visual features: In addition to geometrical information, a CRF model learnt with the VEB algorithm can seamlessly integrate the vision data provided by a monocular colour camera. A first step consists of registering the vision sensor and the laser range-finder with respect to each other using the calibration procedure described in [23]. The laser returns can then be projected into the associated image. The visual features extracted from this image capture color and texture information in the window (or ROI) centered around the laser return. The edge length of the window is set to be 1 meter for a range of 4 meters. This size is converted into number of pixels using the camera's intrinsic parameters and adjusted depending on the range measurement. Changing the size of the extracted patch as a function of range is a way to deal with the variation in scales as an object moves from the background to the foreground of the image. It was verified that the use of a size varying window improves the experimental results by 4%.

The visual feature function has the following form:

$$\mathbf{f}_{\mathrm{visu}}(p_i, p_{i-1}) = \mathrm{concat}\left(\mathbf{f}_{\mathrm{texture}}(\mathrm{p_i}, \mathrm{p_{i-1}}), \mathbf{f}_{\mathrm{colour}}(\mathrm{p_i}, \mathrm{p_{i-1}})\right), \quad (8)$$

where p_i is the image patch corresponding to return i. $\mathbf{f}_{\mathrm{texture}}(p_i, p_{i-1})$ returns a vector containing the steerable pyramid [19] coefficients of image patch i, and the difference between the steerable pyramids computed at patch i and at patch $i-1$. $\mathbf{f}_{\mathrm{colour}}(p_i, p_{i-1})$ returns a vector containing the 3D RGB colour histogram of patch i and of its difference with patch $i-1$. Only neighbour $i-1$ is used to limit the dimensionality of $\mathbf{f}_{\mathrm{visu}}$ which is already around 7000.

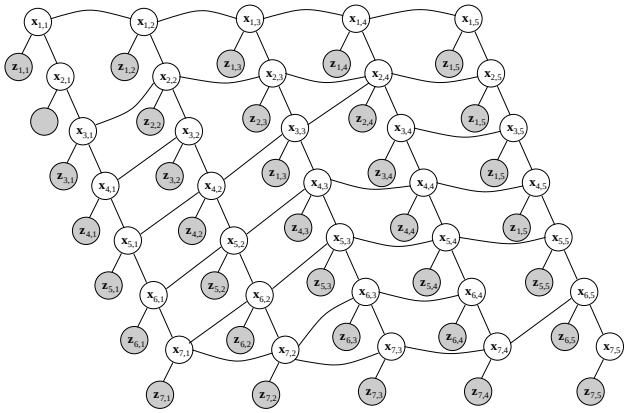

Fig. 2. Graphical model of the spatio-temporal classifier. Nodes $\mathbf{x}_{i,j}$ represent the i-th laser beam observed at time j. Temporal links are generated between time slices based on the ICP matching algorithm.

B. Recognition over time

Due to the sequential nature of robotics applications, a substantial amount of information can be gained by taking into account prior and posterior data when available. We now present a model that achieves temporal smoothing in addition to exploiting the structure of one scan. This model is displayed in Fig. 2.

In this work, the temporal connections are instantiated such that they represent the associations found by the Iterative Closest Point (ICP) algorithm [24]. The pairwise potentials assigned to these connections are set to identity. Mathematically, $\phi_{\mathrm{temporal}}(\mathbf{x}_i, \mathbf{x}_j) = \delta(\mathbf{x}_i, \mathbf{x}_j)$, where δ is the indicator function. This set-up is justified by the fact that ICP associates returns that were generated by the same physical point. It follows that the integration of temporal information does not require additional learning. In earlier stages of this research, attempts have been made to learn the temporal relationships from data. Our tests show that setting ϕ_{temporal} to identity leads to better results.

Corresponding to different variants of temporal state estimation, our spatio-temporal model can be deployed to perform three different types of estimation.

- Off-line smoothing: All scans in a temporal sequence are connected using ICP. BP is then run in the whole network to estimate the class of each laser return in the sequence. During BP, each node sends to its neighbours the messages defined in (4) through structural and temporal links (vertical and horizontal links respectively in Fig. 2). In our experiments, BP is run for 100 iterations.
- On-line fixed-lag smoothing: Here, scans are added to the model in an on-line fashion. To label a specific scan, the system waits until a certain number of future scans becomes available, and then runs BP taking past and future scans into account.
- On-line filtering: In this case the spatio-temporal model only includes scans up to the current time point.

V. EXPERIMENTS

The experiments were performed using outdoor data collected with a modified car travelling at 0 to 40 km/h. The car drove along several loops in the university campus which has structured areas with buildings, walls and cars, and areas less structured with bush, trees and lawn fields. The overall dataset contains 4,500 images which represents 20 mins of logging. Laser data was acquired at a frequency of 4Hz using a SICK laser. The models presented in Sec. IV are used to estimate the class of each return in the laser scans. Here, the classification problem is binary and involves the classes "car" and " other".

Table I summarizes the experimental results in terms of classification accuracy. The accuracies are given in percentages and computed using 10 fold cross validation on a set of 100 manually labeled scans. For each cross validation, the different models were trained for 200 iterations. The VEB algorithm was run allowing the learning of pairwise relationships only after iteration 100. We found that this increases the weights on the local feature and improves classification results.

training set	geo only	visu only	geo+visu	geo+visu
number of time slices in the model	1	1	1	∓ 10
CRF	68.93	81.79	83.26	88.08
logitboost	67.64	81.52	83.22	$\times$

TABLE I

CLASSIFICATION ACCURACY (IN %)

The first line of Table I indicates the types of features used to learn the classifier. Three different training sets are used: one using geometric features only, one containing visual features only, and a third one containing both geometric and visual features. The second line of table I indicates the number of time slices in the network used to perform classification. "1" means that a network as the one presented in Fig. 1 was used. "$\mp$ 10 " refers to the classifier shown in Fig. 2 instantiated with the 10 scans observed before and after the labeled scan.

Two types of classifiers were used: CRFs and logitboost classifiers. CRFs take into account the neighbours to perform classification (Fig. 1). Logitboost learns a classifier that only supports independent classification of each scan return without using neighborhood information [3]. Logitboost is used here for comparison purposes to investigate the gain in accuracy obtained with a classifier that takes into account the structure of the scan.

The first three columns of Table I show that classification results are improving as richer features are used for learning. The first three columns also show that the CRF models lead to slightly more accurate classification. The improvement brought by a CRF classifier is made clearer when classification results are expressed in terms of the Receiver Operating Characteristics (ROC) shown in 3.

Additionally, as presented in Sec. IV-B, our model can readily be extended into a spatio-temporal model. The latter

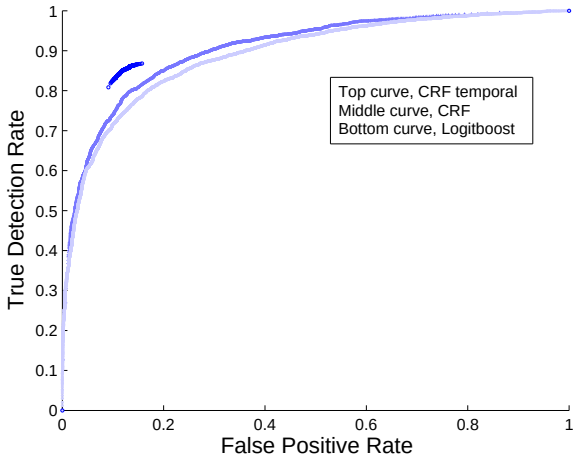

Fig. 3. ROC curves. Models learnt using visu+geo features.

leads to an improvement of 5% in classification accuracy (right column of table I). This shows that the proposed spatio-temporal model, through the use of past and future information, is better for object recognition. The associated ROC curve displayed in Fig. 3 shows the same trend: it is above the two others. The cross in the bottom right of the table refers to the fact that logitboost does not allow the incorporation of temporal information in a straightforward manner.

CRF models also generate better segmentation of cars in laser scans. This can be quantified using the metric called String Edit Distance (SED)[18]. Intuitively, this metric tells us whether classification results capture the true arrangement of objects in a scene. It penalizes series of estimates that do not respect the true sequence of blocks with the same label. For example, given the ground truth "ccooccoo" (where 'c' and 'o' stand for 'car' and 'other', respectively), the estimated sequence "cocococo" is more penalized (larger SED) than "ooccooccoo". This is because the latter estimate is more similar to the true sequence in terms of blocks of returns with the same label. Note that in this example the SED is larger for the sequence with higher classification accuracy which illustrates the ability of the SED metric to capture different properties.

Table II presents the classification results in terms of SED. The values show that the spatio-temporal model gives the best results in terms of classification accuracy as well as in terms of SED. The CRF classifiers, through their ability to represent spatial and temporal dependencies, are better able to capture the true arrangement of the observed objects. This property is extremely beneficial for segmentation tasks, which is beyond the scope of this paper.

These results match with the ones presented in [4] where it is shown that a CRF based approach is better able to capture the structure of indoor environments.

Fig. 4 shows four examples of classification results. It can be seen that the spatio-temporal model gives the best results. While the logitboost classifier tends to alternate correct and incorrect classification across one scan, the ability of the

Classifier (training set = geo+visu)	logitboost	CRF	CRF
number of time slices in the model	1	1	$\mp$ 10
String Edit Distance	9.5	5.6	2.4

TABLE II
STRING EDIT DISTANCES

CRF classifiers to capture the true arrangement is illustrated by the block like distribution of the inferred labels. Figure 4(b) shows the three classifiers failing in a very dark area of the image (right of the image). In the rest of the image which is still quite dark, as well as in images with various lighting conditions (Fig. 4(a), 4(c) and 4(d)) the saptio-temporal model does provide good classification results.

Inference in a one time slice CRF takes less than a millisecond on a Intel Xeon 2.33GHz desktop computer. Inference in the spatio-temporal model made of 21 time slices and containing about 1500 nodes takes on average 15 milliseconds. This shows that the proposed spatio-temporal model is appropriate for real-time applications. Learning the model requires around 40 minutes and can be performed off-line.

VI. CONCLUSIONS AND FUTURE WORK

We have presented a general probabilistic model for object recognition that incorporates spatial and temporal dependencies. The model is developed in the context of CRFs trained with Virtual Evidence Boosting (VEB).

Experiments were performed in an urban environment for recognition of cars. Laser and monocular camera information were used to detect cars under different illumination conditions, viewpoints, scales and occlusion. The experiments demonstrate that our approach achieves superior classification performance by modeling the beams of a laser scan jointly and by integrating observations over time. This is expected as the additional information obtained as the vehicle approaches an object, observing it in larger scales, can correct past predictions. Furthermore, by building on CRF models, the approach can incorporate arbitrary, correlated, continuous and discrete features extracted from sensor data. VEB parameter learning automatically extracts the most useful feature functions from the data.

While the focus of this paper was on binary car classification, our framework can be readily applied to multi-class object recognition. We are currently investigating the learning of additional object classes, such as persons, buildings, trees, and other vegetation. The ability to on-line estimate such a variety of objects will be extremely helpful for navigation and building expressive models of outdoor environments.

One limitation of our current system is its dependence on the availability of fully labeled training data. However, VEB training can be applied to partially labeled data. In this case, only a subset of laser beams are labeled, and the model parameters are learned by computing the evaluation function at the labeled data points only [5]. Using partial labeling, our approach can be applied to far larger and hence diverse sets of laser scans and images, which results in better generalization performance. First experiments with partially labeled data show very promising results.

VII. ACKNOWLEDGMENTS

The authors would like to thank Lin Liao for providing the VEB code and Jose Guivant, Juan Nieto, Roman Katz 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] W. Efenberger and V. Graefe. Distance-invariant object recognition in natural scenes. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 3, pages 1433–1439, 1996.

[2] Y. Freund and R.E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1), 1997.

[3] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *The Annals of Statistics*, 28(2):337–407, 2000.

[4] 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.

[5] F. Jiao, S. Wang, C. Lee, R. Greiner, and D. Schuurmans. Semi-supervised conditional random fields for improved sequence segmentation and labeling. In *Proc. of Joint Conference of the International Committee on Computational Linguistics and the Association for Computational Linguistics*, 2006.

[6] 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.

[7] 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.

[8] L. Liao, D. Fox, and H. Kautz. Extracting places and activities from GPS traces using hierarchical conditional random fields. *International Journal of Robotics Research (IJRR)*, 26(1), 2007.

[9] B. Limketkai, L. Liao, and D. Fox. Relational object maps for mobile robots. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2005.

[10] 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.

[11] 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.

[12] 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.

[13] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, Inc., 1988.

[14] A. Quattoni, M. Collins, and T. Darrell. Conditional random fields for object recognition. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.

[15] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. In *Proceedings of the IEEE*. IEEE, 1989. IEEE Log Number 8825949.

[16] 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.

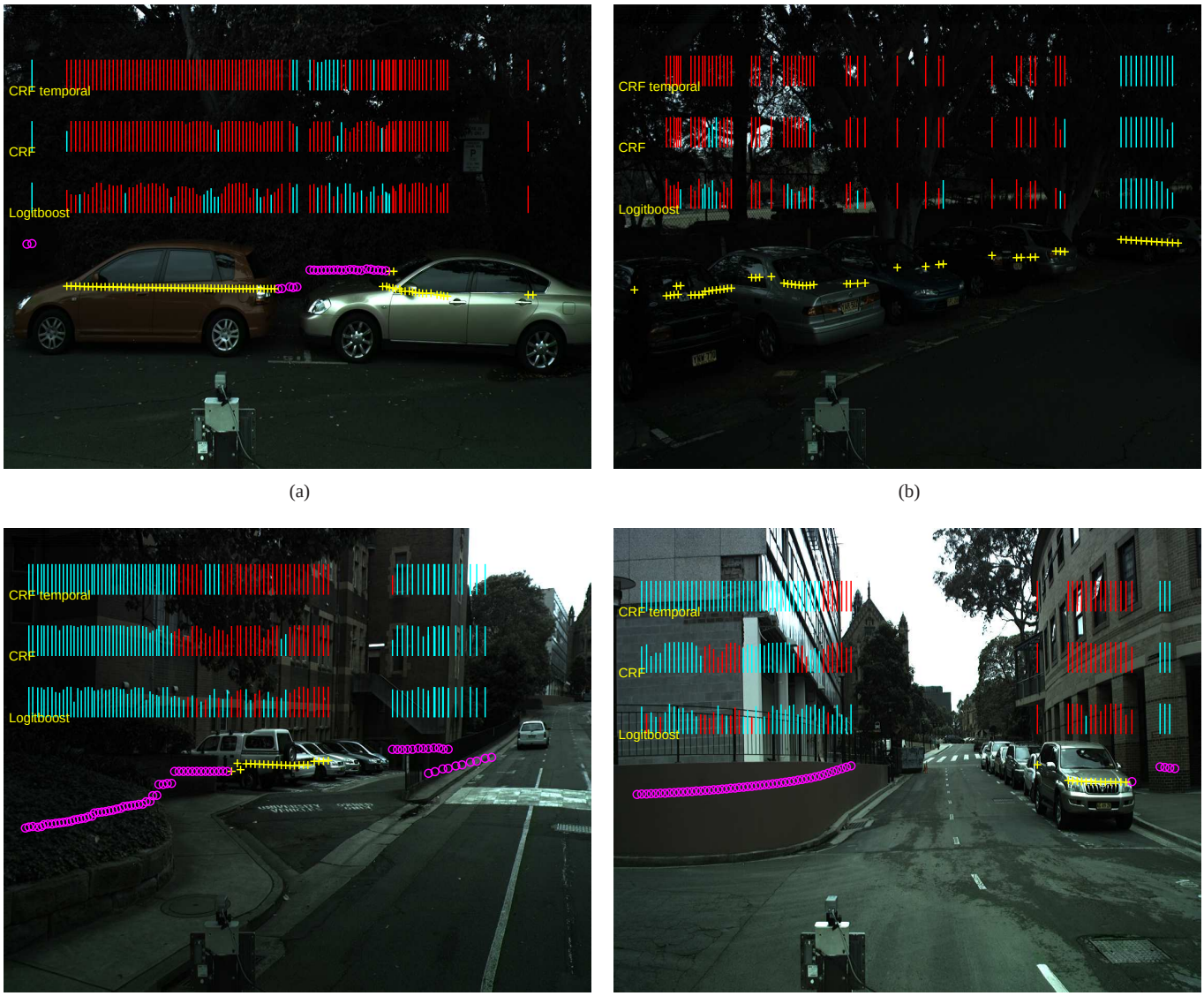

Fig. 4. Examples of classification results. The label of the returns are displayed with the markers + in yellow (brighter) and ○ in magenta (darker) for the class "car" and the class "other" respectively. The height of the bar above each return represents the confidence associated with the inferred label. The colour of the bar indicates the inferred label: red (darker) means that the inferred label is "car" and cyan (brighter) refers to the label "other". The classifiers used to generate the different estimates are precised on the left.

[17] F. T. Ramos, S. Kumar, B. Upcroft, and H. F. Durrant-Whyte. Recognising and segmenting objects in natural environments. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Beijing, China, 2006.

[18] D. Sankoff and J. Kruskal, editors. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison-Wesley, 1983.

[19] E. Simoncelli and W. Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *Proc. of 2nd International Conference on Image Processing*, 1995.

[20] A. Subramanya, A. Raj, J. Bilmes, and D. Fox. Hierarchical models for activity recognition. In *Proc. of the International Workshop on Multimedia Signal Processing (MMSP)*, 2006.

[21] C. Sutton and A. McCallum. An introduction to conditional random fields for relational learning. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning*, chapter hi. MIT Press, 2006.

[22] P. Viola and M. Jones. Robust real-time object detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[23] 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.

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