ASSESSING MAP QUALITY AND ERROR CAUSATION USING CONDITIONAL RANDOM FIELDS

Manjari Chandran-Ramesh and Paul Newman
Robotics Research Group, University of Oxford, Oxford, OX13PJ
{manjari,pnewman}@robots.ox.ac.uk

Abstract: This paper is about assessing the quality of maps built by a mobile robot. We extend previous work, which used solely geometric considerations, and use both temporal and spatial properties of the map to perform a binary classification of "plausible" and "suspicious". The use of the former allows the existence of low quality areas of the map to be attributed to missed loop closure events or local, online mapping errors. With an eye on our intended domain of urban operation, we adopt a Conditional Random Field as the probabilistic framework in which to model the spatial and temporal relationships between planar patches. The map quality labels are derived by using standard graph cuts optimization techniques. The approach is then illustrated with map created of an urban environment using data from a $3D$ laser range scanner mounted on a mobile robot.

Keywords: Mapping, Map Quality, Navigation, Mobile Robotics

1. INTRODUCTION

Mapping an environment accurately is an important area of research in the field of mobile robotics and has received much attention. However, the task of explicitly evaluating the quality of these maps has not received much attention. Being able a assess the quality of a map is an important problem. Firstly it allows the value of the map to be quantified in the light of its intended use — for example "how useful is a map for future localization?". Secondly, it opens the door to planning remedial action — "this region is poor and needs revisiting".

However, a definition for a map quality measure can be to a degree an abstract measure. Previous work has made an attempt to define this abstractness as the quality measure belonging to binary classes – "plausible" and "suspicious", where "plausible" regions have well-defined, non-self-intersecting object borders. Frameworks that implement the Markov property has been shown to be effective in capturing contextual information (Lafferty *et al.*, 2001; Liao *et al.*, 2007; Anguelov *et al.*, 2005; Triebel *et al.*, 2006). Hence the Conditional Random Fields framework has been used to assess the quality of a given $3D$ point-cloud map. The classification is performed using the spatial property of the points. In this work, we also consider the temporal property of the points, and importantly do so in the same unified probabilistic framework. The use of the temporal property suggests causation of the low quality regions as either missed loop closure or online mapping error.

The approach we take in this paper is to first segment the given 3D point-cloud map into plane patches. These plane patches are then given to a classifier to be labelled. The problem of classification is considered as a supervised learning problem. Hence there are two steps namely the learning step where the classifier is trained on manually labelled data to learn the set of parameters and the inference step where based on the parameters learnt, the map is classified.

This paper is divided into the following sections. A brief description of Conditional Random Fields and using it for binary classification is dealt with in Section 2. Section 3 briefly describes the spatial compatibility measure for completeness and introduces the temporal compatibility measure. Section 4 describes the steps involved in the classification of the map as "plausible", "suspicious" due to missing loop closure and "suspicious" due to local online mapping error. The effectiveness of the approach is then illustrated using maps built of urban environments in section 5.

2. USING CONDITIONAL RANDOM FIELDS TO DETERMINE PARTITIONS ON MAP QUALITY

In this section, the use of a Conditional Random Field framework for the purpose of classification of portions of a given point-cloud map is described. A conditional random field (CRF) can be viewed as a Markov random field globally conditioned on the random variable representing the observation sequence (Wallach, 2004) where a Markov random field is an undirected graphical model which has a set of nodes each of which corresponds to a random variable or group of variables, as well as a set of links each of which connects a pair of nodes (Besag, 1986; Geman and Geman, 1984). For a detailed explanation of CRFs please refer (Lafferty *et al.*, 2001).

In this work, the classification is modelled as a two stage hierarchical binary classification. The labelling problem at each stage consists of a network of N

nodes that represent the random variable, X, over the data sequence having the possible labels: $y_i \in \{-1, 1\}$ represented by the random variable Y. The nodes here are the plane patches, consisting of a subset of points, segmented from the $3D$ point-cloud and their alignment with the neighboring plane patches constitute the edges.

The conditional distribution, $p(y|x)$, is factorized into a product of fully connected sub-graphs or clique potentials $\phi_c(x_c, y_c,)$ where $c \in C$ is a clique in the set of cliques C and x_c and y_c are the nodes and its labels in that clique. The potential function represents the constraints on the configuration or the "compatibility" between the nodes in the clique. When pairwise CRFs are considered each clique has local or node potentials, $\phi_i(x_i, y_i)$ for each node i, and pairwise or edge potentials,$\phi_{ij}(x_{ij}, y_i, y_j)$ for an edge between nodes i and j. The conditional distribution can be written as

$$p(y|x) = \frac{1}{Z(x)} \prod_{i=1}^{N} \phi_i(x_i, y_i) \prod_{ij \in E} \phi_{ij}(x_{ij}, y_i, y_j) \quad (1)$$

where $Z(x)$ is the partition function given by

$$Z(x) = \sum_{y} \prod_{i=1}^{N} \phi_i(x_i, y_i) \prod_{ij \in E} \phi_{ij}(x_{ij}, y_i, y_j)$$

N is the number of nodes, E is the set of edges $\{ij\}(i < j)$ in the graph, $\phi_i(x_i, y_i)$ is the node or unary potential, $\phi_{ij}(x_{ij}, y_i, y_j)$ is the edge or binary potential.

Using the Hammersley-Clifford fundamental theorem of random fields (Lafferty *et al.*, 2001), the potential functions can now take the form

$$p_\Theta(y|x) = \frac{1}{Z(x)} \exp\left(\sum_{i \in N, k} \mu_k f_k(x_i, y_i) + \sum_{ij \in E, k} \lambda_k g_k(x_{ij}, y_i, y_j) \right) \quad (2)$$

where f_k and g_k are vectors of local and pairwise features respectively and $\Theta = (\mu_1, ..., \mu_k; \lambda_1, ..., \lambda_k)$ are the parameters or weights to be estimated from training data. The estimation of these parameters from the training data is the learning step when using CRFs for the purpose of classification. Consider a training set $\{x^{(t)}, y^{(t)}\}$ that are independently and identically distributed. All cliques are made to share the same parameters to reduce the amount of training data required. The distribution in equation (2) over all the training data as a function of the parameter set Θ, is the likelihood given by $p(\{y^{(t)}\}|\{x^{(t)}\}, \Theta)$. Typically, the logarithm of the likelihood is used for estimation. The log-likelihood for a CRF is given by

$$L(\Theta) = \sum_{t} \left[\log \frac{1}{Z(x^t, \Theta)} + \sum_{k} \mu_k f_k(x^t, y^t) + \sum_{k} \lambda_k g_k(x^t, y^t) \right] \quad (3)$$

The above function is concave and hence there is a guaranteed convergence to the global maximum (Wallach, 2004). Parameter estimation in CRFs is an actively researched field and there exist various techniques. In this work, the maximum psuedo-likelihood method has been used to train the parameter set Θ. For more details about this method, please refer (Liao *et al.*, 2007; Besag, 1975).

Once parameters have been estimated in the learning step, they can be used to infer the labels of an unlabelled data set. This is the inference step and is done by maximizing the conditional distribution of the labels given the feature vectors and the parameter set. This can be written as

$$Y = \arg\max_{y} p_\Theta(y|x) \quad (4)$$

where Y is the array of labels for the nodes. Optimizations based on graph cuts (Boykov *et al.*, 2001; Kolmogorov and Zabin, 2004; Boykov and Kolmogorov, 2004; Szeliski *et al.*, 2006) are a popular method to do this kind of maximization as they are guaranteed to find the global maximum when binary labels are required. Now, the compatibility measures, spatial and temporal, used to calculate the feature vectors f_k and g_k are defined in the next section.

3. COMPATIBILITY MEASURES

This section defines the spatial and temporal compatibility which are used to compute the node and edge potentials in the CRF framework. The spatial properties are used to classify the map as "plausible" and "suspicious" in a first pass. The temporal properties are used to further identify the probable cause of the first pass classification. In the first stage of classification, the spatial features, x_i and x_{ij} define the vectors f_k and g_k or the node and edge potentials respectively. In the second stage of classification, the temporal features, t_i and t_{ij} define the node and edge potentials respectively. For completeness, the spatial features introduced in (Chandran-Ramesh and Newman, 2007) are briefly described before the temporal features are introduced.

3.1 Using spatial compatibility

The spatial features are required to answer the questions – "is the subset of points acceptable as a plane patch?" and "is the alignment of the plane patch with respect to its neighbors reasonable?" To check if a subset of points represent a plane patch, the metrics defined should measure how accurately the points fit a plane and whether the fitted plane is more two-dimensional than cubic in nature. To check if the alignment between neighboring plane patches is reasonable, the various cases of alignments first needs to explored. These alignments can be broadly classified into plane patches that are parallel, those that are not parallel, but do not intersect and those that intersect. Of these alignments, the parallel planes that are separated only by a small distance and the intersecting plane patches that do not represent corners, are considered "suspicious" alignments.

The intuitive considerations are formalized as features x_i and x_{ij} given below.

$$\begin{aligned} x_{i1} &= \frac{1}{\frac{1}{M_i}\sum_{j=1}^{M_i} d^2_{j\to i}} \\ x_{i2} &= \frac{1}{\overrightarrow{Z}.\overrightarrow{n_i}} \\ x_{i3} &= \frac{V_i}{A_i} \end{aligned} \tag{5}$$

where x_{i1}, x_{i2}, x_{i3} denote the spatial features of the i^{th} plane patch, M_i refers to the number of points in the i^{th} plane patch, $d_{j\to i}$ is the distance of each point j in the i^{th} plane patch to plane patch fitted, Z is the Z axis vector, n_i is the normal to the i^{th} plane patch, V_i is the volume of the $3D$ convex hull fitted to the points in the i^{th} plane patch and A_i is the area of the $2D$ convex hull fitted to the points of the i^{th} plane patch, projected to the XY plane.

$$x_{ij} = 1 - \{W_{ij}(\rho_{ij}) \times (A_{ij} + R_{ij})\} \tag{6}$$

where x_{ij} is the feature for measuring the plausibility of the geometry between the i^{th} plane patch and the j^{th} plane patch. W_{ij} is the weight as an exponential function of ρ_{ij} which is the closest distance between the plane patches i and j, calculated as the shortest distance between points in both plane patches. A_{ij} is the normalized overlapping area between the plane patches i and j, when the patches are projected on to each other and R_{ij} is the normalized ratio of the intersecting edges between the plane patches and is only for intersecting planes, scoring 0 when the plane patches are parallel. For further details and a more thorough explanation of these features, please refer (Chandran-Ramesh and Newman, 2007).

3.2 Using Temporal Compatibility

The temporal features are used to further classify the "suspicious" regions of the map into the causes of error namely, due to local mapping error and due to missed loop closure. In areas that denote loop closure, data has been collected by the robot at two time stamps that are considerably apart due to the robot revisiting the environment. This large difference in time stamp is used to differentiate the areas requiring loop closure from other areas. The questions that the temporal features try to answer are – "how likely the points in a plane patch are all part of a non-loop closure region?" and "how likely do two neighboring plane patches belong to non-loop closure scenario?" This can be formalized into the node potential feature t_i and the edge potential feature t_{ij}.

The node potential requires to score how likely the points in a plane patch come from observations with similar time stamps and hence all belong to a non-loop closure region. Hence the variance in time of the points in the plane patch p_i constitute the temporal feature t_i.

$$t_i = Var(t_{ik}) = E[(t_{ik} - \mu)^2] \tag{7}$$

where t_i is the temporal feature of the plane patch p_i, t_{ik} is the time of each point k in the plane patch and μ is the mean time of the plane patch.

The edge potential needs to capture how likely two neighboring plane patches are "suspicious" spatially due to local mapping error. Alternatively, the edge potential function requires to score how likely the neighboring plane patches are from a missed loop closure area. The time difference between the two plane patches can be used to score this.

$$t_{ij} = \frac{t_{p_i} - t_{p_j}}{t_{min}} \tag{8}$$

where t_{ij} is the measure of likelihood of missed loop closure between plane patches p_i and p_j, t_{p_i} and t_{p_j} are the mean time stamps of plane patches p_i and p_j and t_{min} is the minimum time difference between any two consecutive robot poses.

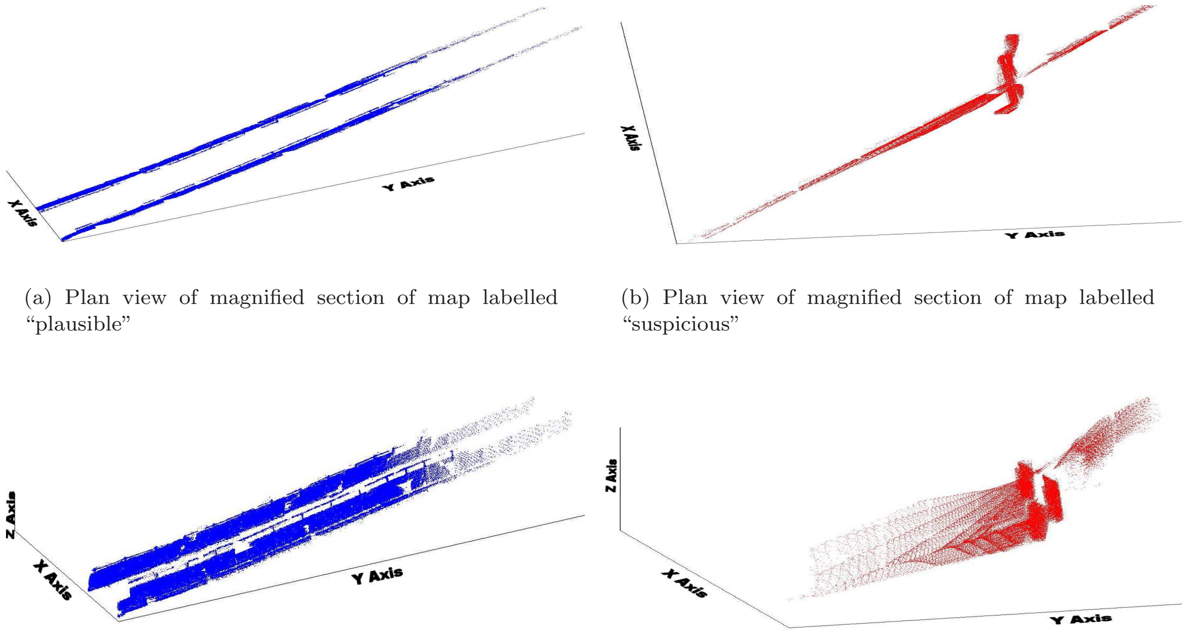
Now, with all the features defined, the steps involved in obtaining the map quality and error causation labels are explained in the next section.

4. ASSIGNING MAP QUALITY AND ERROR CAUSATION LABELS

The classification problem is divided into a hierarchical approach where the regions are first classified as "plausible" and "suspicious" and then in the second stage the "suspicious" regions are classified as due to missing loop closure or due to local mapping error. Each stage is a binary classification problem, with the algorithm finally returning three labels. In the first stage, the two possible labels for these plane patches are: $y_i \in \{-1, 1\}$ where -1 denotes "suspicious" and 1 denotes "plausible" and in the second stage, the two possible labels for the plane patches are: $y_i \in \{-1, 1\}$ where 1 denotes "suspicious" due to missed loop closure and -1 denotes "suspicious" due to local mapping errors.

In this work, the $3D$ point-cloud is segmented into plane patches consisting of a subset of points. These plane patches constitute the nodes and their spatial compatibility or the temporal compatibility with neighboring planes patches constitute the edges. When selecting the neighborhood for each plane, care must be taken that enough contextual information is provided while the computation costs of inference and learning are not compromised. In this work, the neighborhood system is defined as the *four* closest plane patches to the current plane patch with respect to euclidean distance within a radius of s meters.

The classification is a supervised learning problem. Hence first the parameters, Θ need to be learnt



(a) Plan view of magnified section of map labelled "plausible"

(b) Plan view of magnified section of map labelled "suspicious"

(c) $3D$ view of magnified section of map labelled "plausible"

(d) $3D$ view of magnified section of map labelled "suspicious"

Fig. 1. Figure shows the magnified section of a map in $3D$ that has been classified by the algorithm. The "plausible" portions are marked in blue and the "suspicious" portions in red. For clarity, the plan view of this portion is also shown, above the magnified portion. In (a, c), the two plane patches are at a reasonable distance apart. Hence the algorithm correctly labels this portion as "plausible". In (b, d), there are two planes that intersect, however not representing a corner. This is correctly labelled as "suspicious".

using the training data, that has been manually labelled, before the unlabelled map can be classified. In this case, there are two sets of parameters, one for each stage of classification. These are $\Theta^x = (\mu_1^x, \mu_2^x, \mu_3^x; \lambda_1^x)$ and $\Theta^t = (\mu_1^t; \lambda_1^t)$ for the spatial and temporal features respectively. Using maximum psuedo-likelihood estimation (Liao *et al.*, 2007), the learning step is performed once for parameters Θ^x (equations (3, 5, 6)) and once for parameters Θ^t (equations (3, 7, 8)), substituting t_i and t_{ij} for x_i and x_{ij} in equation (3).Once these parameters are known, the unlabelled map can be given to the algorithm and the labels inferred.

The input to the classifier are the plane patches that have been segmented from the given point-cloud and the output is the label of "plausible" or "suspicious" due to missing loop closure or "suspicious" due to local mapping errors, assigned to each of these plane patches.

The steps are as follows

(1) Spatial features of the planes, x_i, for each of the N planes are calculated using equation (5)
(2) Neighborhood for each of the planes is determined as the four closest planes with respect to euclidean distance
(3) Spatial measure of plausibility of alignment, x_{ij}, for each set of neighbors is calculated using equation (6)
(4) Substituting features x_i, x_{ij} and Θ^x in equation (2), the joint probability is maximized (refer equation (4)) using the optimization algorithm which returns the array of labels Y_1 labelling plane patches as "plausible" or "suspicious"
(5) The "suspicious" regions are then given to the second stage of the algorithm and the temporal feature, t_i is calculated using equation (7)
(6) Neighborhood for each of the planes in these subregions is determined as the four closest planes with respect to euclidean distance
(7) Temporal measure of likelihood of missed loop closure between plane patches, t_{ij}, is calculated for each set of neighbors using equation (8)
(8) Substituting features t_i, t_{ij} and Θ^t in equation (2), the joint probability is again maximized (refer equation (4)) which now returns the array of labels Y_2 labelling the current set of plane patches as "suspicious" due to missed loop closure or "suspicious" otherwise due to local mapping errors
(9) The two label arrays Y_1 and Y_2 are combined to give the label array Y which has 3 classes namely "plausible", "suspicious" due to missed loop closure and "suspicious" due to local mapping errors

5. EXPERIMENTAL RESULTS

In this section, the effectiveness of the above algorithm is illustrated using data gathered from an urban environment by a mobile robot. A mobile platform equipped with a $3D$ SICK laser scanner was used to capture this data. The environment chosen was that of a typical urban environment consisting of office buildings, roadways, foliage, railings, people, moving cars and archways. Vehicle routes were planned to include loops.

The $3D$ point-cloud map built by the mapping algorithm is first segmented into plane patches using the region growing based approach (Weingarten *et al.*, 2003). These plane patches are then manually labelled and from these, approximately $1300,000$ points segmented into 500 planes was used in the training step. Maximum psuedo-likelihood using the CRFtoolbox [1] was run until convergence and used to learn both the parameter sets, Θ^x and Θ^t. On inspection of the parameter set, it was found that the weights for the edge features in both the spatial and temporal stages are a factor of ten higher than the weights for the node features. This makes the labelling more sensitive to the alignments between plane patches, which is desired. Using the parameters learnt, the rest of the plane patches was given as input to the map quality section of the algorithm. This section of the algorithm then returned a label for each patch as “plausible” and “suspicious”. The “suspicious” regions of the map was then given as input to the second section of the algorithm, which returned whether the region is “suspicious” due to missed loop closure or local mapping errors.

Figure (1) shows a magnified section of a map that has been classified by the algorithm. The “plausible” portions are marked in blue and the “suspicious” portions in red. For clarity, the plan view of this portion is also shown above the $3D$ view. In figures (1(a), 1(c)), the “plausible” region of the map consists of a stretch of wall on either side of the robot’s pathway. Since these two plane patches are at a reasonable distance apart, the classifier correctly labels this portion as “plausible”. In figures (1(b), 1(d)), the two planes are intersecting. However, this does not represent a corner and it is not acceptable for two plane patches to intersect like this. These two plane patches are correctly labelled as “suspicious”.

Figure (2) shows a map consisting of 1444 planes that has been classified by the algorithm. For clarity, the plan view of this map is shown. In figure (2(a)), the color code of blue for “plausible”, red for “suspicious” due to local mapping error and green for “suspicious” due to missing loop closure is used. The robot has travelled through a typical urban environment. The maps in this case were built using scan matching with pose based SLAM algorithms. There are various errors in the final map such as errors due to scan-matching and missed loop closure. It is seen that the algorithm first classifies these inaccuracies in mapping as “suspicious” and then classifies the “suspicious” region caused by missing loop closure as well. Figure (2(b)) highlights the regions that have been labelled as ”plausible”, figure (2(c)) highlights the regions labelled ”suspicious” due to local mapping error and figure (2(d)) highlights the region labelled ”suspicious” due to missed loop closure.

The percentage of correct classification was found to be 79.67%. Table (1) summarize the results achieved in terms of a confusion matrix.

6. CONCLUSIONS

This work addresses the question of how to evaluate the intrinsic quality of maps built by a mobile robot. An CRF is used to model in a probabilistic sense the spatial and temporal compatibility of map components. To classify regions of the map as “plausible” or “suspicious”, a two-tier bipartite segmentation is used. Given that the latter class is indicated by improbable geometry, by considering the temporal compatibility between neighboring components of the map (which in this instance are simply planar patches) we can reason about the likely cause of the poor map quality – loop closure errors or otherwise(scan-matching) [2] . The proposed method is illustrated using maps generated with data gathered by $3D$ laser scanners moving in an urban environment. Future work will involve modifying the algorithm to combine the spatial and temporal compatibility into a single step instead of the current two-tier approach.

7. ACKNOWLEDGMENTS

The authors would like to thank members of the Mobile Robotics Group for their help with the labelling of the laser data.

REFERENCES

Anguelov, Dragomir, Ben Taskar, Vassil Chatalbashev, Daphne Koller, Dinkar Gupta, Geremy Heitz and Andrew Ng (2005). Discriminative learning of markov random fields for segmentation of 3d scan data. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*. Vol. 2. pp. 169–176.

Besag, J (1975). Statistical analysis of non-lattice data. *The Statistician* **24**, 179–195.

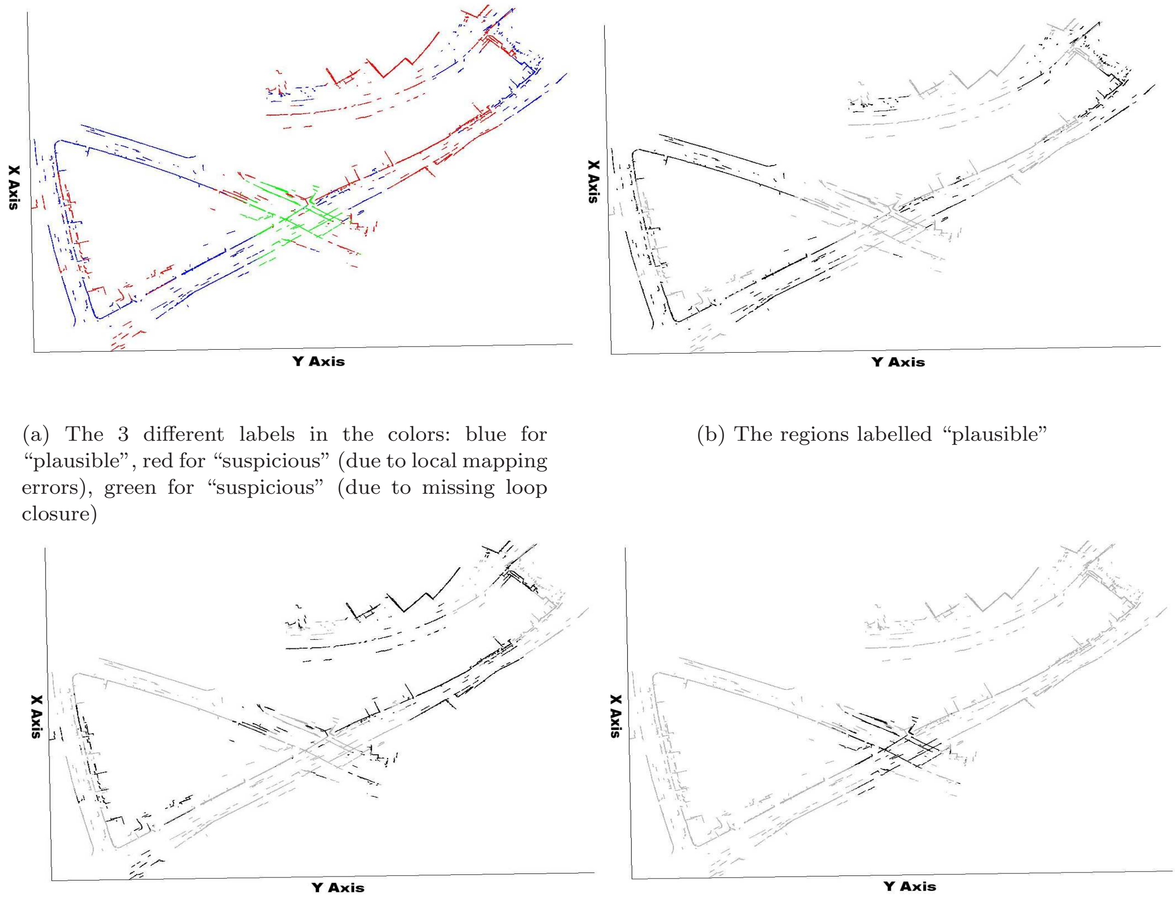
Besag, J. (1986). On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society* **48(3)**, 259–302.

Boykov, Yuri and Vladimir Kolmogorov (2004). An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. and Mach. Intell.* **26**(9), 1124–1137.

Boykov, Yuri, Olga Veksler and Ramin Zabih (2001). Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(11), 1222–1239.

[1] http://cs.ubc.ca/ murphyk/Software/CRF/crf.html

[2] because the maps in this case were built with pose based SLAM algorithm using scan matching



(a) The 3 different labels in the colors: blue for "plausible", red for "suspicious" (due to local mapping errors), green for "suspicious" (due to missing loop closure)

(b) The regions labelled "plausible"

(c) The regions labelled "suspicious" due to local mapping errors

(d) The regions labelled "suspicious" due to missed loop closure

Fig. 2. Plan views of a map consisting of 1444 planes that has been classified by the algorithm.

Truth **Inferred**	**Plausible**	**Suspicious** **(local mapping error)**	**Suspicious** **(missed loop closure)**
Plausible	88.86%	6.19%	4.95%
Suspicious **(local mapping error)**	23.20%	73.63%	4.17%
Suspicious **(missed loop closure)**	14.75%	8.41%	76.84%

Table 1. Confusion Matrix for the map shown in Figure(2) consisting of 1444 planes. The percentage of correct classification was found to be 79.67%.

Chandran-Ramesh, Manjari and Paul Newman (2007). Assessing map quality using conditional random fields. *to be published in Proc. of the International Conf. on Field and Service Robotics.*

Geman, S. and D. Geman (1984). Stochastic relaxation, gibbs distribution and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6(6)**, 721–741.

Kolmogorov, V. and R. Zabin (2004). What energy functions can be minimized via graph cuts?. *IEEE Trans Pattern Analysis and Machine Intelligence* **26**(2), 147–159.

Lafferty, John, Andrew McCallum and Fernando Pereira (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: *Proc. 18th International Conf. on Machine Learning.* Morgan Kaufmann, San Francisco, CA. pp. 282–289.

Liao, Lin, Dieter Fox and Henry Kautz (2007). Extracting places and activities from gps traces using hierarchical conditional random fields. *The International Journal of Robotics Research* **26(1)**, 119–134.

Szeliski, R., R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen and C. Rother (2006). A comparative study of energy minimization methods for markov random fields. *European Conference on Computer Vision* **2**, 16–29.

Triebel, R., K. Kersting and W. Burgard (2006). Robust 3D scan point classification using associative markov networks. In: *Proc. of the International Conference on Robotics and Automation.*

Wallach, Hanna M. (2004). Conditional random fields: An introduction.

Weingarten, J., G. Gruener and R. Siegwart (2003). A fast and robust 3D feature extraction algorithm for structured environment reconstruction. In: *Proc. of the 11th International Conf. on Advanced Robotics.*