2007 International Conference on Cyberworlds

Conditional Random Field for 3D point clouds with Adaptive Data Reduction

E. H. Lim, D. Suter, *Member, IEEE*
Institute for Vision Systems Engineering,
Monash University, Victoria, Australia.
{ eehui.lim | d.suter }@eng.monash.edu.au

***Abstract*—We proposed using Conditional Random Fields with adaptive data reduction for the classification of 3D point clouds acquired from a Riegl Terrestrial laser scanner. The training and inference of the acquired large outdoor urban data can be time consuming. We approach the problem by computing an adaptive support region for each data point using 3D scale theory. For training and inference of the discriminative Conditional Random Fields, smaller set of data samples that contains relevant information within the support region is selected instead of using all point cloud data. We tested the algorithm on synthetically generated data and urban point clouds data acquired from the laser scanner. The computed support region is also used in feature extraction for urban point clouds data. The results showed improvement in the training and inference rate while maintaining comparable classification accuracy.**

***Index Terms*— Classifications, Conditional Random Fields, LIDAR data, machine learning, scale theory**

I. INTRODUCTION

Modelling of urban environment is an important research area, with extensive applications that include regional planning, virtual reality, precise navigation and disaster management. However, the amount of acquired data is usually large and therefore it is memory expensive and require a long time to process the data.

In order to reduce the size of data storage, geometric fitting of the point clouds is recommended. As the point clouds have the following properties: occlusions, varying density, multiple objects and cluttered vegetations - direct geometric model fitting is difficult and challenging. A useful first step will be to classify the raw data points into planar data groups that can be fitted with simple geometry such as a plane. With classification, we will then process different the data types with most appropriate methods.

Many of the previous attempts at urban modelling uses aerial LIDAR data where the acquired data consists of 2D (or 2.5D) bird-eye view point clouds. Methods applied in aerial urban modelling to classify the data are inappropriate for data classification in terrestrial urban modelling. For example, vegetation removal is preformed by filtering via changes in the height differences; however rooftop information is generally unavailable in terrestrial data acquisition. With a Riegl LMS-Z420i terrestrial laser scanner, we need a classification technique that deals with 3D data. In this paper, we performed classifications of terrestrial urban data into object classes using CRF [1].

CRF has been extensively studied for sequential and image segmentation, classification and recognition. For large scale 3D point clouds with extensive training data, a discriminative CRF model is more appropriate than the traditional generative model. However, large training data often requires long period of training process. The need to infer a large amount of data takes a long time as well.

Recognizing this problem, we developed a method that is capable of adaptively reducing the number of classification data. The training and inference of the CRF is based on a reduced data set of the original point clouds. The concept is such that *geometrically similar features* data is omitted from training and inference. With the method, the total processing time required for training and testing can be reduced. The proposed algorithm combines the extraction of distinct features from the point clouds, a discriminative graphical probabilistic model – CRF and 3D scale theory for classification.

We validated our method with synthetically generated data and urban data acquired from a terrestrial Riegl laser scanner. The results show that the training and inference time is greatly reduced while maintaining comparable classification accuracy.

II. BACKGROUND

For 3D point cloud classification, distinct features that capture the relevant relationships among observations are extracted in the first step. In previous works, popular features include intensity [2], height [2-5], surface curvature [5], spin image [3], normals [4], Eigenvalues corresponding to the covariance matrix of k neighbouring data [6] and colour [7]; and these are often combined together or treated independently.

With the extracted features, instead of using a manually fixed threshold [6, 8], supervised-learning models can be trained to recognize which data type the point clouds belonged to. Generative models or discriminative models have been studied for the supervised learning problem. With a generative model, Lalonde et al [9] learned the distribution of the extracted features by fitting a Gaussian Mixture Model using the Expectation Maximization algorithm. Other popular generative models include: Bayes classifier, Hidden Markov Models and Maximum Entropy Markov Models which define a joint probability distribution of the observation and labeling sequences p(X,Y).

Discriminative models such as logistic regression, Conditional Random Fields[1] and Markov Random Fields, specify the probability of a label given an observation

0-7695-3005-2/07 $25.00 © 2007 IEEE
DOI 10.1109/CW.2007.30

sequence p(Y|X). These models have also been employed in supervised image or object classifications problem. Using both generative and discriminative models, Wolf et al. [10] classified 3D points into navigable and non-navigable regions locally with Hidden Markov Models during data acquisition, followed by global segmentation with Markov Random Fields to refine the model afterwards.

Conditional Random Fields [1] are undirected graphical models with which have shown promising results in text processing [1, 11], image segmentation [12, 13], DNA sequence prediction [14] and table or diagram structure extraction from documents [15, 16]. Most applications of CRFs in these cases deal with sequential data or two dimensional images, which are different from 3D point clouds that have irregular grids. The first use of the discriminative graphical model in 3D data that consists of multiple objects is in the Associative Markov Network [3]. In [3] , a fixed number of neighboring points is randomly picked in the experiment: three points are taken randomly in a fixed radius sphere and another three randomly in a fixed radius cylinder to provide vertical restriction.

For fast training and inference in CRFs, recent advances include applying Stochastic Meta-Descent (SMD) which is a stochastic gradient method with gain vector adaptation for accelerated parameter estimation [17]. Cohn [18] reported efficient inference in large CRF data which reduce the time complexity quadratic in the number of labels and number of cliques.

Our approach is similar to data reductions using kd-tree pruning in [19]. Instead of reducing data according to position, our approach identifies and removes redundant data points based on the geometric structure properties such as planarity.

III. MODEL

A. Conditional Random Field

We model 3D point clouds with a CRF [20] such that:

Let $x=x_1, \ldots, x_N$ be the observed feature vectors of some N points. Each feature vectors can consist of a combination of feature descriptors such as heights, colors, SPIN image and estimated normals.

Let $y=y_1,\ldots,y_N$ be the labels in L given the observable point clouds. In urban modeling, labels can be low level such as: 'planar' and 'cluttered'; or higher level such as: 'building', 'vegetation', 'tree trunk', 'grass' and 'man-made pathway'.

The CRF with parameters $\theta = \{\lambda_i, \lambda_{ij}\}$, defines the conditional probability for a label given an input observation to be:

$$P_\theta(y \mid x) = \frac{1}{Z} \prod_{(i,j)\in C} \Psi_{ij}(y_i, y_j, x_i, x_j) \prod_{i=1}^{N} \Psi_i(y_i, x_i) \quad (1)$$

Z is the partition functions which normalized the probability. C is the set of cliques which includes the node and its connecting neighbors. The selection of the neighbors or edges is discussed in the following section.

The node potential Ψ_i is given by:

$$\Psi_i(y_i, x_i) = \exp\{\sum_L (\lambda_i^L x_i) y_i^L\} \quad (2)$$

The edge potential Ψ_{ij} is given by:

$$\Psi_{ij}(y_i, y_j, x_i, x_j) = \exp\{\sum_L (\lambda_{ij} x_i x_j) y_i^L y_j^L\} \quad (3)$$

Where

$$y_i^L = \begin{cases} 1, & \text{label of point } i \text{ is } L \\ 0, & \text{otherwise} \end{cases}$$

A CRF *learns* by finding the node weights $\lambda_i = \{\lambda_i^1, \ldots, \lambda_i^L\}$ and edge weights $\lambda_{ij} = \{\lambda_{ij}^1, \ldots, \lambda_{ij}^L\}$ to maximize the log-likelihood. With a Gaussian prior of variance σ_L^2, the log-likelihood is penalized as follows:

$$L_\theta = \sum_{i=1}^{N} \log P_\theta(y \mid x) - \sum_L \frac{\theta^2}{2\sigma_L^2} \quad (4)$$

where the second summation provides smoothing to avoid over-fitting [21]. The scaled conjugate gradient optimization algorithm is used for the maximization.

Given the observation x, *inference* in CRF is to find a label y_{max} which is the most likely

$$y_{max} = \arg\max_y p_\theta(y \mid x) \quad (5)$$

Since exact inference can be intractable in such models, approximate inference using belief propagation is performed for finding y_{max}.

B. Adaptive Data Reduction

For data reduction, a support region is estimated for every node using 3D scale theory [9, 22]. The size of the support region r is determined adaptively which depends on the curvature, density and standard deviation of noise of the data.

The scalable support region is obtained via iterative procedure explained as follows [22]: With an initial k number of neighboring points (for example, k_o = 15) for point p, the radius r is computed as the distance from p to the further of the kth-nearest neighbors.

With the estimated radius of support region, the density ρ and curvature κ of p is computed:

$$\rho \leftarrow k / \pi r^2 \quad (6)$$

$$\kappa \leftarrow 2d / \mu^2 \quad (7)$$

d is the shortest distance between the point and the least square fitted plane.

A new support region radius is next computed with the known estimation of curvature and density:

$$r = \left(\frac{1}{\kappa}\left(d_1 \frac{\sigma_n}{\sqrt{\varepsilon\rho}} + d_2\sigma_n^2\right)\right)^{\frac{1}{3}} \quad (8)$$

The total number of nearest neighbor within the support region is then computed:

$$k \leftarrow \left[\pi\rho r_{new}^2\right] \quad (9)$$

The process is repeated until k saturates.

σ is the estimated standard deviation of noise which can be obtained experimentally. ε is a small positive number which is set to be 0.01. d_1 and d_2 are small constants that depend on

the sampling distribution of the point clouds. To estimate the value of d_1 and d_2, the following linear system is solved using SVD:

$$\begin{bmatrix} \frac{\sigma_n}{\kappa_0\sqrt{\varepsilon\rho_0}} & \frac{\sigma_n^2}{\kappa_0} \\ \vdots & \vdots \\ \frac{\sigma_n}{\kappa_n\sqrt{\varepsilon\rho_n}} & \frac{\sigma_n^2}{\kappa_n} \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} = \begin{bmatrix} r_0^3 \\ \vdots \\ r_n^3 \end{bmatrix} \quad (10)$$

Refer to [9] for more details. In our experiment, d_1=1 and d_2=4 yield good results.

After the above procedure, for every node, there exists $p = p_1,...p_k$ points within support region r. The p data points within the support region are highly likely to be labeled as the same data class due to the similarity in geometric properties and locations. Therefore instead of using all points in estimating the CRF parameters, a smaller sample of data is selected from p.

The above selection algorithm is explained as follows: A point is randomly picked from the point clouds data and its adaptive support region size and number of nearest neighbours is computed. Within the region, c data points are selected randomly as the neighbors that are connected to the node in the CRF. All points within the support region, regardless of being picked as the neighbors of the node or not, are marked as processed. The process is repeated by picking another point that does not belong to any computed nearest neighbor, until all points have been processed. This is similar to a segmented group of pixels in a segmented image.

In inference, c points are randomly chosen from p using the same algorithm in training point selection. All data points in p will then carry the label of the majority of the connecting neighbors in the CRF.

Therefore, in our CRF, instead of $x=x_1, ...,x_N$, the observed feature vectors will be reduced to $x'=x_1, ...,x_{N'}$, $N'<N$, where N' points are chosen from N. Similarly, the labels vector $y=y_1,...,y_N$ is therefore reduced to $y'=y_1,...,y_{N'}$.

C. Feature Descriptors

Each feature vector consists of a combination of feature descriptors, which are computed from the geometry of data points within the support region.

The three largest eigenvalues of the covariance matrix of the coordinates of $p = p_1,...p_k$ points are computed to form the saliency features [9, 23]. With the relationship between the eigenvalues, these features capture the spatial distribution of points in a local region:

$$\begin{bmatrix} point-ness \\ surface-ness \\ curve-ness \end{bmatrix} = \begin{bmatrix} \log(\lambda_1) \\ \log(\lambda_2 - \lambda_1) \\ \log(\lambda_3 - \lambda_2) \end{bmatrix} \quad (11)$$

Let $\lambda_1>\lambda_2>\lambda_3$ be the eigenvalues of the covariance matrix of the k nearest neighbours. In case of clutter, $\lambda_1\approx\lambda_2\approx\lambda_3$ and there is no dominant direction. For points on surfaces $\lambda_1,\lambda_2>>\lambda_3$ and for linear structures $\lambda_1>>\lambda_2,\lambda_3$.

Another feature descriptor depends on the normals vector, estimated as the eigenvector with the largest eigenvalue in previous computation of saliency features. The angle between the normal vector and the vertical vector is used instead of the normals. This is because the angle is more useful in differentiating terrain and building data, based on the assumption that terrain are mostly flat and walls are mostly vertical.

$$\theta = \arccos(n \bullet [0 \quad 1 \quad 0]) \quad (12)$$

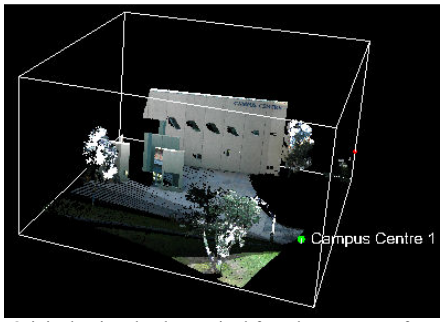

Fig. 3 Original point clouds acquired from laser scanner for testing with color mapped from calibrated camera images (ground truth)

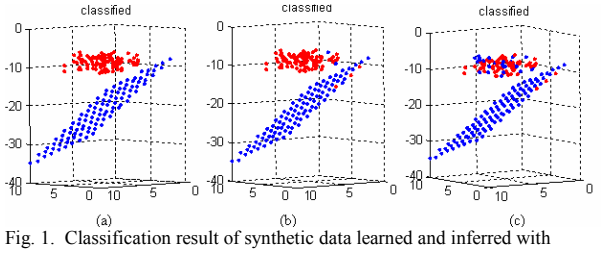

Fig. 1. Classification result of synthetic data learned and inferred with
(a) CRF with adaptive data reduction
(b) CRF without adaptive data reduction
(c) Logistic regression

IV. Experiment

We validated our algorithm with the following synthetic and real-world data:

A. Synthetic data sets

We synthetically generated two clusters of data to evaluate the performance of the proposed algorithm. It consists of two sets (for training and inference purpose) of a hundred points that form a plane with Gaussian noise of σ=0.1; and a hundred point clutter data that represent vegetations. We compare our algorithm with CRF without data reduction, where the connecting neighbors are selected from a fixed support region. We also compare the algorithm with a discriminative classifier that does not take edges potential into account – logistic regression. Comparison between discriminative and generative classifier for urban data can be found in [23].

1) CRF with adaptive data reduction

Six points are randomly selected from the adaptive support region as the edges in CRF. The saliency features are used as the feature vector. From Table I, we can see that the training data is reduced to 42 and it required 12.46s for parameter estimation.

For inference, the testing data is reduced to 52. Most of the data reduction occurs within the plane regions where the curvature is lower and therefore support region is larger (the support region is inversely proportional to the curvature). As it is possible for the support regions to overlap, the final label will be chosen from the majority of the possible label for each point. All points are correctly classified – blue color as plane data's label and red color as cluttered data's label.

2) CRF without adaptive data reduction

For selection of edge points, three points are randomly picked from a fixed radius and another three points randomly picked from a fixed cylinder [3]. As shown in Table I, the time taken to train the CRF with all data is much longer.

For inference, the recognition accuracy is 93.9%. The reason for a few misclassifications shown might be due to the different data selections method. In the proposed algorithm, the selected neighbors are from an adaptive radius; therefore they are highly likely to be from the same class.

3) Logistic regression

$$P_\theta(y \mid x) = \frac{1}{Z} \prod_{i=1}^{N} \Psi_i(y_i, x_i) \qquad (13)$$

This is similar to a CRF but only takes node potentials into account. The time taken for training and testing matches the CRF with adaptive data reduction. However, without spatial information, it is easy to misclassify cluttered region as planar. Therefore the recognition accuracy is only 83.5%. For example, in the computation of saliency features of a cluttered point, it is possible that the *k* nearest points selected to form the covariance matrix are approximately coplanar.

Fig. 2. Riegl LMS-Z420i Terrestrial Laser Scanner equipped with a calibrated Nikon D100 6 Mega Pixel digital camera

B. Urban data sets

With the Riegl terrestrial laser scanner data as shown in Fig.2, training samples can be obtained by projecting labeled 2D images onto 3D point clouds with information of the camera and scanner calibration matrix. This can be done using RiScan Pro, the software accompanied with the Riegl terrestrial laser scanner. The operating range of the scanner is 800m with accuracy of 10mm. The field of view is 80º x 360º and the scanner can be tilted up to 180º.

We performed learning and classification of the data using the proposed algorithm. To estimate the standard deviation of noise in the real-world data, we manually segmented a few clean walls data with different distances from the scanner. We then performed least square plane fitting onto each wall and observed the standard deviation of the noise for each wall. The noise magnitude is different for walls at different distance and the average is estimated to be σ=0.0072.

As a result of data reduction, 57,734 data points for training of the CRF is reduced to 5850 data points. The training data consists of points with almost equal ratio from each class. The feature descriptors consist of the saliency features and the angle between the normal vector and the vertical vector. The time taken for CRF training with four classes is reduced from more than twelve hours to five hours.

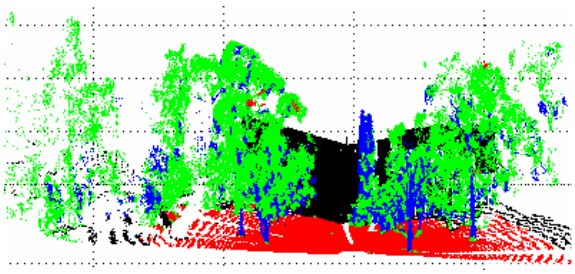

Fig. 5. Classified point clouds – Red represents 'terrain'; black represents 'buildings'; green represents 'vegetations' and blue represents 'tree branches'

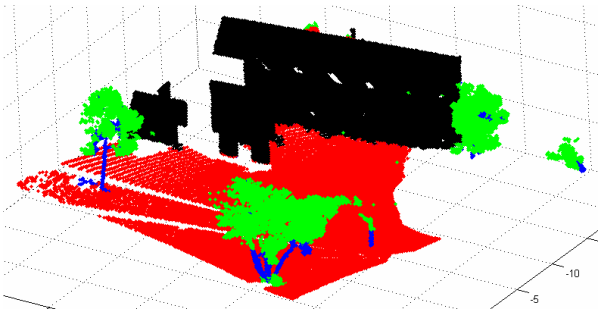

Fig. 4. Classified point clouds – Red represents 'terrain'; black represents 'buildings'; green represents 'vegetations' and blue represents 'tree branches'

For inference, 61,135 point cloud data as shown in Fig.3 is reduced to 5941 via the adaptive data reduction. The point cloud data for testing purpose in Fig.3 is shown with the color mapped from the calibrated camera images. It only required 37.672s for the inference and has classification accuracy of 0.996, as shown in Fig. 4. Most of the misclassifications occur at the trees region, where thin branches that are occluded are recognized as part of the vegetations.

TABLE I
RESULTS FOR SYNTHETIC DATA EXAMPLE

	Accuracy (0-1)	Runtime (Training)	Runtime (Testing)	Training iterations	Reduction to (Train data 200)	Reduction to (Test data 200)
CRF with adaptive point reduction	1	12.46s	0.109s	23	42	52
CRF without adaptive data reduction	0.939	77.96s	0.516s	37	N/A	N/A
Logistic Regression	0.835	11.344	0.156s	18	N/A	N/A

In Fig.5, 424,726 point cloud data is reduced to 24,621 for inference. The misclassifications occurred at the sparse terrain region and are mainly due to the estimation of normals in feature extractions. The slope of the terrain causes the estimated normals of the sparse and incomplete coverage region to be inaccurate. Also, some parts of the dense cylindrical-like vegetation data are labeled as 'trunk'.

In the future, we plan to run the algorithm on data combined from more scan locations so that it will fill out most of the occlusions. We also we plan to try it on a larger number of object classes and on larger data scale. We are also interested in including more feature descriptors such as normalised colours and laser returned intensity to assist in differentiating trunk and vegetation data.

V. CONCLUSION

In this paper, we showed that the training and inference time for the CRF can be reduced with smaller samples of data, while still maintaining comparable classification accuracy. Repeating similar values is not as important for a discriminative graphical model, unlike with a generative model where the priors requires multiple samples to estimate the probability distribution functions. Therefore data reduction is possible for CRF due to the property which is discriminative. It is important to ensure the CRF is trained with a variety of possible values.

REFERENCES

[1] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," *Proc. 18th International Conf. on Machine Learning*, 2001.

[2] L. Matikainen, J. Hyyppa, and H. Hyyppä, "Automatic detection of buildings from laser scanner data for map updating," *ISPRS Commission III. Workshop 3-d reconstruction from airborne laserscanner and InSAR data*, 2003.

[3] D. Anguelov, B. Taskar, V. Chatalbashev, D. Koller, D. Gupta, G. Heitz, and A. Ng, "Discriminative learning of Markov random fields for segmentation of 3D scan data," Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, San Diego, CA, United States, 2005.

[4] F. Rottensteiner, "Automatic generation of high-quality building models from lidar data," *IEEE Computer Graphics and Applications*, vol. 23, pp. 42-50, 2003.

[5] P. Krishnamoorthy, K. L. Boyer, and P. J. Flynn, "Robust detection of buildings in digital surface models," Proceedings 16th International Conference on Pattern Recognition, Quebec City, Que., Canada, 2002.

[6] I. Stamos and P. K. Allen, "Integration of range and image sensing for photo-realistic 3D modeling," Proceedings 2000 ICRA. Millennium Conference. , San Francisco, CA, USA, 2000.

[7] D. D. Lichti, "Spectral filtering and classification of terrestrial laser scanner point clouds," *Photogrammetric Record*, vol. 20, pp. 218-240, 2005.

[8] R. Unnikrishnan and M. Hebert, "Robust extraction of multiple structures from non-uniformly sampled data," IROS 2003, Las Vegas, NV, USA, 2003.

[9] J. F. Lalonde, R. Unnikrishnan, N. Vandapel, and M. Hebert, "Scale selection for classification of point-sampled 3D surfaces," Proceedings. Fifth International Conference on 3-D Digital Imaging and Modeling, Ottawa, Ont., Canada, 2005.

[10] D. F. Wolf, G. S. Sukhatme, D. Fox, and W. Burgard, "Autonomous Terrain Mapping and Classification Using Hidden Markov Models," presented at (ICRA), Proc. of the IEEE International Conference on Robotics and Automation, 2005.

[11] W. Li and A. McCallum, "Rapid development of hindi named entity recognition using conditional random fields and feature induction," *ACM Transactions on Asian Language Information Processing*, vol. 2, pp. 290-294, 2003.

[12] K. Sanjiv and M. Hebert, "Discriminative random fields: a discriminative framework for contextual interaction in classification," Proceedings Ninth IEEE International Conference on Computer Vision, Nice, France, 2003.

[13] X. He, R. S. Zemel, and M. A. Carreira-Perpinan, "Multiscale conditional random fields for image labeling," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, United States, 2004.

[14] D. H. Tran, T. H. Pham, K. Satou, and T. B. Ho, "Conditional random fields for predicting and analyzing histone occupancy, acetylation and methylation areas in DNA sequences," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Budapest, Hungary, 2006.

[15] D. Pinto, A. McCallum, X. Wei, and W. Bruce Croft, "Table Extraction Using Conditional Random Fields," SIGIR Forum (ACM Special Interest Group on Information Retrieval), Toronto, Ont., Canada, 2003.

[16] Y. Qi, M. Szummer, and T. P. Minka, "Diagram structure recognition by Bayesian conditional random fields," CVPR 2005, San Diego, CA, United States, 2005.

[17] S. V. N. Vishwanathan, N. N. Schraudolph, M. W. Schmidt, and K. P. Murphy, "Accelerated training of conditional random fields with stochastic gradient methods," ICML 2006 - Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, United States, 2006.

[18] T. Cohn, "Efficient inference in large conditional random fields," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Berlin, Germany, 2006.

[19] R. Triebel, K. Kersting, and W. Burgard, "Robust 3D scan point classification using associative Markov networks," Proceedings - IEEE International Conference on Robotics and Automation, Orlando, FL, United States, 2006.

[20] C. Sutton and A. McCallum, "An Introduction to Conditional Random Fields for Relational Learning," *Introduction to Statistical Relational Learning. Edited by Lise Getoor and Ben Taskar. MIT Press*, 2006.

[21] S. F. Chen and R. Rosenfeld, "A Gaussian prior for smoothing maximum entropy models," *Technical Report CMUCS-99-108, Carnegie Mellon University*, 1999.

[22] N. J. Mitra, A. Nguyen, and L. J. Guibas, "Estimating surface normals in noisy point cloud data," *Int. J. Comput. Geometry Appl. 14*, vol. (4-5), pp. 261-276, 2004.

[23] E. H. Lim and D. Suter, "Classification of 3d lidar point clouds for urban modelling," *Image and Vision Computing, New Zealand, Nov. 2006*, pp. pages 149-154.