Non-associative Higher-Order Markov Networks for Point Cloud Classification

Mohammad Najafi, Sarah Taghavi Namin,
Mathieu Salzmann, and Lars Petersson

Australian National University (ANU)
NICTA*, Canberra, Australia
{mohammad.najafi,sarah.namin,mathieu.salzmann,lars.petersson}@nicta.com.au

Abstract. In this paper, we introduce a non-associative higher-order graphical model to tackle the problem of semantic labeling of 3D point clouds. For this task, existing higher-order models overlook the relationships between the different classes and simply encourage the nodes in the cliques to have consistent labelings. We address this issue by devising a set of non-associative context patterns that describe higher-order geometric relationships between different class labels within the cliques. To this end, we propose a method to extract informative cliques in 3D point clouds that provide more knowledge about the context of the scene. We evaluate our approach on three challenging outdoor point cloud datasets. Our experiments evidence the benefits of our non-associative higher-order Markov networks over state-of-the-art point cloud labeling techniques.

Keywords: Non-associative Markov networks, Higher-order graphical models, 3D point clouds, Semantic labeling.

1 Introduction

Semantic labeling of 3D point clouds for terrain classification remains a very challenging task, despite recent advances in the field. Outdoor environments are to a large extent irregular in nature and often present complex relationships between the different objects in the scene. Furthermore, the substantial presence of noise in data captured outdoors makes labeling even more difficult. In this paper, we introduce a non-associative higher-order Markov network to address the problem of outdoor terrain classification from 3D point cloud data.

In the past few years, pairwise graphical models have been frequently used for indoor and outdoor point cloud labeling [2,32,28,21,1,19,29]. However, pairwise networks can generally not adequately describe the complex contextual information that exists in natural scenes. In contrast, higher-order networks enable us to better model this information and take into account the structural relationships present between groups of objects in the data. In the context of 3D

* NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the ARC through the ICT Centre of Excellence program.

D. Fleet et al. (Eds.): ECCV 2014, Part V, LNCS 8693, pp. 500–515, 2014.

point cloud classification, a handful of approaches have exploited higher-order models in the form of Associative Markov Networks (AMN) [22,6]. While AMNs consider groups of multiple neighboring nodes jointly, they only encourage these nodes to have an identical label. Therefore, AMNs cannot describe complex relationships between the different classes in the scene and, as a result, have only limited ability to model contextual information. To the best of our knowledge, no model has yet managed to exploit the full representative power of higher-order graphical models for 3D point cloud labeling.

In this paper, we introduce a new higher-order model for 3D point cloud classification that takes into account the non-associative geometric context between different classes. This lets us exploit more information than common pairwise models or associative higher-order models to describe the semantic structure of the scene. As a consequence, our model typically yields more accurate labelings.

More specifically, we build a graph in which each node represents a segment (i.e., group) of 3D points. We then build higher-order cliques by projecting the 3D segments to the ground plane and grouping the segments with substantial overlap. Intuitively, in outdoor scenes, grouping segments along the vertical direction will carry more information than along horizontal ones (e.g., leaves are above tree trunks, which are above the ground). To model this information, we devise four geometric context patterns that describe non-associative relationships between the segments in the cliques. Importantly, these context patterns are independent of the number and size of the segments inside the cliques.

We evaluate our model on three benchmark point cloud datasets (VMR-Oakland-V2, RSE-RSS and GML-PCV). Our approach outperforms state-of-the-art point cloud labeling techniques, which evidences the importance of modeling the complex higher-order relations of the classes in the scene.

2 Related Work

There is a considerable amount of literature on point cloud classification in both indoor and outdoor environments. In particular, over the years, there has been a strong focus on designing new feature types, such as FPFH (Fast Point Feature Histogram) [26], histogram descriptors [3], hierarchical kernel descriptors [4], and on adapting geometric and shape-based features [7,16] to improve the performance of point cloud classification systems. As with RGB images, the performance of local features can typically be improved by exploiting the context of the scene via a graphical model.

Graphical models enable us to encode the spatial and semantic relationships between objects via a set of edges between the nodes in a graph. A number of works [21,2,18,25] have studied the impact of pairwise graphical models on point cloud classification and have demonstrated that adding a label consistency constraint between neighboring nodes improves the classification accuracy significantly. However, these simple label consistency constraints, which define an AMN, often suffer from the drawback of over-smoothing the labeling.

To address this problem, the authors of [28,1,19] investigated the use of pairwise non-AMNs for point cloud labeling. Non-AMNs can exploit the complex contextual information existing between the objects in the scene by exploring various combinations of classes rather than just enforcing homogeneous labelings of the graph nodes. For instance, the observation that A is "above" B cannot be modeled with an AMN, whereas non-AMNs can encode this information. While existing non-AMNs have proven useful for both indoor [1] and outdoor [28] point cloud classification, the current models remain limited to modeling pairwise interactions.

In contrast, higher-order models can be used to capture the complex relationships in the scene that cannot be described using pairwise models [10,11,31,12,30,14]. In our context, in [22,6], Munoz et al. exploited $\mathcal{P}^n$ Potts potentials [10] on groups of multiple 3D points. In [9], a *Voxel-CRF* framework was introduced to tackle the occlusion and 2D-3D mismatch problems by utilizing a higher-order model based on Hough voting and categorical object detection. In both cases, however, the resulting higher-order graphical model is an AMN, and is thus limited to encoding simple label consistency potentials.

The main contribution of our work lies in proposing a non-AMN higher-order graphical model that better describes the scene context and thus yields improved 3D point cloud classification. Our higher-order potentials belong to the category of *pattern-based* potentials [12]. However, in contrast to most instances in this category (e.g., $\mathcal{P}^n$ Potts model, co-occurrence potentials), our potentials account for the geometric context that exists in the scene, and thus form a non-AMN.

Some recent works on point cloud labeling have proposed to incorporate contextual information without using a graphical model [33,8,23]. In particular, in [33], which is the most relevant work here, the authors used a sequence of hierarchical classifiers at different scales (i.e., at point level and at segment level). Due to the non-standard form of their model, they had to design a special inference method. Here, in contrast, we can leverage the vast research on inference in higher-order graphical models to propose a principled approach to point cloud classification.

3 Method

In this section, we introduce our approach to point cloud labeling. To this end, we first present our higher-order CRF. For a comprehensive discussion of CRFs, we refer the reader to [15].

Given N 3D point segments $\mathbf{x} = [x_1, \ldots, x_2, x_N]$ obtained from a point cloud, our goal is to assign a label $y_i \in [1, \cdots, L]$ to each segment x_i. To this end, we construct a Condition Random Field (CRF) over the labels, where each node corresponds to a segment. In this CRF, the joint distribution of the labels of all nodes given the segments can be expressed as

$$\mathbf{P}(\mathbf{y}|\mathbf{x}) = \frac{1}{Z}\exp\left(-\sum_{i=1}^{N}\mathbf{\Phi}(y_i, x_i) - \sum_{(ij)\in\mathcal{E}}\mathbf{\Psi}_{\mathbf{p}}(y_i, y_j, x_i, x_j) - \sum_{c\in C}\mathbf{\Psi}_{\mathbf{h}}(y_c, x_c)\right) \tag{1}$$

where Z is the partition function, $\mathcal{E}$ is the set of second-order (pairwise) edges and C is the set of higher-order cliques in the graph. The unary potential function $\mathbf{\Phi}$ expresses the likelihood of an individual segment to be assigned to each class. The pairwise potential $\mathbf{\Psi_p}$ imposes consistent labeling to the neighboring nodes. In contrast, the clique potential $\mathbf{\Psi_h}$ encodes the compatibility of the different possible class assignments of multiple segments. As will be shown later, we make use of this clique potential to encode the geometric relationships between groups of segments.

To obtain the best labeling for the problem at hand, we seek to compute a MAP estimate of the labels given by $\underset{\mathbf{y}}{arg\,max}\,\mathbf{P}(\mathbf{y}|\mathbf{x})$. This can be achieved by minimizing the energy corresponding to the CRF, given by

$$\mathbf{E}(\mathbf{y}|\mathbf{x}) = \sum_{i=1}^{N} \mathbf{\Phi}(x_i, y_i) + \sum_{(ij)\in\mathcal{E}} \mathbf{\Psi_p}(y_i, y_j, x_i, x_j) + \sum_{c\in C} \mathbf{\Psi_h}(y_c, x_c) \qquad (2)$$

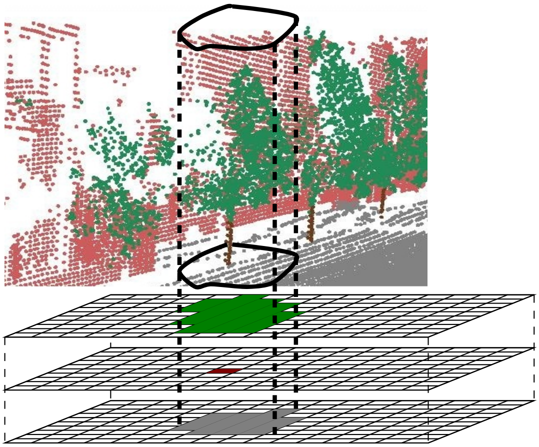
Minimizing this energy is achieved by performing inference in the CRF. To this end, here, we employ Loopy Belief Propagation [24] .

In the remainder of this section, we present the potentials that we use in the energy of Eq. 2. In particular, we introduce new pattern-based potentials that, as opposed to most existing pattern-based potentials, let us model complex geometric relationships across groups of segments.

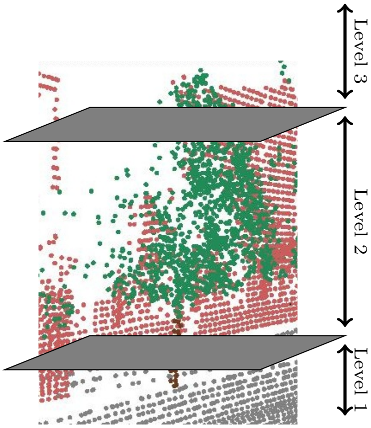
3.1 Higher-Order Context-Based Potentials

Clique Structure. To be able to capture informative semantic context patterns, we construct cliques from segments that are located in the same vertical structures in the point cloud. The intuition behind this is that the horizontal placement of objects in outdoor scenes is often arbitrary (e.g., a car can be located anywhere near a building) and thus conveys less geometric information. In contrast, the relative vertical positioning of objects is often well-constrained (e.g., leaves are above tree-trunks which are above the ground). To build our cliques, we therefore project the segments to the ground plane (which is achieved by removing the z-coordinate of all the points) and find the overlapping segments on this ground plane. More specifically, we create a clique for each segment i and add any segment with a significant overlap with i (i.e., more than 50% overlap) to this clique. Cliques containing a single segment are then discarded. This strategy to create cliques is illustrated in Fig.1-a. While one could think of using a simple grid-based technique to determine the base of the vertical structure of the cliques, in the presence of thin segments such as *tree trunks* and *utility poles*, this approach would be very sensitive to the exact placement of the grid. In contrast, in our scheme all the segments are completely surrounded by at least one clique structure.

Pattern-Based Potentials. As mentioned earlier, in this work we design new pattern-based potentials to encode the geometric relationships within the cliques



a) The vertical structure of a clique in our model. The cliques are created by analyzing every individual segment and checking whether its projection on the ground plane overlaps with the projection of other segments in the point cloud. Here for instance, the projection of the *leaves* covers the *tree trunk* and has a substantial overlap with the *ground*. Hence, a clique from these three segments is formed and our context patterns are extracted from this vertical structure.



c) Height signature pattern. The vertical structure of the clique (shown in Fig.1-a) is cut horizontally into K levels (here $K = 3$). Then each level is explored to check if any of the L class labels is present. The resulting pattern vector for this example is given in Fig.1-d.

Simple Co-occurrence

W	P	G	L	T	B	V
0	0	**1**	**1**	**1**	0	0

b) The simple co-occurrence pattern records the class labels that are found within the vertical structure.

Height Signature (3 levels)

W	P	G	L	T	B	V	W	P	G	L	T	B	V	W	P	G	L	T	B	V
0	0	**1**	0	**1**	0	0	0	0	0	**1**	**1**	0	0	0	0	0	**1**	0	0	0
Level 1							Level 2							Level 3						

d) The height signature pattern shows how the class labels inside the clique are spread vertically. The pattern vector is computed according to Fig. 1-c.

Geometric Co-occurrence

	W	P	G	L	T	B	V
W	0	0	0	0	0	0	0
P	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0
L	0	0	**1**	0	**1**	0	0
T	0	0	**1**	0	0	0	0
B	0	0	0	0	0	0	0
V	0	0	0	0	0	0	0

e) The geometric co-occurrence pattern indicates how the class labels are vertically located inside the clique. Element (i,j) of this matrix is **1** if there is at least one segment with label i, above another segment with label j.

Within Clique Adjacency

	W	P	G	L	T	B	V
W	-	0	0	0	0	0	0
P	-	-	0	0	0	0	0
G	-	-	-	0	**1**	0	0
L	-	-	-	-	**1**	0	0
T	-	-	-	-	-	0	0
B	-	-	-	-	-	-	0
V	-	-	-	-	-	-	-

f) The within clique adjacency indicates which class labels are connected to each other inside the clique.

Fig. 1. Extracting the cliques and the higher-order context patterns from the point cloud. Here, the classes are {W:*wire*, P:*pole*, G:*ground*, L:*leaves*, T:*tree trunk*, B:*building*, V:*vehicle*}.

of our graph. In their general form, pattern-based potentials were introduced by Komodakis and Paragios [12] as potential functions defined as

$$\boldsymbol{\Psi}_{\mathbf{h}}(\mathbf{P}) = \begin{cases} \mathrm{H}(\mathbf{P}) & \mathbf{P} \in \mathcal{P} \\ \mathrm{H}_{\max} & \text{otherwise} \end{cases} \tag{3}$$

where $\mathbf{P}$ is a context pattern vector which describes the clique, $\mathcal{P}$ is the set of all pattern vectors that are considered valid and $\mathrm{H}_{\max}$ is the cost assigned to the patterns that are not listed in $\mathcal{P}$ (i.e., invalid patterns). This formulation is very general and only imposes that $\mathrm{H}_{\max} \geq \mathrm{H}(\mathbf{P})$. However, most existing methods employ such potentials to define simple label consistency constraints, such as $\mathcal{P}^n$ Potts and co-occurrence potentials.

Here, we make use of these potentials to define much more complex relationships between the segments in a clique. In particular, we compute four higher-order patterns defined as $\mathbf{P}_1$: *Simple Co-occurrence*, $\mathbf{P}_2$: *Geometric Co-occurrence*, $\mathbf{P}_3$: *Within Clique Adjacency* and $\mathbf{P}_4$: *Height Signature.* Our complete context pattern is then obtained by concatenating these patterns as

$$\mathbf{P} = [\mathbf{P}_1{}^{\intercal}, \mathbf{P}_2{}^{\intercal}, \mathbf{P}_3{}^{\intercal}, \mathbf{P}_4{}^{\intercal}]^{\intercal}. \tag{4}$$

As will be shown below, the primary advantage of our context patterns is that they are defined based on the class labels of the segments. In other words, we analyze the relationships of the abstract class labels rather than of the specific segments inside the cliques. This property makes our patterns invariant to the size and number of the segments from each class [14]. In our work, to create the set of valid patterns, we make use of the training data and record all the observed context patterns. The collection of observed patterns along with their number of occurrences forms the codebook $\mathcal{P}$. In practice, we ignore cliques of order 6 or higher to keep inference computationally tractable. Furthermore, we take into account all the patterns regardless of their number of occurrences. The intuition is that even patterns that have been observed a small number of times, can be important. We set $\mathrm{H}(\mathbf{P}) = 0$ in Eq. 3, which means that we assign no higher-order cost to the valid patterns. The optimization algorithm then tries to find a labeling of the cliques such that they form valid patterns, while also having low unary and pairwise costs.

In the following, we describe the four different patterns that we employ in more detail.

Simple Co-occurrence. Label co-occurrence is a pattern vector that indicates which classes are present inside a higher-order clique. We represent the co-occurrence pattern by $\mathbf{P}_1 : \{p_1^i\}_{i=1:L}$ which is a binary vector with L elements, where L is the number of class labels. If a segment with class label i is present inside the clique, p_1^i is set to 1 (see Fig. 1-b).

Geometric Co-occurrence. The main drawback of simple co-occurrence is that it just provides us with a symmetric description of the clique and can not capture the geometric relationships between the nodes. For instance, the label configuration of *tree trunk* above *leaves* is undesirable, but the simple co-occurrence pattern vector for this clique will make it a valid configuration. To address this problem, we utilize non-associative features to build a geometric co-occurrence pattern. To this end, we project all the 3D segments onto the ground. Then, for each clique, all segment pairs with a significant projection overlap (larger than 50%) are recorded, and the segment with a higher centroid is considered to be *above* the other one. We encode the *above* relationships between any pair of class labels within the clique as an $L \times L$ binary matrix (Fig. 1-e), which can then form the pattern $\mathbf{P}_2 : \{p_2^i\}_{i=1:L^2}$. Note that, while we compare pairs of segments inside the cliques, the final pattern vector considers all the pairs jointly. Therefore, our geometric co-occurrence potential cannot be expressed as a pairwise potential.

Within Clique Adjacency. To make the context pattern more informative, we check whether there is a spatial connection between any pair of class labels within the clique. Here, we consider that two 3D segments are spatially connected if the shortest Euclidean distance between any two of their points is lower than a pre-defined threshold (in practice 0.6m). This pattern can be stored in the $L(L-1)/2$ dimensional vector $\mathbf{P}_3 : \{p_3^i\}_{i=1:L(L-1)/2}$ (see Fig. 1-f).

Height Signature. This context pattern acts as a vertical location prior in our classification framework. It indicates whether a specific class label is observed in a certain range of height above the ground. To compute this pattern, we partition the point cloud inside each clique into K horizontal levels. At each level, we then record the presence of any of the L classes. This results in the pattern of height signature $\mathbf{P}_4 : \{p_4^i\}_{i=1:LK}$ (see Fig. 1-(c,d)). In practice, we divide the vertical space into $K = 3$ partitions whose boundaries are determined during training.

3.2 Pairwise Potential

In addition to the higher-order terms, we also encode pairwise potentials in our graphical model. In particular, we specify a pairwise link for each pair of 3D segments that are neighbors. Two segments are treated as neighbors if the shortest distance between any two of their points is less than a pre-defined threshold (in practice 0.6 m). We then define a pairwise potential that depends on the class labels of the segments, as well as on their local shape features. This potential can be expressed as

$$\mathbf{\Psi}_{\mathbf{p}}(y_i, y_j, \theta_i, \theta_j) = \begin{cases} \frac{1}{1+|\theta_i - \theta_j|/T} & (y_i \neq y_j) \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where T is a normalization factor set to 90° in practice, and θ is the angle between the direction of the normal vector of the segment and the direction of the vertical axis. Here, the normal vector of a segment is computed by taking the average of the normal vectors of all its points. Intuitively, this potential favors assigning identical labels to two segments if their normal vectors have a similar deviation from the vertical axis.

3.3 Unary Potential

Feature Set. Our unary potential relies on a classifier applied to features extracted at each point of the cloud. In particular, we use the following features: (i) FPFH descriptors that describe the geometric relationships between a point and its neighbors in terms of distance and normal vector orientations [26]; (ii) Eigenvalue features that provide us with measures of scatter, linearity and planarity of a point distribution. (iii) Deviation of the normal vector direction of each point from the z-axis, which helps distinguishing between the horizontal and vertical planar surfaces; (iv) Height of the point.

The FPFH and Eigenvalue features are computed over two local neighborhoods around the point of interest. To obtain the height of each point, a proper estimation of the ground level is essential. As the ground points are not evenly distributed on a horizontal surface, particularly in complex outdoor environments, we perform local approximations of the ground by considering horizontal patches in the point cloud and taking the lowest point as a part of the ground.

Point-Wise Classification. Given the aforementioned features, we employ a probabilistic SVM classifier [20,5] to compute the class probabilities for each 3D point. We then compute the class probability vector of each segment by averaging over the class probabilities of all its constituent 3D points. The unary potential in our graphical model is obtained by taking the negative logarithm of this probability vector. In practice, we used an RBF kernel in our SVM classifier, and set the hyper-parameters of the SVM to $C = 5$ and $\gamma = 0.1$.

3.4 Segmentation

As mentioned throughout this section, we use point segments as nodes in our graphical model. This lets us effectively handle very large point clouds. To obtain these segments, we first apply the efficient fully connected CRF (Dense-CRF) [13] to the results of the point-wise classifier using Gaussian kernels on 3D positions and surface normals (implemented in PCL [27]). This allows us to reduce the noise and produce point classification results that are better suited to segmentation. The final segments are computed by dividing the entire set into L distinct groups, corresponding to the labeling of the Dense-CRF, and clustering each group into smaller segments via k-means clustering. We found that this two-step segmentation scheme yields a cleaner set of segments than directly applying k-means clustering to the point cloud. The number of segments, k, is determined by the k-means algorithm of PCL (about 300 segments in practice).

4 Experiments

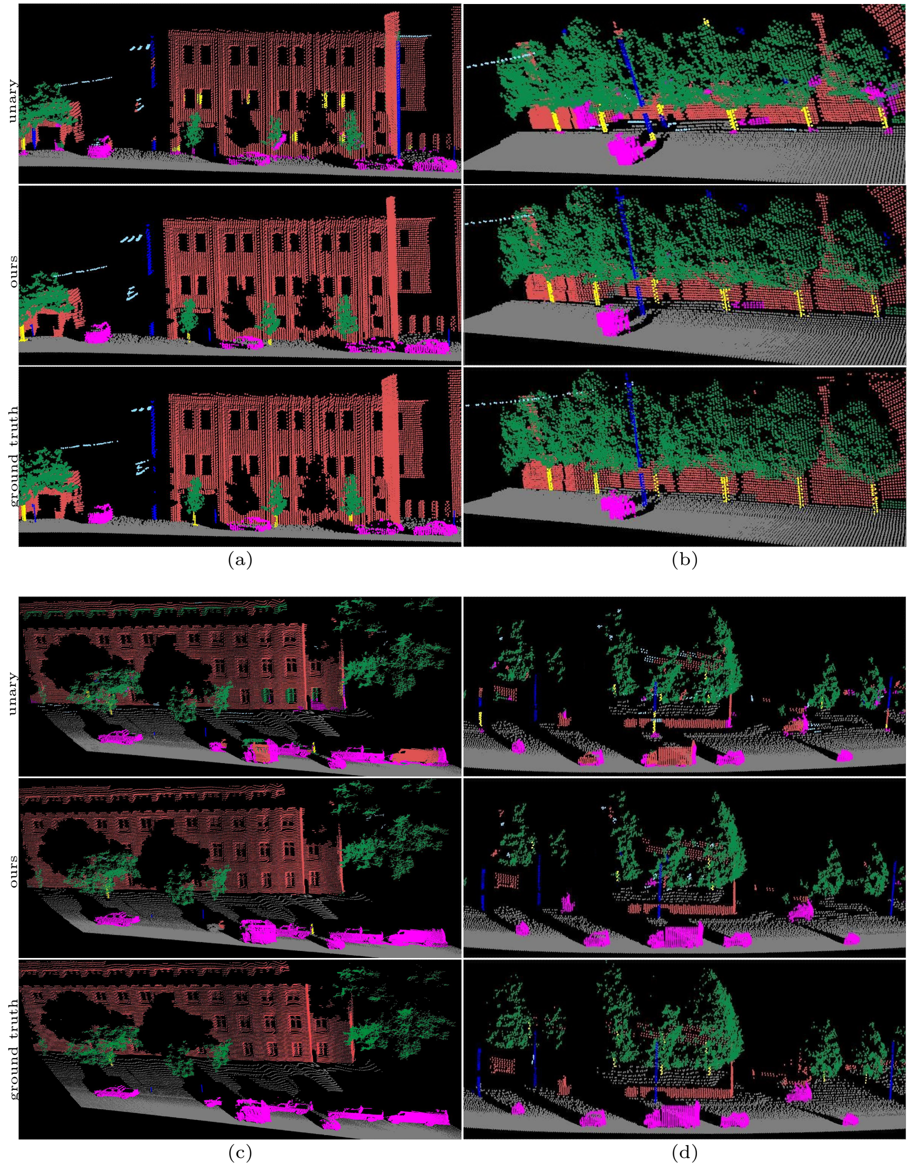
We evaluated the performance of our method using the same three datasets as in [33]. The first dataset, `VMR-Oakland-V2`[1], represents street scenes collected using a terrestrial laser scanner. It is composed of approximately 3 million 3D points separated into 36 point cloud blocks (pcd-files). The points are labeled according to seven categories of outdoor objects, i.e., *wire*, *pole*, *ground*, *leaves*, *tree trunk*, *building* and *vehicle*. The number of points belonging to each class is strongly unbalanced, which makes training very challenging. To facilitate the comparison with previously-reported results on this dataset, we follow the evaluation procedure of [33], which sets aside 6 pcd-files to tune the parameters of the classifier and defines 30 pcd-files to train and test the model. These 30 files are further split into 5 sets, which let us perform 5-fold cross-validation.

Table 1 reports the performance of our approach and of state-of-the-art point cloud labeling baselines in terms of the precision, recall and F1-score ($\text{F1} = \frac{precision \times recall \times 2}{precision + recall}$) for each class. Our approach yields an average F1-score of 0.79, which is higher than the state-of-the-art on this dataset [8]. Note that the performance of the unary potentials is 0.63, which was impressively improved by our non-associative higher-order model. This confirms the importance of our context-aware higher-order potentials. Note that we also computed the F1-scores of the non-associative pairwise model (NA-pairwise) incorporating all our pattern potentials, but computed only on pairwise cliques (formed using our region overlap criterion). This model achieved an average F1-score of 0.73, which shows that, while it yields a better performance than the simple associative pairwise model (0.65), it is outperformed by our higher-order model. In addition, we performed an ablation study in which the results of our model using a single type of higher-order potential at a time were computed. This led to the average F1-scores of 0.74, 0.75, 0.75 and 0.72 for $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$ and $\mathbf{P}_4$, respectively.

Fig. 2 illustrates how our method can improve the results of the unary potential. For a more detailed analysis, we magnified one of the regions of Fig. 2-a in Fig. 3-a. Note that the segment located underneath the tree leaves was originally incorrectly classified as *vehicle* by the unary potentials. Since the pattern {*leaves*-**above & adjacent**-*vehicle*-**above & adj.**-*ground*} does not occur in the codebook $\mathcal{P}$ generated from the training data, it is penalized in our non-associative graphical model. As depicted in Fig. 3, our labeling yields the valid (and correct) pattern {*leaves*-**above & adj.**-*trunk*-**above & adj.**-*ground*}. Fig. 3 illustrates other cases where our non-associative higher-order model has leveraged the geometrical relationships between several clusters in a clique to find the correct labels of the nodes.

Fig. 4 illustrates a failure case of our approach. In this image, the unary potential has classified the top of the building as vegetation. This resulted in the pattern $\mathbf{p}^0$: {*leaves*-**above & adj.**-*building*} which does not exist in the training pattern codebook $\mathcal{P}$. Since the building pillars look very similar to the *tree trunk* class and are beneath and connected to the top segment labeled as leaves,

[1] `http://www.cs.cmu.edu/~vmr/datasets/oakland_3d/`



(a)

(b)

(c)

(d)

Fig. 2. Qualitative results of four different scenes in `VMR-Oakland-V2`. For each scene, we show the results of (top) our unary potentials, (middle) our full model. Ground-truth labels are shown in the bottom image. The classes are colored as {*wire*: white, *pole*: blue, *ground*: gray, *leaves*: green, *tree trunk*: yellow, *building*: brown, *vehicle*: pink}.

Table 1. Classification results for VMR-Oakland-V2. We report the results of: Non-associative higher-order model (*NAHO*, our method), Stacked 3D Parsing (*S3DP*) [33], the efficient inference method of Hu et al. [8], Non-associative pairwise model (NA-pairwise), simple associative pairwise model and our unary potentials.

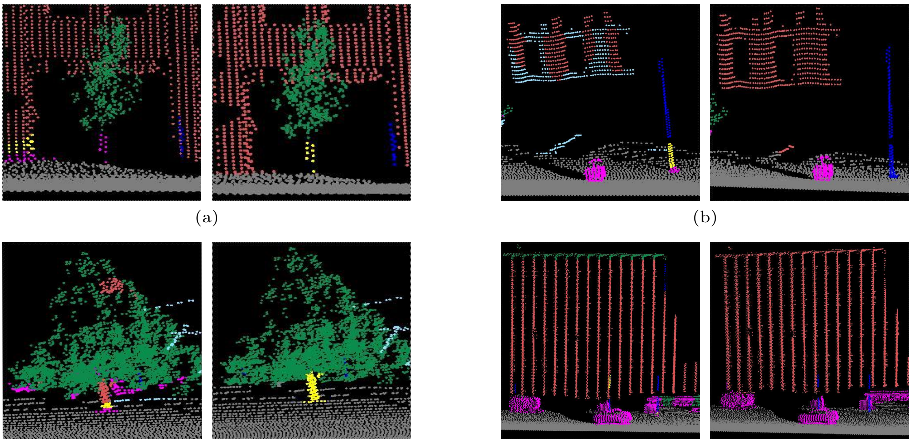
		Wire	Pole	Ground	Leaves	Tree Trunk	Building	Vehicle	avg
Recall	**NAHO (ours)**	**.89**	.56	**.99**	.94	**.49**	.94	**.87**	
	Hu et al. [8]	.61	.62	.98	.95	.30	**.97**	.72	
	S3DP [33]	.75	**.67**	.98	.93	.41	.93	.74	
	NA-Pairwise	.85	.48	**.99**	**.97**	.25	.93	.78	
	Pairwise	.78	.54	.98	.92	.32	.90	.52	
	Unary Potential	.73	.60	**.99**	.91	.38	.89	.49	
Precision	**NAHO (ours)**	.66	.70	**.99**	.95	.52	.91	.75	
	Hu et al. [8]	**.86**	**.72**	.97	**.96**	**.72**	.92	**.85**	
	S3DP [33]	.73	.51	**.99**	**.96**	.65	.83	.79	
	NA-Pairwise	.40	.70	**.99**	.93	.61	**.94**	.76	
	Pairwise	.30	.37	**.99**	.95	.41	.83	.52	
	Unary Potential	.34	.25	**.99**	**.96**	.37	.81	.47	
F1-score	**NAHO (ours)**	**.76**	.62	**.99**	.94	**.50**	.92	**.81**	**.79**
	Hu et al. [8]	.72	**.67**	.98	**.96**	.43	**.94**	.78	.78
	S3DP [33]	.74	.58	.98	.94	**.50**	.88	.76	.76
	NA-Pairwise	.54	.57	**.99**	.95	.35	.93	.77	.73
	Pairwise	.43	.44	.98	.93	.36	.87	.53	.65
	Unary Potential	.46	.35	**.99**	.93	.37	.85	.48	.63

the model matches the pattern $\mathbf{p}^1$: {*leaves*-**above & adj.**-*trunk*} to this pair of segments. In addition, trees with the same height as this building have been observed in the training data, which means that the height signature context is also supporting the undesirable pattern $\mathbf{p}^1$ for this clique. The final decision is thus left to the unary classifier, which due to the similarity of the building pillar to a tree trunk assigns the wrong labels to these segments. A similar situation is shown in Fig. 3-d, where, in contrast, the problem was resolved, thanks to the considerable height of the building pillars.

As a second experiment, we used the GML-PCV[2] dataset. This dataset consists of two separate aerial point clouds A and B, each of which contains about 2M points and is divided into two approximately equally-sized splits for training and test. The object classes present in this dataset are *ground*, *building*, *vehicle*, *bushes/low vegetation* and *trees/high vegetation*. Due to the lack of samples from the vehicles class in dataset B, this class is commonly dropped from the evaluation procedure. Table 2 provides the results of our approach and state-of-the-art baselines on this dataset. Note that, as before, our system outperforms the state of the art ([33]) on this dataset.

GML-PCV is probably the most challenging dataset in our study, due to the presence of many steep slopes and hills, which incur large variations of the ground height. This issue adversely affects our context patterns that are extracted from the clique structures. To address this problem, we performed ground estimation in small patches of 5 m × 5 m. Furthermore, note that this aerial data provides us with a bird's eye view of the scenes which yields much fewer informative vertical

[2] http://graphics.cs.msu.ru/en/node/922



(a) (b)

(c) (d)

Fig. 3. Examples of misclassifications of the unary potentials (left image) which are fixed using our higher-order model (right image). Context pattern vectors that are not found in the pattern codebook are penalized and thus corrected by our approach. The classes are color-coded as in Fig. 2.

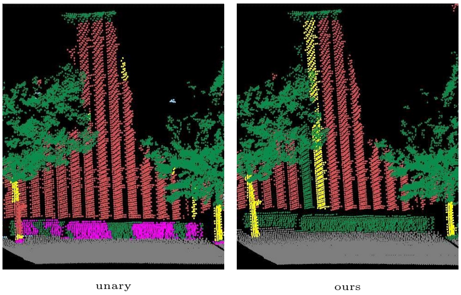


Fig. 4. Example where context was not sufficient to correct the unary results. The presence of leaves on top of the building in conjunction with the similarity of the building pillars to the class of *tree trunk* has caused the higher-order model to consider this scene as *leaves*-above-*trunk*. Note that some other regions of this point cloud were corrected by our model. Class labels are color-coded as in Fig. 2.

patterns. Therefore, most of the extracted cliques contain only two segments. Nonetheless, our approach managed to extract the relevant information from the data (e.g., *height signature*) to overcome these problems. Qualitative results on this dataset are depicted in Fig. 5-a, where the non-associative higher-order model was able to recover some of the buildings and disambiguate low-vegetation from high-vegetation in some regions.

Finally, we evaluated our model on the RSE-RSS[3] dataset [17], which contains 10 blocks of point clouds from urban scenes, captured using a terrestrial LIDAR

[3] http://www.cs.washington.edu/homes/kevinlai/datasets.html

Table 2. Classification results for the dataset `GML-PCV` using different approaches: Non-associative higher-order model (*NAHO*, our method), Stacked 3D Parsing (*S3DP*) [33], non-associative pairwise model (NA-pairwise) and Unary Potentials.

Dataset A		**Ground**	**Building**	**Vehicle**	**High Veg**	**Low Veg**	**avg**
Recall	**NAHO (ours)**	.94	.72	.38	.97	.72	
	S3DP [33]	**.98**	**.77**	.10	**.98**	.36	
	NA-pairwise	.93	.70	.37	.97	**.74**	
	Unary Potential	.90	.73	**.40**	.96	.73	
Precision	**NAHO (ours)**	.97	.81	.42	.98	.17	
	S3DP [33]	.95	**.91**	**.54**	**.99**	**.31**	
	NA-pairwise	.96	.76	.41	.98	.17	
	Unary Potential	**.98**	.49	.40	**.99**	.13	
F1-score	**NAHO (ours)**	.95	.76	**.40**	**.98**	.28	**.67**
	S3DP [33]	**.96**	**.83**	.17	**.98**	**.33**	.66
	NA-pairwise	.94	.73	.39	.97	.28	.66
	Unary Potential	.94	.59	**.40**	.97	.22	.62

Dataset B		**Ground**	**Building**	**High Veg**	**Low Veg**	**avg**
Recall	**NAHO (ours)**	**.99**	**.93**	**.97**	**.55**	
	S3DP [33]	**.99**	.92	**.97**	.52	
	NA-pairwise	.99	.83	.93	.54	
	Unary Potential	**.99**	.77	.96	.37	
Precision	**NAHO (ours)**	**.99**	**.91**	**.97**	**.57**	
	S3DP [33]	**.99**	.83	**.97**	.53	
	NA-pairwise	.99	.86	.97	.51	
	Unary Potential	.98	.90	.94	.40	
F1-score	**NAHO (ours)**	**.99**	**.92**	**.97**	**.56**	**.86**
	S3DP [33]	**.99**	.87	**.97**	.52	.84
	NA-pairwise	.99	.84	.95	.52	.83
	Unary Potential	.98	.83	.95	.38	.79

Table 3. Classification results for the dataset `RSE-RSS` using different approaches: Non-associative higher-order model (*NAHO*, our method), Stacked 3D Parsing (*S3DP*) [33], non-associative pairwise model (NA-pairwise) and Unary Potentials.

		Background	**Street Sign**	**Ground**	**Tree**	**House**	**Fence**	**Person**	**Vehicle**	**avg**
Recall	**NAHO (ours)**	.81	**.51**	**.93**	**.75**	.81	**.61**	.39	**.49**	
	NA-pairwise	**.83**	.25	**.93**	.74	.81	.27	.44	.43	
	Unary Potential	.78	.41	.92	.69	**.82**	.51	**.57**	.43	
Precision	**NAHO (ours)**	.96	**.12**	.91	**.68**	**.88**	.27	.18	**.44**	
	NA-pairwise	.92	.04	.92	.66	.86	**.40**	**.25**	.41	
	Unary Potential	**.97**	.07	**.93**	.67	.82	.32	.10	.40	
F1-score	**NAHO (ours)**	**.88**	.19	.92	**.71**	**.84**	.37	.25	.46	**.58**
	S3DP [33]	.79	**.28**	**.94**	.66	.83	.31	.20	**.49**	.56
	NA-pairwise	.87	.07	.92	.70	.83	.32	**.32**	.42	.56
	Unary Potential	.86	.12	.92	.68	.82	**.39**	.17	.41	.54

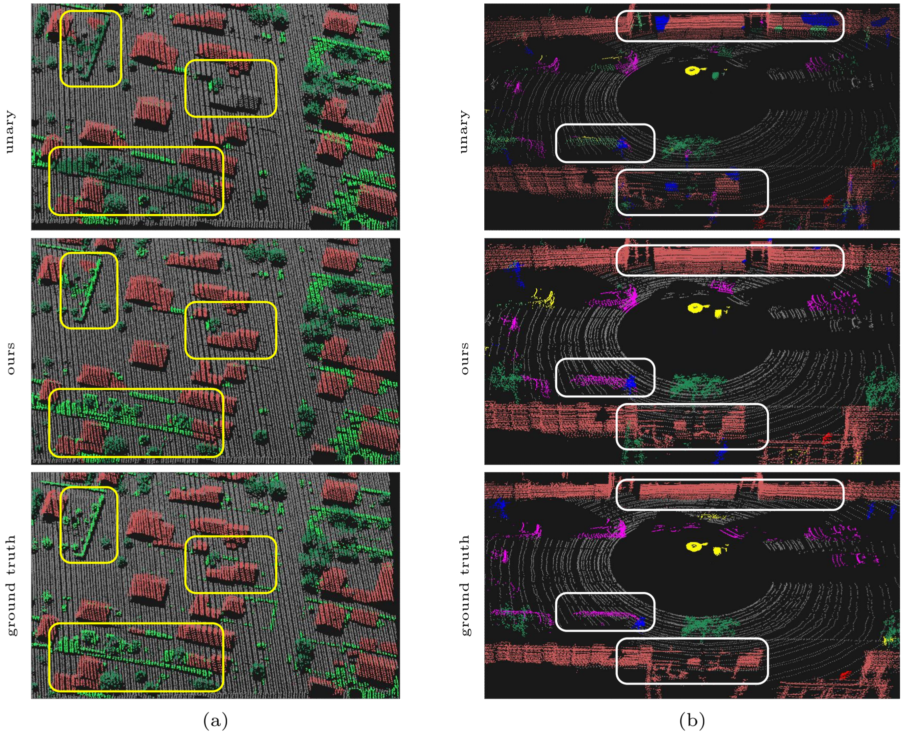


Fig. 5. Qualitative results of two scenes from `GML-PCV` (a) and `RSE-RSS` (b). For each scene, we show the results of (top) our unary potentials, and (middle) our full model. Ground-truth is shown in the bottom image. The highlighted frames indicate the regions whose labels were corrected using our model. Color codes for (a): {*ground*: gray, *building*: brown, *high-vegetation*: dark green, *low-vegetation*: bright green}, and for (b): {*background*: yellow, *street signs*: blue, *ground*: gray, *tree*: green, *building*: brown, *person*: red, *vehicle*: pink}.

scanner. The dataset is composed of 3D points from eight object categories: *street sign*, *ground*, *tree*, *building*, *fence*, *person*, *vehicle* and *background*, which includes every object not belonging to the previous classes. Table 3 reports the performance of our method obtained using the evaluation procedure of [33].

As discussed in [33], it is very difficult to record descriptive context patterns from this dataset. Nevertheless, as depicted in Table 3, our higher-order model has improved the F1-scores of the unary classifier significantly. This improvement is mostly noticeable in the classes *street sign*, *tree*, *person* and *vehicle*. One reason behind this improvement could be the size of these objects and the fact that they are more likely to be included in clique structures (Fig. 1) with descriptive context information. Fig. 5-b provides qualitative results of our method on one scene of this dataset.

5 Conclusion

In this paper, we have introduced a non-associative higher-order CRF to address the problem of semantic 3D point classification. In contrast to many conventional higher-order models, which simply favor identical labeling of the nodes inside the cliques, our model accounts for complex relationships between the different class labels. To model such contextual information we have introduced a set of new higher-order pattern-based potentials. We have evaluated our method on three challenging outdoor point cloud datasets and achieved superior results compared to state-of-the-art techniques. This indicates the importance of exploiting non-associative higher-order models to encode the geometric relationships between objects in outdoor scenes. In the future, we intend to study how such non-associative potentials can be applied to RGB image semantic labeling.

References

1. Anand, A., Koppula, H., Joachims, T., Saxena, A.: Contextually guided semantic labeling and search for 3d point clouds. IJRR (2012)
2. Anguelov, D., Taskar, B., Chatalbashev, V., Koller, D., Gupta, D., Heitz, G., Ng, A.: Discriminative learning of markov random fields for segmentation of 3d scan data. In: CVPR, pp. 169–176 (2005)
3. Behley, J., Steinhage, V., Cremers, A.B.: Performance of histogram descriptors for the classification of 3d laser range data in urban environments. In: ICRA, pp. 4391–4398 (2012)
4. Bo, L., Lai, K., Ren, X., Fox, D.: Object recognition with hierarchical kernel descriptors. In: CVPR, pp. 1729–1736 (2011)
5. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM TIST 2, 27:1–27:27 (2011)
6. Daniel Munoz, N.V., Hebert, M.: Onboard contextual classification of 3-d point clouds with learned high-order markov random fields. In: ICRA (2009)
7. Gould, S., Baumstarck, P., Quigley, M., Ng, A., Koller, D.: Integrating visual and range data for robotic object detection. In: ECCV Workshop (2008)
8. Hu, H., Munoz, D., Bagnell, J.A., Hebert, M.: Efficient 3-d scene analysis from streaming data. In: ICRA, pp. 2297–2304 (2013)
9. Kim, B.S., Kohli, P., Savarese, S.: 3d scene understanding by voxel-crf. In: ICCV (2013)
10. Kohli, P., Kumar, M., Torr, P.: P3 and beyond: Move making algorithms for solving higher order functions. PAMI 31(9), 1645–1656 (2009)
11. Kohli, P., Ladicky, L., Torr, P.: Robust higher order potentials for enforcing label consistency. IJCV 82(3) (2009)
12. Komodakis, N., Paragios, N.: Beyond pairwise energies: Efficient optimization for higher-order mrfs. In: CVPR, pp. 2985–2992 (2009)
13. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. In: NIPS, pp. 109–117 (2011)
14. Ladicky, L., Russell, C., Kohli, P., Torr, P.H.S.: Inference methods for crfs with co-occurrence statistics. IJCV 103(2), 213–225 (2013)
15. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: ICML, pp. 282–289 (2001)

16. Lai, K., Bo, L., Ren, X., Fox, D.: Sparse distance learning for object recognition combining rgb and depth information. In: ICRA, pp. 4007–4013 (2011)
17. Lai, K., Fox, D.: Object recognition in 3d point clouds using web data and domain adaptation. IJRR 29(8), 1019–1037 (2010)
18. Lim, E.H., Suter, D.: 3d terrestrial lidar classifications with super-voxels and multi-scale conditional random fields. CAD 41(10), 701–710 (2009)
19. Lin, D., Fidler, S., Urtasun, R.: Holistic scene understanding for 3d object detection with rgbd cameras. In: ICCV (2013)
20. Lin, H.T., Lin, C.J., Weng, R.C.: A note on platt's probabilistic outputs for support vector machines. ML 68(3), 267–276 (2007)
21. Lu, Y., Rasmussen, C.: Simplified markov random fields for efficient semantic labeling of 3d point clouds. In: IROS, pp. 2690–2697 (2012)
22. Munoz, D., Bagnell, J.A., Vandapel, N., Hebert, M.: Contextual classification with functional max-margin markov networks. In: CVPR, pp. 975–982 (2009)
23. Munoz, D., Bagnell, J.A., Hebert, M.: Co-inference for multi-modal scene analysis. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VI. LNCS, vol. 7577, pp. 668–681. Springer, Heidelberg (2012)
24. Murphy, K.P., Weiss, Y., Jordan, M.I.: Loopy belief propagation for approximate inference: An empirical study. In: UAI, pp. 467–475 (1999)
25. Niemeyer, J., Rottensteiner, F., Soergel, U.: Contextual classification of lidar data and building object detection in urban areas. ISPRS JPRS 87, 152–165 (2014)
26. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: ICRA, pp. 1848–1853 (2009)
27. Rusu, R.B., Cousins, S.: 3D is here: Point Cloud Library (PCL). In: ICRA (2011)
28. Shapovalov, R., Velizhev, A., Barinova, O.: Non-associative markov networks for 3d point cloud classification. In: PCV, vol. 38, pp. 103–108 (2010)
29. Shapovalov, R., Vetrov, D., Kohli, P.: Spatial inference machines. In: CVPR, pp. 2985–2992 (2013)
30. Vineet, V., Warrell, J., Torr, P.H.S.: Filter-based mean-field inference for random fields with higher-order terms and product label-spaces. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part V. LNCS, vol. 7576, pp. 31–44. Springer, Heidelberg (2012)
31. Wegner, J.D., Montoya-Zegarra, J.A., Schindler, K.: A higher-order crf model for road network extraction. In: CVPR, pp. 1698–1705 (2013)
32. Xiong, X., Huber, D.: Using context to create semantic 3d models of indoor environments. In: BMVC. pp. 45.1–45.11 (2010)
33. Xiong, X., Munoz, D., Bagnell, J.A.D., Hebert, M.: 3-d scene analysis via sequenced predictions over points and regions. In: ICRA (2011)