Hierarchical Higher-order Regression Forest Fields: An Application to 3D Indoor Scene Labelling

Trung T. Pham, Ian Reid, Yasir Latif
School of Computer Science
The University of Adelaide
{trung.pham,ian.reid,yasir.latif}@adelaide.edu.au

Stephen Gould
Research School of Computer Science
The Australian National University
stephen.gould@anu.edu.au

Abstract

This paper addresses the problem of semantic segmentation of 3D indoor scenes reconstructed from RGB-D images. Traditionally label prediction for 3D points is tackled by employing graphical models that capture scene features and complex relations between different class labels. However, the existing work is restricted to pairwise conditional random fields, which are insufficient when encoding rich scene context. In this work we propose models with higher-order potentials to describe complex relational information from the 3D scenes. Specifically, we relax the labelling problem to a regression, and generalize the higher-order associative P^n *Potts model to a new family of* arbitrary *higher-order models based on regression forests. We show that these models, like the robust P^n models, can still be decomposed into the sum of pairwise terms by introducing auxiliary variables. Moreover, our proposed higher-order models also permit extension to hierarchical random fields, which allows for the integration of scene context and features computed at different scales. Our potential functions are constructed based on regression forests encoding Gaussian densities that admit efficient inference. The parameters of our model are learned from training data using a structured learning approach. Results on two datasets show clear improvements over current state-of-the-art methods.*

1. Introduction

In recent years significant progress has been made in Structure from Motion (SfM) and Visual Simultaneous Localisation and Mapping (VSLAM) and it is now possible to perform real-time, dense 3D reconstruction in an indoor environment using either a single camera [18] or RGB-D sensor [17]. However, even such dense representations are simply sets of 3D points encoding the scene geometry, and are often inadequate for high-level applications such as mobile robot navigation or manipulation [2]. In this paper we aim to generate more meaningful representations of 3D scenes in terms of semantic regions and objects, rather than pure 3D point clouds.

Although there has been remarkable progress in 2D image semantic segmentation, 3D scene semantic labelling has not received as much attention or achievement (though some notable exceptions exist, *e.g.*, [12, 8]). Unlike 2D images which capture specific views, 3D reconstructed point clouds cover the whole scene with a large number of things and stuff, making label prediction more challenging. Yet 3D data inherently carries rich contextual information beneficial for the label prediction task if successfully exploited. Koppula et al. [12] proposed a graphical model with sophisticated pairwise potentials to encode contextual relations between different objects for 3D indoor scene understanding. The model parameters are learned from training data using a Structured Support Vector Machine (SSVM) approach. Kahler and Reid [8] learn similar potential functions using Decision Tree Fields (DTF) [19] and Regression Tree Fields (RTF) [7]. The tree based method [8] is more efficient and yields similar segmentation accuracies. Although initial promising results have been reported, both of these models restrict themselves to pairwise potential functions limiting their ability to encode complex geometrical and topological characteristics of 3D scenes.

We propose to overcome the above-mentioned limitations of previous work by introducing a novel higher-order model for semantic 3D indoor scene labelling. Our higher-order potentials can capture much more structural information embedded in the scene than the common pairwise potentials, thus effectively yield more accurate label predictions. Moreover, we extend the proposed higher-order models to hierarchical models that allow the incorporation of features and contextual information evaluated at different scales for even better performance.

Specifically, we generalize the popular higher-order *associative* P^n Potts model [9] and its robustified version [10] to a new family of *arbitrary* higher-order models. Similar to the robust P^n models, our higher-order potentials can be decomposed into the sum of pairwise terms by introducing auxiliary variables, thus admitting efficient inference and learning. The main difference is that rather than penalizing the count of variables (within a clique) that disagree with the dominant label as in the Potts models, our higher-order potentials employ more arbitrary penalty functions to

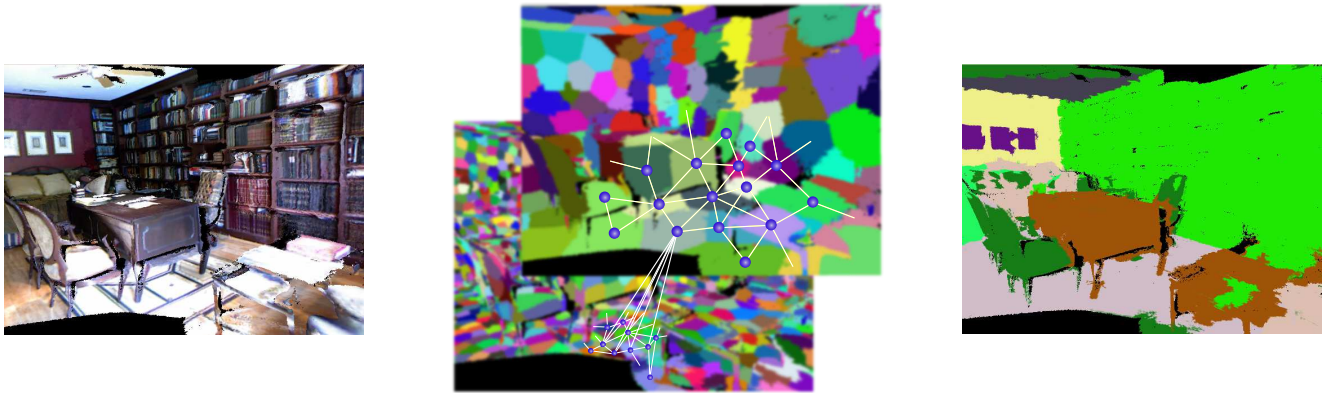


Figure 1: An overview of our 3D scene semantic labelling approach. Given a 3D point cloud (left) reconstructed from RGB-D images, we decompose the scene into layers (*e.g.* segments, super-segments), construct a hierarchical graph (middle), and finally predict a category label for each segment (right) using a hierarchical higher-order regression trees based CRF model. For clarity, we only show non-overlapping super-segments, in practice super-segments could overlap each others.

describe complex relations. The main drawback is that the proposed functions are no longer submodular and cannot be optimised efficiently using graph-cuts. To overcome this difficulty we use Gaussian functions over encoded continuous vector-valued labels, which are particularly convenient to optimise (*i.e.* find the mean/mode of the Gaussian). Importantly, the Gaussian parameters (*i.e.*, mean and covariance matrix) are drawn from learned regression trees conditioned on the data, resulting in expressive potentials unlike standard Gaussian CRFs [26]. Inspired by the Regression Tree Fields recently proposed in [7], we learn regression trees and associate different Gaussian parameters with each leaf node. Conditioned on the clique data, active Gaussian parameters for the higher-order potential functions are selected via decision rules. Both the tree structures and the Gaussian parameters at the leaves can be effectively learned from training data.

Another important contribution of our work is a novel non-parametric hierarchical CRF model for 3D scene semantic labelling that allows us to integrate scene context as well as features at different scales. To do so, we extend our proposed higher-order regression forest CRF by introducing hierarchical connections between auxiliary variables. More specifically, we first oversegment a 3D point cloud into hierarchy of layers—a segment (*i.e.*, groups of 3D points) layer, a super-segments (*i.e.*, groups of segments) layer, and so on. We then build a hierarchical graph in which base-layer nodes and cliques represent segments and groups of segments, respectively, second-layer nodes correspond to super-segments, etc. Fig. 1 shows an overview of our 3D scene labelling pipeline. We empirically evaluate our proposed hierarchical higher-order model using two 3D indoor scene datasets, namely Cornell-RGBD [12] and NYU Depth [25]. The experimental results show that our models, which better exploit the scene context, greatly outperform the standard pairwise CRF models.

2. Related Work

Recently, there has been considerable efforts devoted to automatic labelling of 3D point clouds acquired from laser scans or RGB-D sensors (Kinect) for both outdoor [23, 28, 16] and indoor [27, 12, 8] environments; though these efforts are not yet as well developed as their 2D image counterparts. Similar to standard (2D) semantic image segmentation, the 3D point cloud labelling task aims at assigning a category label to each 3D point in the cloud. A common solution is to train a classifier (*e.g.*, random forest or support vector machine) to assign a semantic label to each point independently. This line of work usually requires rich feature descriptors extracted at every point [22, 4]. Nevertheless when the features are not sufficiently discriminative to predict labels correctly, exploiting contextual information of 3D scenes can significantly improve performance [3, 14, 12, 8]. Such contextual priors are commonly encoded using conditional Markov random fields (CRFs).

A number of 3D scene understanding methods make the assumption that neighbouring points should belong to the same object. Such simple smoothness priors can be encoded using pairwise associative CRFs [3, 14]. Though improvements have been observed, associative potentials are clearly insufficient to encode complex relations between different classes in 3D data, for example, sky *above* grass, computers *on top of* tables. Moreover, associative potentials tend to over-smooth the labelling. To address these issues, non-associative pairwise graphical models have been proposed [24, 23]. However, as shown in [12], models with both associative and non-associative terms are more appropriate since not every relation is non-associative. As a consequence, the authors introduced a parsimonious model with coupled associate and non-associative pairwise potentials. The model is learned using a Structured Support Vector Machine approach. In [8], similar pairwise potentials are trained using Decision Tree Fields [19] and Regression Tree Fields [7]. While these models have proven useful for

indoor scene semantic segmentation, they are restricted to pairwise potentials which are in many cases cannot encode complex geometrical and topological arrangements between different objects in 3D scenes. We address this inadequacy by proposing CRF models with sophisticated higher-order potentials.

Higher-order potentials have shown their superiority in many (image) labelling problems [10, 11, 15, 21]. Consider, for example, the robust P^n Potts models [10], which enforces label consistency over image regions (*i.e*. superpixels) and has been shown to outperform standard pairwise smoothness potentials. Nevertheless, such higher-order smoothness potentials fail to encode complex non-associative relations between labels, and thus are restrictive for our problem. Our work extends the robust P^n models to a new class of rich, arbitrary (both associative and non-associative depending on data content) potentials for 3D indoor scene semantic labelling. Although pattern based higher-order potentials (*e.g*. [11, 15]) allow complex relations to be encoded, these potential are only applicable to fixed-size cliques, thus less flexible than ours.

A few works consider higher-order potentials for point cloud labelling. For instance Najafi et al. [16] introduce a non-associative higher-order Markov network for classifying 3D point clouds arising from laser scans. The authors introduced various pattern-based higher-order potentials such as simple label co-occurrences, geometric co-occurrences to encode scene context. The patterns are learned from a large amount of training data. As expected, their higher-order models yield improved point cloud classification as compared to the pairwise counterparts. However, since inference is solved using loopy belief propagation algorithm, their method does not allow large cliques (ignoring cliques of order six or higher) and has no convergence guarantees. Moreover, the model parameters are learned by a cross-validation strategy, which is not effective for higher-order terms as shown in [20]. In contrast, our models allow arbitrary clique sizes, and all the parameters are effectively learned using a structured learning approach.

3. 3D Indoor Scene Semantic Labelling

Given a sequence of RGB-D images recorded from a Kinect sensor, our system starts by reconstructing a 3D dense volumetric representation of the scene. It then extracts features and finally semantically labels the 3D points. Except for the final label prediction, all pre-processing steps in our model are similar to previous works [12][8]. We therefore focus on describing the label prediction algorithm in the next sections, and defer discussion of the 3D reconstruction and feature extraction until Sec. 5.

Assume we are given a reconstructed point cloud $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_N\}$. Our goal is to assign a semantic label $\boldsymbol{y}_i$ (*e.g*. table, computer, etc.) to each 3D point $\boldsymbol{x}_i$. Note that the semantic labels $\boldsymbol{y}_i$ can be encoded using either discrete values or continuous vectors [7]. A typical assignment is denoted as $\boldsymbol{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_N\}$. The "best" labelling can be computed by optimising an energy function, which relates observations (*i.e*., 3D points) and labels

$$\boldsymbol{Y}^* = \arg\min_{\boldsymbol{Y}} E(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}), \tag{1}$$

where $\mathcal{W}$ are the model parameters. To this end, one needs to define an appropriate energy function E, an efficient algorithm for inference as well as a training algorithm to learn the model parameters $\mathcal{W}$. Next we will describe these tasks in detail.

4. Random Fields For Label Prediction

Conditional Markov random fields (CRFs) are a class of structured prediction model that is well suited for predicting labels in point clouds and images. Due to computational concerns during learning and inference, pairwise CRFs are commonly used in practice. Here the energy function for the CRF includes unary and pairwise terms only:

$$\begin{aligned} E(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) = &\sum_{i \in \mathcal{V}} \psi_u^i(\boldsymbol{y}_i, \boldsymbol{X}, \mathcal{W}) \\ &+ \sum_{(i,j) \in \mathcal{E}} \psi_p^{ij}(\boldsymbol{y}_i, \boldsymbol{y}_j, \boldsymbol{X}, \mathcal{W}), \end{aligned} \tag{2}$$

where $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ is a graph with a set of vertices $\mathcal{V}$ indexing random variables and a set of edges $\mathcal{E}$ connecting pairs of variables (often only those in a local neighbourhood). The functions $\psi_u^i(.)$ and $\psi_p^{ij}(.)$ are unary and pairwise potentials, respectively. The unary term usually represents the negative log-likelihood of independently assigning label $\boldsymbol{y}_i$ to node $\boldsymbol{x}_i$, while the pairwise term models prior knowledge of the underlying scene, *e.g*., label smoothness. Pairwise models are generally inadequate at capturing complex scene context (*e.g*., higher-dependencies between multiple nodes) and in many cases can result in poor quality segmentations. Our main contribution in this work is in introducing a new class of higher-order potential to model rich contextual information existing in 3D scenes, which can still be optimised efficiently.

4.1. Higher-order Potentials

A higher-order CRF energy can be written as:

$$\begin{aligned} E(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) = &\sum_{i \in \mathcal{V}} \psi_u^i(\boldsymbol{y}_i, \boldsymbol{X}, \mathcal{W}) \\ &+ \sum_{(i,j) \in \mathcal{E}} \psi_p^{ij}(\boldsymbol{y}_i, \boldsymbol{y}_j, \boldsymbol{x}, \mathcal{W}) + \sum_{c \in \mathcal{C}} \psi_h^c(\boldsymbol{y}_c, \boldsymbol{X}, \mathcal{W}), \end{aligned} \tag{3}$$

where $\mathcal{C}$ is a set of cliques; $\boldsymbol{y}_c = \{\boldsymbol{y}_i\}_{i \in c}$. The higher-order potentials $\psi_h^c(.)$ are inherently more powerful than pairwise potentials but present difficulties for learning and inference except for some very special forms. In the discrete case (*i.e*., $\boldsymbol{y}_i \in \mathcal{L} = \{1, 2, \dots\}$), for example, the most well-known higher-order function in computer vision is the robust P^n Potts model [10], defined as:

$$\psi_h^c(\boldsymbol{y}_c) = \min_{l \in \mathcal{L}} \left(\gamma_c^{max}, \gamma_c^l + \sum_{i \in c} w_i k_c^l \Delta(\boldsymbol{y}_i \neq l) \right), \tag{4}$$

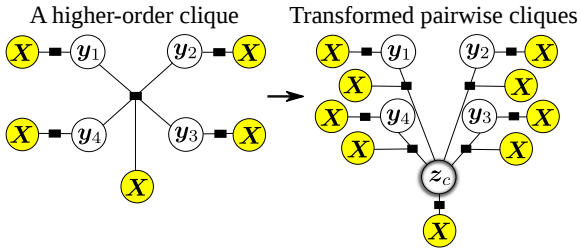


Figure 2: Left: A toy factor graph with a higher-order clique. Right: The higher-order clique can be transformed to a set of pairwise cliques by introducing an auxiliary node.

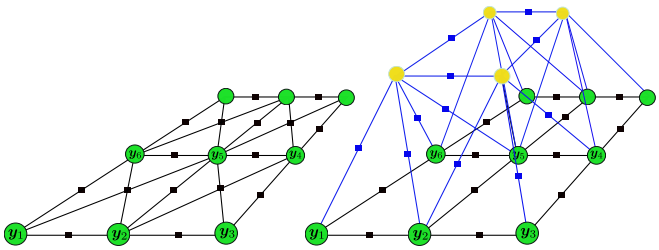


Figure 3: Left: A simple factor graph with pairwise and higher-order cliques. Right: A hierarchical graph includes a base layer, inter-layer connections and a second layer.

where γ_c^l, w_i, and k_c^l are model parameters; Δ is the indicator function. The energy is truncated by a constant γ_c^{max}. Intuitively, the robust P^n Potts models encourage all the nodes within a clique c to take the same label. Function (4) assigns a cost $w_i k_c^l$ for every node i within the clique c that does not take the dominant label $l^\star$, up to a maximum amount γ_c^{max}. As shown in [10] function (4) can be transformed to submodular pairwise functions and thus can be optimised efficiently using graph-cuts based move-making methods [5]. However a limitation of the robust P^n model is that it is associative and consequently unable to encode rich relational information between different labels.

In this work we propose more flexible higher-order potentials to capture geometric relationships within the cliques. For example, we wish to favour a local region with labels {keyboard, mouse, table-top} and penalize a local region with labels {keyboard, fridge, ceiling}. Essentially, variables in a higher-order clique are allowed to take different labels so long as these labels are all contextually consistent. To that end, we generalize the robust P^n Potts model to the following higher-order potential:

$$\psi_h^c(\boldsymbol{y}_c, \boldsymbol{X}, \mathcal{W}) = \min_{\boldsymbol{z}} \phi_u^c(\boldsymbol{z}, \boldsymbol{X}, \mathcal{W}) + \sum_{i \in c} \phi_p^c(\boldsymbol{y}_i, \boldsymbol{z}, \boldsymbol{X}, \mathcal{W}) \tag{5}$$

where $\boldsymbol{z}$ is an auxiliary variable, $\phi_u^c(.)$ and $\phi_p^c(.)$ are arbitrary functions that capture a preference for $\boldsymbol{z}$ and the interaction energy between $\boldsymbol{z}$ and $\boldsymbol{y}_i$, respectively. Both functions are conditioned on clique features. The intuition is that given a clique c with a set of labels $\{\boldsymbol{y}_i : i \in c\}$, the potential (5) will find the best "label" $\boldsymbol{z}$ for the whole clique c, which summarizes the contents of the clique and enforces contextual consistency on the label of every node $\boldsymbol{y}_i$ in the clique $i \in c$. Fig. 2 illustrates the factor graphs.

The richness of potential (5) raises two important issues. First, unlike the robust P^n Potts model, (5) cannot be optimised using efficient graph-cut based algorithms as it now contains general (non-submodular) terms $\phi_u^c(.)$ and $\phi_p^c(.)$. Second, the variable $\boldsymbol{z}$ needs to encode contextual information (*i.e.*, valid label sets) rather than just a single dominant label as in (4).

To address the above issues, we model the variables $\boldsymbol{z}$ and $\boldsymbol{y}_i$ as continuous vectors. Particularly, $\boldsymbol{z}, \boldsymbol{y}_i \in \mathbb{R}^m$ where m is the number of considered labels. The nth entry of $\boldsymbol{z}$ (or $\boldsymbol{y}_i$) represents the confidence of taking the nth label. Effectively $\boldsymbol{z}$ represents multiple labels with different confidences. Moreover we formulate $\phi_u^c(.)$ and $\phi_p^c(.)$ as multivariate Gaussian functions to facilitate efficient optimisation. Importantly, to overcome the restrictiveness of a single Gaussian model, we generate different Gaussian functions depending on the clique data contexts. In Sec. 4.3, we will show how to use regression trees to map from clique data to local Gaussian models.

The complete energy function of our higher-order CRF is

$$E(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) = E_u(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) + E_p(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) + \min_{\boldsymbol{Z}} \sum_{c \in \mathcal{C}} \left(\phi_u^c(\boldsymbol{z}_c, \boldsymbol{X}, \mathcal{W}) + \sum_{i \in c} \phi_p^c(\boldsymbol{y}_i, \boldsymbol{z}_c, \boldsymbol{X}, \mathcal{W}) \right), \tag{6}$$

where $E_u(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W})$ and $E_p(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W})$ are unary and pairwise energies as defined in (2), and $\boldsymbol{Z} = \{\boldsymbol{z}_c\}$ is a set of auxiliary variables (one for each clique).

4.2. Hierarchical CRFs

Inspired by [13], we extend our higher-order CRFs to hierarchical pairwise CRFs. Suppose that we are interested in further encoding pairwise relationships between auxiliary variables, we arrive at the following energy function:

$$E(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) = E_u(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) + E_p(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) + \min_{\boldsymbol{Z}} \sum_{c \in \mathcal{C}} \sum_{i \in c} \phi_p^c(\boldsymbol{y}_i, \boldsymbol{z}_c, \boldsymbol{X}, \mathcal{W}) + \sum_{c \in \mathcal{C}} \phi_u^c(\boldsymbol{z}_c, \boldsymbol{X}, \mathcal{W}) + \sum_{(i,j) \in \mathcal{C}} \psi_p^c(\boldsymbol{z}_i, \boldsymbol{z}_j, \boldsymbol{X}, \mathcal{W}). \tag{7}$$

The energy (7) can be interpreted as follows. The first two terms are unary and pairwise for the base layer, the third term involves inter-layer connections, and the last two terms are unary and pairwise for the second layer. Fig. 3 depicts a factor graph of a two-layer hierarchical CRF.

There is no technical reason why we cannot add more layers to our model. As above we can continue adding higher-order potentials over the auxiliary variables in the second layer and end up with a three-layer hierarchical CRFs. In general, the energy of our hierarchical CRFs can

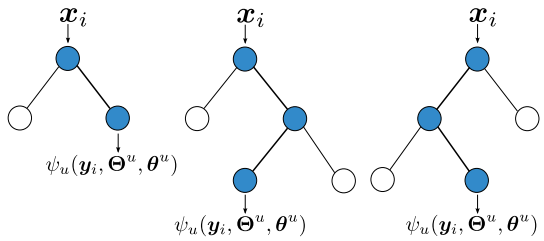


Figure 4: Illustration of regression forest fields for unary potentials. Unary features are passed down the trees to select appropriate parameters at leaf nodes. The final unary cost are computed by summing up the energies from different trees in the forest.

be written recursively as

$$E(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) = E_u(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) + E_p(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W}) + \min_{\boldsymbol{Z}} E_{conn}(\boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{X}, \mathcal{W}) + E(\boldsymbol{Z}, \boldsymbol{X}, \mathcal{W}), \tag{8}$$

where

$$E_{conn}(\boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{X}, \mathcal{W}) = \sum_{c \in \mathcal{C}} \sum_{i \in c} \phi_p^c(\boldsymbol{y}_i, \boldsymbol{z}_c, \boldsymbol{X}, \mathcal{W}) \tag{9}$$

is the inter-layer connection energy; $E(\boldsymbol{Z}, \boldsymbol{X}, \mathcal{W})$ has the same form as $E(\boldsymbol{Y}, \boldsymbol{X}, \mathcal{W})$.

Ladicky et al. [13] showed that hierarchical CRFs have considerable advantages over standard CRFs, in that they allow feature extraction and encoding of scene context at multiple scales, and thus harvest more useful statistics from the observed data. Consequentially, hierarchical CRFs yield better label predictions. Compared to the hierarchical CRF proposed in [13], our model is more expressive and flexible since it considers not only associative but also non-associative potentials.

4.3. Regression Forest Based CRFs

Having described the intuition behind our higher-order and hierarchical CRF models, in this section we detail the actual formulations of $\psi_u^i(.)$, $\psi_p^{ij}(.)$, $\phi_u^c(.)$, $\phi_p^c(.)$ and $\phi_p^c(.)$, which are used in energies (6) and (8). Inspired by Jancsary et al. [7], we construct our potential functions using regression trees, which are non-parametric and take quadratic forms. For example, the unary potential is defined as:

$$\psi_u^i(\boldsymbol{y}_i, \boldsymbol{X}, \mathcal{W}) = \frac{1}{2} \boldsymbol{y}_i^T \boldsymbol{\Theta}_i^u(\boldsymbol{X}, \mathcal{W}) \boldsymbol{y}_i - \boldsymbol{y}_i \boldsymbol{\theta}_i^u(\boldsymbol{X}, \mathcal{W}), \tag{10}$$

where $\boldsymbol{y}_i \in \mathbb{R}^m$, positive-definite matrix $\boldsymbol{\Theta}_i^u(\boldsymbol{X}, \mathcal{W}) \in \mathbb{S}_{++}^m$ and vector $\boldsymbol{\theta}_i^u(\boldsymbol{X}, \mathcal{W}) \in \mathbb{R}^m$ are local unary parameters. These parameters are extracted from the global model parameters $\mathcal{W}$ and observations $\boldsymbol{X}$ via regression trees. In particular, we first learn regression trees for the unary factor, and also Gaussian parameters (*i.e.* $\boldsymbol{\Theta}_i^u(\boldsymbol{X}, \mathcal{W})$ and $\boldsymbol{\theta}_i^u(\boldsymbol{X}, \mathcal{W})$) at leaf nodes from training data (see Sec. 4.4 for details). Then the unary energy $\psi_u^i(\boldsymbol{y}_i, \boldsymbol{X}, \mathcal{W})$ is computed by passing the features extracted at $\boldsymbol{x}_i$ down the regression trees, and eventually reaching leaf nodes via decision rules. Finally the energy $\psi_u^i(\boldsymbol{y}_i, \boldsymbol{X}, \mathcal{W})$ can be calculated trivially via (10) (see Fig. 4).

Similarly, the pairwise energy is defined as:

$$\psi_p^{ij}(\boldsymbol{y}_i, \boldsymbol{y}_j, \boldsymbol{X}, \mathcal{W}) = \frac{1}{2} \boldsymbol{y}_{ij}^T \boldsymbol{\Theta}_{ij}^p(\boldsymbol{X}, \mathcal{W}) \boldsymbol{y}_{ij} - \boldsymbol{y}_{ij} \boldsymbol{\theta}_{ij}^p(\boldsymbol{X}, \mathcal{W}), \tag{11}$$

where $\boldsymbol{y}_{ij}$ is simply a concatenation of $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$, *e.g.* $\boldsymbol{y}_{ij} = [\boldsymbol{y}_i, \boldsymbol{y}_j]$. $\boldsymbol{\Theta}_{ij}^p(\boldsymbol{X}, \mathcal{W}) \in \mathbb{S}_{++}^{2m}$ and $\boldsymbol{\theta}_{ij}^p(\boldsymbol{X}, \mathcal{W}) \in \mathbb{R}^{2m}$ are pairwise parameters. Similar to the unary factor, we also learn a regression forest for the pairwise factor and the corresponding Gaussian parameters at leaf nodes from training data. Note that all the parameters (the unary, pairwise and others factors) are jointly learned (see Sec. 4.4). At test time, we pass the features extracted at $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ down the regression trees to select the active local Gaussian model for (11).

Thanks to the graph construction in Sec. 4.1 (see Fig. 2), our higher-order energy is equivalent to a summation of unary and pairwise energies $\phi_u^c(.)$, $\phi_p^c(.)$ over the auxiliary variables. Accordingly, they can be defined as:

$$\phi_u^c(\boldsymbol{z}_c, \boldsymbol{X}, \mathcal{W}) = \frac{1}{2} \boldsymbol{z}_c^T \boldsymbol{\Theta}_c^u(\boldsymbol{X}, \mathcal{W}) \boldsymbol{z}_c - \boldsymbol{z}_c \boldsymbol{\theta}_c^u(\boldsymbol{X}, \mathcal{W}), \tag{12}$$

$$\phi_p^c(\boldsymbol{y}_i, \boldsymbol{z}_c, \boldsymbol{X}, \mathcal{W}) = \frac{1}{2} [\boldsymbol{y}_i \ \boldsymbol{z}_c]^T \boldsymbol{\Theta}_{ic}^p(\boldsymbol{X}, \mathcal{W}) [\boldsymbol{y}_i \ \boldsymbol{z}_c] - [\boldsymbol{y}_i \ \boldsymbol{z}_c]^T \boldsymbol{\theta}_{ic}^p(\boldsymbol{X}, \mathcal{W}). \tag{13}$$

Finally the pairwise energies between auxiliary variables are defined in an analogous manner to (11) and (13).

Note that the quadratic energies (10), (11), (12) and (13) are actually the canonical forms of the corresponding Gaussian densities with, for instance, mean $\mu = \boldsymbol{\Theta}_i^u(\boldsymbol{X}, \mathcal{W})^{-1} \boldsymbol{\theta}_i^u$ and covariance $\Sigma = \boldsymbol{\Theta}_i^u(\boldsymbol{X}, \mathcal{W})^{-1}$ for the unary energy, which are particularly convenient for inference and learning. However, unlike the restricted unimodality Gaussian CRFs [26], regression trees based CRFs are much more flexible and powerful as the potentials are conditioned on the data via sophisticated regression trees. Effectively they admit to generate different Gaussian models for different data-dependent contexts. Moreover, the potentials can be either associative or non-associative depending on the input data and thus which leaf nodes are selected.

4.4. Inference and Learning

For notational simplicity let us consider a two-layer model. Since all the terms are quadratic forms, we can stack the model parameters to end up with a single quadratic energy function:

$$E(\tilde{\boldsymbol{Y}}, \boldsymbol{X}, \mathcal{W}) = \frac{1}{2} \tilde{\boldsymbol{Y}}^T \boldsymbol{\Theta}(\boldsymbol{X}, \mathcal{W}) \tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{Y}} \boldsymbol{\theta}(\boldsymbol{X}, \mathcal{W}) \tag{14}$$

where $\tilde{Y} = [\boldsymbol{Y} \ \boldsymbol{Z}]$ ($\boldsymbol{Z}$ are auxiliary variables for the second layer); $\boldsymbol{\Theta}$ and $\boldsymbol{\theta}$ are stacked parameters, $\boldsymbol{\Theta} \in \mathbb{S}_{++}^{m(|\mathcal{V}||\mathcal{C}|)}$ and $\boldsymbol{\theta} \in \mathbb{R}^{m(|\mathcal{V}||\mathcal{C}|)}$.

Assume that we have learned the forest structures and parameters $\mathcal{W}$, the inference task is to find the optimal prediction $\tilde{\boldsymbol{Y}}^*$, which can be computed analytically as $\tilde{\boldsymbol{Y}}^* = \boldsymbol{\Theta}(\boldsymbol{X}, \mathcal{W})^{-1}\boldsymbol{\theta}(\boldsymbol{X}, \mathcal{W})$. However, for large problems computing the inverses of the high-dimensional matrices $\boldsymbol{\Theta}(\boldsymbol{X}, \mathcal{W})$ is prohibitively expensive. Many large-scale linear equation solvers can be used to solve this problem. In our work we resort to the standard L-BFGS optimisation method, as done in [8]. Given the optimal $\boldsymbol{Y}^*$, the actual predicted labels can be computed trivially by extracting the index of largest entry in each vector $\boldsymbol{y}_i^*$.

To learn the trees and the model parameters $\mathcal{W}$, we follow the two-steps approach proposed in [7, 8]. Particularly, given training data including a pair of observations and ground true labels $\langle \boldsymbol{X}, \boldsymbol{Y} \rangle$ (concatenated from a large collection of training instances), we first learn the regression forests for unary, pairwise and higher-order terms separately using the classical variance reduction criterion [6]. We randomly split the data into smaller equal-size subsets which are used to learn the trees separately. Once the tree structures have been learned, the second step is to optimise the parameters $\mathcal{W}$, *i.e.*, the Gaussian parameters at each leaf of the trees. Since the objective function (14) is high-dimensional, optimising $\mathcal{W}$ using the maximum likelihood method (MLE) is infeasible since it requires expensive computation of the inverse of the matrix $\boldsymbol{\Theta}$ (for computing gradients). Thus we use the pseudo-likelihood (MPLE) method instead as suggested by Jancsary et al. [7]. The idea is to decompose the large optimisation problem into smaller ones, each of which can be solved efficiently. As a consequence, we need to solve the following optimisation problem:

$$\begin{aligned} \mathcal{W}^* \in \arg\min_{\mathcal{W}\in\Omega} & -\sum_{i\in\mathcal{V}} \log(p(\boldsymbol{y}_i \mid \tilde{\boldsymbol{Y}} \setminus \boldsymbol{y}_i, \boldsymbol{X}, \mathcal{W})) \\ & -\sum_{c\in\mathcal{C}} \log(p(\boldsymbol{z}_c \mid \tilde{\boldsymbol{Y}} \setminus \boldsymbol{z}_c, \boldsymbol{X}, \mathcal{W})), \end{aligned} \tag{15}$$

where Ω is a constraint set that enforces the matrices $\boldsymbol{\Theta}_{\cdot}^{u}$, $\boldsymbol{\Theta}_{\cdot}^{p}$ be positive-definite. The (log) probability of variable $\boldsymbol{y}_i$ conditioned on other variables is given as:

$$\begin{aligned} \log(p(\boldsymbol{y}_i \mid \tilde{\boldsymbol{Y}} \setminus \boldsymbol{y}_i, \boldsymbol{X}, \mathcal{W})) = \psi_u^i(\boldsymbol{y}_i, \boldsymbol{X}, \mathcal{W}) \\ + \sum_{j\in\mathcal{N}_i} \psi_p^{ij}(\boldsymbol{y}_i, \boldsymbol{y}_j, \boldsymbol{X}, \mathcal{W}) + \sum_{c\in\mathcal{C}_i} \phi_p^c(\boldsymbol{y}_i, \boldsymbol{z}_c, \boldsymbol{X}, \mathcal{W}) + k, \end{aligned} \tag{16}$$

where $\mathcal{N}_i$ is a set of neighbours of node i and $\mathcal{C}_i$ is a set of cliques containing node i, and k is a constant. The probability of the variable $\boldsymbol{z}_c$ can be derived similarly. The constrained non-linear problem (15) can be solved by using the projected L-BFGS method. We refer readers to [7] for more details.

5. Implementation Details

So far we have presented our higher-order and hierarchical CRF models for label prediction, we now describe our full 3D indoor semantic scene labelling pipeline which additionally includes the 3D reconstruction and feature extraction subroutines. Our system builds on the work of [8] and unless otherwise stated, the basic steps are the same. A brief description is given as below, details can be found in [8].

5.1. 3D Reconstruction

Starting with a sequence of RGB-D images of an indoor scene captured from a depth sensor (Kinect), we first generate a dense volumetric representation of the scene. This can be achieved using the KinectFusion algorithm [17]. As suggested in [8], we utilise both depth and color (RGB) information for better reconstructions.

5.2. Oversegmentation

As the reconstructed 3D scene is composed of millions of 3D points, learning and inference on such large graphs could be computationally expensive. To reduce the computational expense, we over-segment the point cloud into thousands of segments, each of which hopefully covers part of a distinct object; we then predict labels for the segments instead. We resort to the over-segmentation algorithm proposed in [8], based on SLIC [1] super-pixels for 2D images.

5.3. Graph Construction

Given a set of available segments, we construct a graph whose nodes associate with the segments. We create edges between pairs of segments whose distances are less than a certain threshold d_1 (*e.g.*, $d_1 = 0.3$ meters). To create cliques, we simply group neighbouring segments, which are not necessarily visually and geometrically similar, though more complicated strategies (*e.g.* [16]) can be investigated. In particular, we uniformly select 50% number of segments to be the clique centroids, then for each centroid we add up the segments within a certain distance d_2 from the centroid, resulting in cliques. Unlike the previous associative higher-order models [10, 13, 20] which create cliques by grouping similar pixels/voxels likely from the same object, our cliques are allowed to contain parts of different objects, requiring no expensive clustering algorithms as for instance in [13]. Moreover, our cliques potentially could be overlapped and vary in size. These cliques are also called super-segments. For our hierarchical model, we connect super-segments whose distances between their centroids are smaller than a threshold d_3.

5.4. Feature Extraction

Node Features. In order to predict the labels for the segments accurately, we need to extract expressive and discriminative features for each segment. Similar to the previous work [12, 8], features should encode visual appearance, shape and geometrical properties. The appearance features are computed by using histogram of HSV colors and histogram of gradients (HOG) , whereas shape and geometric features describe, for example, planarity, area, height above the ground floor. (See [8] for the full list of node features and detailed computations.) Finally the extracted features are binned using the cumulative binning strategy [2].

Methods	Parameter settings				Macro P	Macro R	Micro P/R	Training	Inference
	# trees	d_1	d_2	d_3					
Unary	15	-	-	-	31.40	29.20	65.49	12min	0.4sec
Pairwise RTF [8]	15	0.3	-	-	59.00	33.18	76.13	5.5h	17sec
Higher-order RTF	15	0.3	0.3	-	63.44	36.94	**78.44**	8.8h	29sec
Hierarchical RTF	15	0.3	0.3	0.6	**65.01**	36.23	78.00	12.6h	32sec
Pairwise RTF [8]	15	0.6	-	-	61.69	**37.27**	78.37	8.4h	36sec

Table 1: Experimental results on Cornell-RGBD-Dataset [12]. The table shows the average precision and recall over 5-fold cross valiation. Average training and testing time are also included. It can be seen that our higher-order and hierarchical models clearly outperform the unary and pairwise models. Note that the micro precision and micro recall are identical since the algorithms predict one label for each segment.

Methods	Parameter settings				Macro P	Macro R	Micro P/R	Training	Inference
	# trees	d_1	d_2	d_3					
Unary	10	-	-	-	33.38	34.86	46.20	16m	2.0sec
Pairwise RTF [8]	10	0.3	-	-	56.11	42.32	61.28	3.4h	30sec
Higher-order RTF	10	0.3	0.3	-	57.27	42.88	61.83	6h	48sec
Hierarchical RTF	10	0.3	0.3	0.6	**59.24**	**43.09**	**62.67**	9.8h	71sec

Table 2: Comparable results on NYU Depth dataset [25]. Average precision and recall as well as training, testing time for different methods are shown. Again higher-order and hierarchical models have considerable improvement over the pairwise and unary models. The micro precision and micro recall are identical since the algorithms predict one label for each segment.

Clique Features. We take the average over the binned features of all the segments constituting the clique as the clique features.

Pairwise Features. To describe the contextual relations between two segments, we extract features based on the differences of their appearances and their relative geometric arrangement. Specifically, we compute the absolute differences of HSV values and HOG descriptors. The geometric relations include, for example, angle between normals, distance between two centroid, co-planarity, connectivity, etc. Again we adopt features from [8]. Similar to the node features, the pairwise features are binned before being used.

Inter-layer Features. As mentioned previously, our higher-order and hierarchical models require pairwise connections between segments and super-segments (cliques). We simply compute the pairwise features between the segment and the clique centroid.

Pairwise Features for Cliques. Our hierarchical model also requires pairwise features between super-segments. Similar to the pairwise features between segments, we also compute the appearance differences and relative geometric relations between super-segments. Since a super-segment is constructed by adding neighbouring segments to a central segment, we instead compute pairwise features between two central segments.

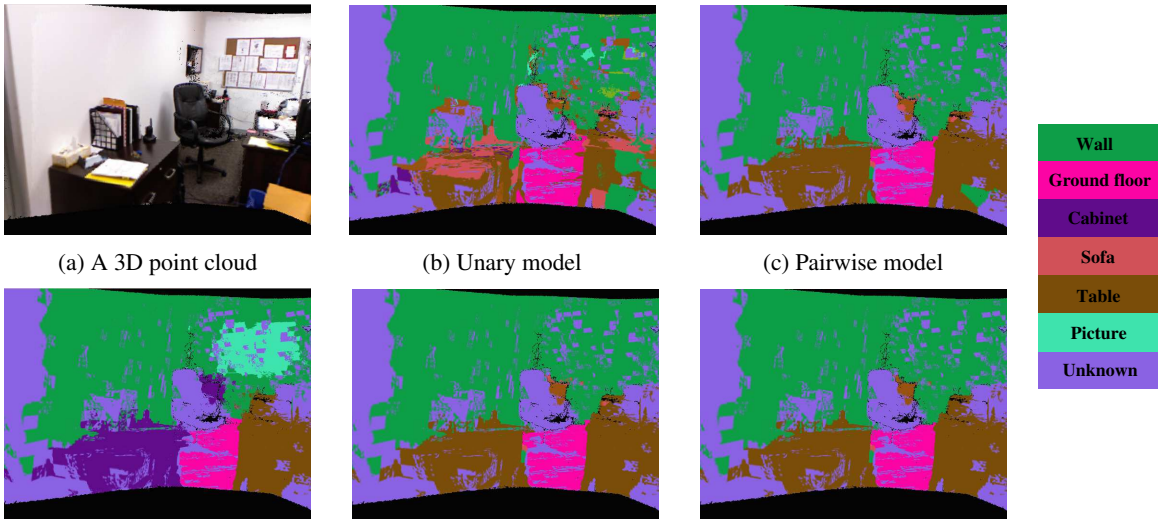
6. Experimental Results

Closely following the protocol described in [8], we conducted experiments on 3D indoor scene understanding using two datasets (Cornell-RGBD-Dataset [2] and NYU Depth [25]). The two datasets are collections of RGB-D images acquired from Kinect sensors. We used 24 office scenes from the Cornell-RGBD-Dataset and 40 home-office sequences from NYU Depth for the experiments. To create ground truth labels for training and testing, we follow the method described in [8]. In particular, the 3D ground truth labels are obtained by re-projecting all the 3D segments into the original images, where ground truth labels have been manually annotated.

We compare our higher-order and hierarchical models against the state-of-the-art pairwise models for 3D indoor scene understanding [8][12]. The trees based method [8] is actually a special case of our framework, where the higher-order terms are omitted. We have not successfully managed to run the SVM based method [12] on the datasets we have considered since it requires a prohibitive amount of memory to process using their publicly available implementation[1]. Nevertheless, using a much smaller dataset, it has been reported in [8] that the SVM based method is comparable with the trees based pairwise model [8]. We also include a model with unary terms only into comparisons to demonstrate the advantages of scene contexts for label prediction. Note that we use the same number of trees and depths for all the models to ensure fair comparison.

We evaluate all methods using 5-fold cross validation. In each run, 80% of data is randomly selected for training and the remained 20% is for testing. As in [12][8][2] we compute the average macro precision and recall, micro precision and recall across the folds for performance comparisons. The numerical results are reported in Table 1 (Cornell-RGBD) and Table 2 (NYU Depth). A sample of

[1] http://pr.cs.cornell.edu/sceneunderstanding/data/data.php



(a) A 3D point cloud (b) Unary model (c) Pairwise model

(d) Ground truth (e) Higher-order model (f) Hierarchical model

Figure 5: A sample of 3D semantic scene labelling results on NYU Depth dataset using different methods. Note that the light purple regions with the unknown label result from either inconsistent labels or missing labels when the 3D scene is projected into multiple ground truth labelled 2D images. We ignored these regions during training and testing.

qualitative results is shown in Fig. 5.

In both datasets, the first impression is that scene contexts are particularly useful for label prediction. It is clear that the models encoding scene context (via pairwise, higher-order potentials) significantly outperform the unary model. Around 10-20% improvements can be observed in precision and recall separately. The results also show that the higher-order model considerably improves the pairwise model in macro precision, but is only slightly better in macro recall and micro precision (see Table 2). This can be explained by the fact that the truth labels for different object classes are very imbalanced (*e.g.*, a majority of labels go to floors and walls while a small amount goes to objects such as laptops, books). As a result, average precision across the class labels (macro precision) is more appropriate for these datasets. Further improvements can be observed in our hierarchical model.

Certainly, our proposed models are more expensive in both learning and inference than the unary and pairwise models since our higher-order and hierarchical graphs are more complicated with more nodes and edges. On the other hand, we have attempted to test denser pairwise models with more edge connections (by increasing the context range d_1). We observed that the denser pairwise models slightly increase the performance, yet the training and inference costs significantly increase. Indeed, the last row in Table 1 reveals that the pairwise model with context range $d_1 = 0.6$ (meters) is still worse than our higher-order model with $d_1 = 0.3, d_2 = 0.3$ (meters), but its inference is about 1.5 times slower.

It is worth mentioning that we have constructed very simple higher-order cliques (*i.e.* by grouping nearby segments) and extracted their features by simply taking average of segments' features for all the experiments. We believe that more sophisticated hierarchical scene decomposition and higher-order clique feature extraction would definitely further improve the performance.

7. Conclusion

We have proposed higher-order and hierarchical CRF models for 3D scene semantic labelling, which have the ability to strongly exploit complex contextual information in 3D scenes. Our higher-order potentials encourage geometrical consistencies within groups of 3D segments, thus more advantageous in extracting relational information between different objects than the pairwise potentials. Furthermore, our hierarchical model enables us to exploit the scene context and extract features at multiple scales. The model considers short range interactions between parts of objects at low levels, and long range interactions between different objects at high levels. We showed that the proposed methods clearly outperform the state-of-the-art pairwise models for 3D semantic scene understanding on two different datasets. It is important to note that our proposed models are general and could be applied to any multi-labelling problems, which we plan to consider in our future work.

Acknowledgements

This research was supported by the Australian Research Council through the Centre of Excellence for Robotic Vision (CE140100016) and Laureate Fellowship (FLFL130100102) to IDR.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Su. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, 2012. 6

[2] A. Anand, H. S. Koppula, T. Joachims, and A. Saxena. Contextually guided semantic labeling and search for three-dimensional point clouds. *Int. J. Rob. Res.*, 32(1):19–34, Jan. 2013. 1, 6, 7

[3] D. Anguelov, B. Taskarf, V. Chatalbashev, D. Koller, D. Gupta, G. Heitz, and A. Ng. Discriminative learning of markov random fields for segmentation of 3d scan data. In *Computer Vision and Pattern Recognition (CVPR)*, 2005. 2

[4] J. Behley, V. Steinhage, and A. Cremers. Performance of histogram descriptors for the classification of 3d laser range data in urban environments. In *Robotics and Automation (ICRA)*, 2012. 2

[5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 23(11):1222–1239, 2001. 4

[6] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984. 6

[7] J. Jancsary, S. Nowozin, T. Sharp, and C. Rother. Regression tree fields - an efficient, non-parametric approach to image labeling problems. In *CVPR*, 2012. 1, 2, 3, 5, 6

[8] O. Kahler and I. Reid. Efficient 3d scene labeling using fields of trees. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2013. 1, 2, 3, 6, 7

[9] P. Kohli, M. Kumar, and P. Torr. P3 beyond: Solving energies with higher order cliques. In *CVPR*, 2007. 1

[10] P. Kohli, L. Ladický, and P. H. Torr. Robust higher order potentials for enforcing label consistency. *Int. J. Comput. Vision*, 82(3):302–324, May 2009. 1, 3, 4, 6

[11] P. Kontschieder, S. Rota Bulo, H. Bischof, and M. Pelillo. Structured class-labels in random forests for semantic image labelling. In *ICCV*, 2011. 3

[12] H. S. Koppula, A. Anand, T. Joachims, and A. Saxena. Semantic labeling of 3d point clouds for indoor scenes. In *Advances in Neural Information Processing Systems 24*, pages 244–252. 2011. 1, 2, 3, 6, 7

[13] L. Ladicky, C. Russell, P. Kohli, and P. Torr. Associative hierarchical random fields. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(6):1056–1077, June 2014. 4, 5, 6

[14] Y. Lu and C. Rasmussen. Simplified markov random fields for efficient semantic labeling of 3d point clouds. In *Intelligent Robots and Systems (IROS)*, 2012. 2

[15] P. Márquez-Neila, P. Kohli, C. Rother, and L. Baumela. Non-parametric higher-order random fields for image segmentation. In *Computer Vision - ECCV 2014*, 2014. 3

[16] M. Najafi, S. Taghavi Namin, M. Salzmann, and L. Petersson. Non-associative higher-order markov networks for point cloud classification. In *ECCV*, 2014. 2, 3, 6

[17] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of the 10th IEEE International Symposium on Mixed and Augmented Reality*, 2011. 1, 6

[18] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. Dtam: Dense tracking and mapping in real-time. In *Proceedings of the 2011 International Conference on Computer Vision*, 2011. 1

[19] S. Nowozin, C. Rother, S. Bagon, T. Sharp, B. Yao, and P. Kohli. Decision tree fields. *Computer Vision, IEEE International Conference on*, 0:1668–1675, 2011. 1, 2

[20] K. Park and S. Gould. On learning higher-order consistency potentials for multi-class pixel labeling. In *Computer Vision ECCV 2012*, pages 202–215. 2012. 3, 6

[21] T. T. Pham, T.-J. Chin, K. Schindler, and D. Suter. Interacting geometric priors for robust multimodel fitting. *Image Processing, IEEE Transactions on*, 23(10):4601–4610, Oct 2014. 3

[22] R. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *ICRA*, 2009. 2

[23] R. Shapovalov and A. Velizhev. Cutting-plane training of non-associative markov network for 3d point cloud segmentation. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), International Conference on*, 2011. 2

[24] R. Shapovalov, A. Velizhev, and O. Barinova. Non-associative markov networks for 3d point cloud classification. In *Photogrammetric Computer Vision and Image Analysis*, 2010. 2

[25] N. Silberman and R. Fergus. Indoor scene segmentation using a structured light sensor. In *ICCV Workshops*, 2011. 2, 7

[26] M. Tappen, C. Liu, E. Adelson, and W. Freeman. Learning gaussian conditional random fields for low-level vision. In *Computer Vision and Pattern Recognition*, 2007. 2, 5

[27] X. Xiong and D. Huber. Using context to create semantic 3d models of indoor environments. In *Proc. BMVC*, 2010. 2

[28] X. Xiong, D. Munoz, J. A. Bagnell, and M. Hebert. 3-d scene analysis via sequenced predictions over points and regions. In *ICRA*, 2011. 2