JSIS3D: Joint Semantic-Instance Segmentation of 3D Point Clouds with Multi-Task Pointwise Networks and Multi-Value Conditional Random Fields

Quang-Hieu Pham[†] Duc Thanh Nguyen[‡] Binh-Son Hua[▷] Gemma Roig[†] Sai-Kit Yeung[◁]
[†]Singapore University of Technology and Design
[‡]Deakin University [▷]The University of Tokyo
[◁]Hong Kong University of Science and Technology

Abstract

Deep learning techniques have become the to-go models for most vision-related tasks on 2D images. However, their power has not been fully realised on several tasks in 3D space, e.g., 3D scene understanding. In this work, we jointly address the problems of semantic and instance segmentation of 3D point clouds. Specifically, we develop a multi-task pointwise network that simultaneously performs two tasks: predicting the semantic classes of 3D points and embedding the points into high-dimensional vectors so that points of the same object instance are represented by similar embeddings. We then propose a multi-value conditional random field model to incorporate the semantic and instance labels and formulate the problem of semantic and instance segmentation as jointly optimising labels in the field model. The proposed method is thoroughly evaluated and compared with existing methods on different indoor scene datasets including S3DIS and SceneNN. Experimental results showed the robustness of the proposed joint semantic-instance segmentation scheme over its single components. Our method also achieved state-of-the-art performance on semantic segmentation.

1. Introduction

The growing popularity of low-cost 3D sensors (*e.g.*, Kinect) and light-field cameras has opened many 3D-based applications such as autonomous driving, robotics, mobile-based navigation, virtual reality, and 3D games. This development also acquires the capability of automatic understanding of 3D data. In 2D domain, common scene understanding tasks including image classification, semantic segmentation, or instance segmentation, have achieved notable results [13, 3]. However, the problem of 3D scene understanding poses much greater challenges, *e.g.*, large-scale and noisy data processing.

Literature has shown that the data of a 3D scene can be represented by a set of images capturing the scene at different viewpoints [14, 46, 42], in a regular grid of volumes [47, 26, 28], or simply in a 3D point cloud [33, 16, 45, 17, 24]. Our work is inspired by the point-based representation for several reasons. Firstly, compared with multi-view and volumetric representations, point clouds offer a more compact and intuitive representation of 3D data. Secondly, recent neural networks directly built on point clouds [33, 16, 24, 45, 17, 18, 22, 23, 48] have shown promising results across multiple tasks such as object recognition and semantic segmentation.

In this paper, we address two fundamental problems in 3D scene understanding: semantic segmentation and instance segmentation. Semantic segmentation aims to identify a class label or object category (*e.g.*, chair, table) for every 3D point in a scene while instance segmentation clusters the scene into object instances. These two problems have often been tackled separately in which instance segmentation/detection is a post-processing task of semantic segmentation [31, 30]. However, we have observed that object categories and object instances are mutually dependent. For instance, shape and appearance features extracted on an instance would help to identify the object category of that instance. On the other hand, if two 3D points are assigned to different object categories, they unlikely belong to the same object instance. Therefore, it is desirable to couple semantic and instance segmentation into a single task. Towards the above motivations, we make the following contributions in our work.

- A network architecture namely multi-task pointwise network (MT-PNet) that simultaneously performs two tasks: predicting the object categories of 3D points in a point cloud, and embedding these 3D points into high-dimensional feature vectors that allow clustering the points into object instances.
- A multi-value conditional random field (MV-CRF) model that formulates the joint optimisation of class labels and object instances into a unified framework,

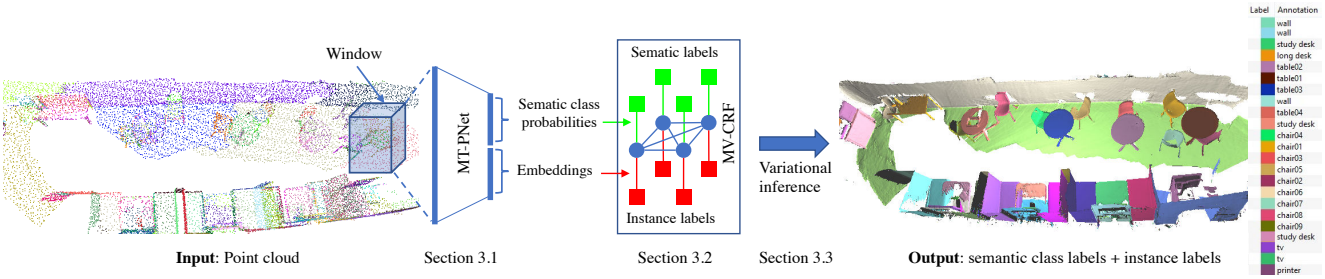


Figure 1. Pipeline of our proposed method. Given an input 3D point cloud, we scan the point cloud by overlapping windows. 3D vertices are then extracted from a window and passed through our multi-task neural network to get the semantic labels and instance embeddings. We then optimise a multi-value conditional random field model to produce the final results. Scene data is retrieved from [15].

which can be efficiently solved using variational mean field technique. To the best of our knowledge, we are the first to explore the joint optimisation of semantics and instances in a unified framework.

- Extensive experiments on different benchmark datasets to validate the proposed method as well as its main components. Experimental results showed that the joint semantic and instance segmentation outperformed each individual task, and the proposed method achieved state-of-the-art performance on semantic segmentation.

The remainder of the paper is organised as follows. Section 2 briefly reviews related work. The proposed method is described in Section 3. Experiments and results are presented and discussed in Section 4. The paper is finally concluded in Section 5.

2. Related Work

This section reviews recent semantic and instance segmentation techniques in 3D space. We especially focus on deep learning-based techniques applied on 3D point clouds due to their proven robustness as well as being contemporary seminal in the field. For the sake of brevity, we later refer to the traditional, category-based semantic segmentation as *semantic segmentation*, and instance-based semantic segmentation as *instance segmentation*.

2.1. Semantic Segmentation

Recent availability of indoor scene datasets [37, 15, 5, 1] has sparked research interests in 3D scene understanding, particularly semantic segmentation. We categorise these recent works into three main categories based on their type of input data, namely multi-view images, volumetric representation, and point clouds.

Multi-view approach. This approach often uses pre-trained models on 2D domain and applies them to 3D space. Per-vertex labels are obtained by back-projecting and fusing 2D predictions from colour or RGB-D images onto 3D space. Predictions on 2D can be done via classifiers, *e.g.*, random forests [14, 36, 46, 42], or deep neural networks [27, 49, 30]. Such techniques can be implemented in tandem with 3D scene reconstruction, creating a real-time semantic reconstruction system. However, this approach suffers from inconsistencies between 2D predictions, and its performance might depend on view placements.

Volumetric approach. The robustness of deep neural networks in solving several scene understanding tasks on images has inspired applying deep neural networks directly in 3D space to solve 3D scene understanding problem. In fact, convolutions on a regular grid, *e.g.*, image structures, can be easily extended to 3D, which leads to deep learning with volumetric representation [47, 26, 28]. To support high-resolution segmentation and reduce memory footprints, a hierarchical data structure such as an octree was proposed to limit convolution operations only on free-space voxels [35]. It has been shown that the performance of semantic segmentation can be improved by solving the problem jointly with scene completion [39, 6].

Point cloud approach. In contrast to volume, point cloud is a compact yet intuitive representation that directly stores attributes of the geometry of a 3D scene via coordinates and normals of vertices. Point clouds arise naturally from commodity devices such as multi-view stereos, depth, and LIDAR sensors. Point clouds can also be converted to other representations such as volumes [40] or mesh [41]. While convolutions can be done conveniently on volumes [40], they are not applicable straightforwardly on point clouds. This problem was first addressed in the work of Qi *et al.* [32], and subsequently explored by several others, *e.g.*, [33, 16, 45, 17, 24, 23, 48]. Semantic segmentation can further be extended to graph convolution to handle large-scale point clouds [22], and with the use of kd-tree to address non-uniform point distributions [18, 12].

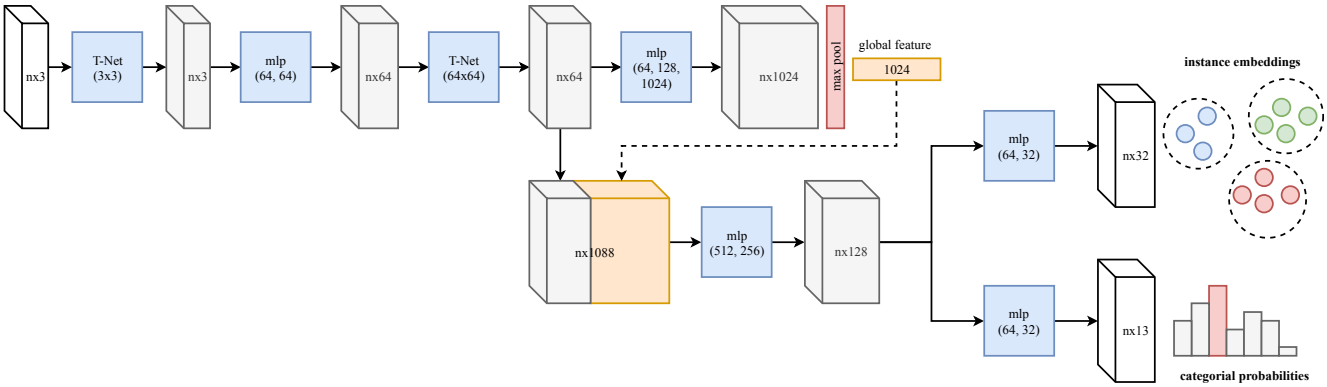


Figure 2. Our proposed MT-PNet architecture, which based on PointNet [32]. The point cloud first go through a feed-forward neural network to compute a 128-dimension feature vector for each point. Here it splits into to branches: one for instance embedding and the other for semantic segmentation.

Conditional Random Fields (CRFs) CRFs are often used in semantic segmentation of 3D scenes, *e.g.*, [41, 14, 20, 46, 42, 27, 34]. In general, CRFs make use of unary and binary potentials capturing characteristics of individual 3D points [46] or meshes [41], and their co-occurrence. To enhance CRFs with prior knowledge, higher-order potentials are introduced [21, 11, 50, 2, 49, 10, 30]. Higher-order potentials, *e.g.*, object detections [21, 2, 30], act as additional cues to help the inference of semantic class labels in CRFs.

2.2. Instance Segmentation

In general, there are two common strategies to tackle instance segmentation. The first strategy is to localise object bounding boxes using object detection techniques, and then find a mask that separates foreground and background within each box. This approach has been shown to work robustly with images [7, 13], while deemed challenging in 3D domain. This probably due to existing 3D object detectors are often not trained from scratch but make use of image features [9, 31, 25]. Extending such approaches with masks is possible but might lead to a sub-optimal and more complicated pipeline.

Instead, given the promising results of semantic segmentation on 3D data [32, 1, 16], the second strategy is to extend a semantic segmentation framework by adding a procedure that proposes object instances. In an early attempt, Wang *et al*. [44] proposed to learn a semantic map and a similarity matrix of point features based on the PointNet in [32]. Authors then proposed an heuristic and non-maximal suppression step to merge similar points into instances.

3. Proposed Method

In this section, we describe our proposed method for semantic and instance segmentation of 3D point clouds. Given a 3D point cloud, we first scan the entire point cloud by overlapping 3D windows. Each window (with its associated 3D vertices) is passed to a neural network for predicting the semantic class labels of the vertices within the window and embedding the vertices into high-dimensional vectors. To enable such tasks, we develop a multi-task pointwise network (MT-PNet) that aims to predict an object class for every 3D point in the scene and at the same time to embed the 3D point with its class label information into a vector. The network encourages 3D points belonging to the same object instance be pulled to each other while pushing those of different object instances as far away from each other as possible. Those class labels and embeddings are then fused into a multi-value conditional random field (MV-CRF) model. The semantic and instance segmentation are finally performed jointly using variational inference. We illustrate the pipeline of our method in Figure 1 and describe its main components in the following sub-sections.

3.1. Multi-Task Pointwise Network (MT-PNet)

Our MT-PNet is based on the feed forward architecture of PointNet proposed by Qi *et al*. in [32] (see Figure 2). Specifically, for an input point cloud of size N, a feature map of size $N \times D$, where D is the dimension of features for each point, is first computed. The MT-PNet then diverges into two different branches performing two tasks: predicting the semantic labels for 3D points and creating their pointwise instance embeddings. The loss of our MT-PNet is the sum of the losses of its two branches,

$$\mathcal{L} = \mathcal{L}_{prediction} + \mathcal{L}_{embedding} \tag{1}$$

The prediction loss $\mathcal{L}_{prediction}$ is defined by the cross-entropy as usual. Inspired by the work in [8], we employ a discriminative function to present the embedding loss $\mathcal{L}_{embedding}$. In particular, suppose that there are K instances and $N_k, k \in \{1, ..., K\}$ is the number of elements in the k-th instance, $\mathbf{e}_j \in \mathbb{R}^d$ is the embedding of point v_j, and $\boldsymbol{\mu}_k$ is the mean of embeddings in the k-th instance. The

embedding loss can be defined as follows,

$$\mathcal{L}_{embedding} = \alpha \cdot \mathcal{L}_{pull} + \beta \cdot \mathcal{L}_{push} + \gamma \cdot \mathcal{L}_{reg} \quad (2)$$

where

$$\mathcal{L}_{pull} = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{N_k}\sum_{j=1}^{N_k}\left[\|\boldsymbol{\mu}_k - \mathbf{e}_j\|_2 - \delta_v\right]_+^2 \quad (3)$$

$$\mathcal{L}_{push} = \frac{1}{K(K-1)}\sum_{k=1}^{K}\sum_{m=1,m\neq k}^{K}\left[2\delta_d - \|\boldsymbol{\mu}_k - \boldsymbol{\mu}_m\|_2\right]_+^2 \quad (4)$$

$$\mathcal{L}_{reg} = \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{\mu}_k\|_2 \quad (5)$$

where $[x]_+ = \max(0, x)$, δ_v and δ_d are respectively the margins for the pull loss $\mathcal{L}_{pull}$ and push loss $\mathcal{L}_{push}$. We set $\alpha = \beta = 1$ and $\gamma = 0.001$ in our implementation.

A simple intuition for this embedding loss is that the pull loss $\mathcal{L}_{pull}$ attracts embeddings towards the centroids, *i.e.*, $\boldsymbol{\mu}_k$, while the push loss $\mathcal{L}_{push}$ keeps these centroids away from each other. The regularisation loss $\mathcal{L}_{reg}$ acts as a small force that draws all centroids towards the origin. As shown in [8], if we set the margin $\delta_d > 2\delta_v$, then each embedding will be closer to its own centroid than other centroids.

3.2. Multi-Value Conditional Random Fields (MV-CRF)

Let $V = \{v_1, .., v_N\}$ be the point cloud of a 3D scene obtained after 3D reconstruction. Each 3D vertex v_j in the point cloud is represented by its 3D location $\mathbf{l}_j = [x_j, y_j, z_j]$, normal $\mathbf{n}_j = [n_{j,x}, n_{j,y}, n_{j,z}]$, and colour $\mathbf{c}_j = [c_{j,R}, c_{j,G}, c_{j,B}]$. By using the proposed MT-PNet, we also obtain an embedding $\mathbf{e}_j \in \mathbb{R}^d$ for each point v_j. Let $L^S = \{l_1^S, ..., l_N^S\}$ be the set of semantic labels that need to be assigned to the point cloud V, where l_j^S represent the semantic class, *e.g.*, chair, table, etc., of v_j. Similarly, let $L^I = \{l_1^I, ..., l_N^I\}$ be the set of instance labels of V, *i.e.*, all vertices of the same object instance will have the same instance label l_j^I. The labels l_j^S and l_j^I are random variables taking values in S and I which are the set of semantic labels and instance labels respectively. Note that S is predefined while I is unknown and needs to be determined through instance segmentation.

We now consider each vertex $v_j \in V$ as a node in a graph, two arbitrary nodes v_j, v_k are connected by an undirected edge, and each vertex v_j is associated with its semantic and instance labels represented by the random variables l_j^S and l_j^I. Our graph defined over V, L^S, and L^I is named multi-value conditional random fields (MV-CRF); this is because each node v_j is associated to two labels (l_j^S, l_j^I) taking values in $S \times I$. The joint semantic-instance segmentation of the point cloud V thus can be formulated via minimising the following energy function,

$$\begin{aligned} E(L^S, L^I|V) = &\sum_j \varphi(l_j^S) + \sum_{(j,k),j<k}\varphi(l_j^S, l_k^S) \\ &+ \sum_j \psi(l_j^I) + \sum_{(j,k),j<k}\psi(l_j^I, l_k^I) \\ &+ \sum_{s\in S}\sum_{i\in I}\phi(s,i) \end{aligned} \quad (6)$$

We note that our MV-CRF substantially differs from existing higher-order CRFs, *e.g.*, [21, 11, 2, 30]. Specifically, in existing higher-order CRFs, higher-orders, *e.g.* object detections, are used as prior knowledge that helps to improve segmentation. In contrast, our MV-CRF treats instance labels and semantic labels equally as unknown and optimises them simultaneously.

The energy function $E(L^S, L^I|V)$ in (6) involves in a number of potentials that incorporate physical constraints (*e.g.*, surface smoothness, geometric proximity) and semantic constraints (*e.g.*, shape consistency between object class and instances) in both semantic and instance labeling. Specifically, the unary potential $\varphi(l_j^S)$ is defined over the semantic labels l_j^S and computed directly from the classification score of MT-PNet as,

$$\varphi(l_j^S = s) \propto -\log p(v_j|l_j^S = s) \quad (7)$$

where s is a possible class label in S and $p(v_j|l_j^S = s)$ is the probability (*e.g.*, softmax value) that our network classifies v_j to the semantic class s.

We have found that vertices of the same object class often share the same distribution of classification scores, *i.e.*, $p(v_j|l_j^S)$. We thus model the pairwise potential $\varphi(l_j^S, l_k^S)$ via the classification scores of both v_j and v_k. Specifically, we define,

$$\varphi(l_j^S, l_k^S) = \omega_{j,k}\exp\left\{-\frac{[p(v_j|l_j^S) - p(v_k|l_k^S)]^2}{2\theta^2}\right\} \quad (8)$$

where $\omega_{j,k}$ is obtained from the Pott compatibility as,

$$\omega_{j,k} = \begin{cases} -1, & \text{if } l_j^{S/I} = l_k^{S/I} \\ 1, & \text{otherwise.} \end{cases} \quad (9)$$

The unary potential $\psi(l_j^I)$ enforces embeddings belonged to the same instance to get as close to their mean embeddings as possible. Intuitively, embeddings of the same instance are expected to convert to their modes in the embedding space. Meanwhile, embeddings of different instances are encouraged to diverge from each other. Specifically, suppose that the instance label set $I = \{i_1, ..., i_K\}$ includes K instances. Suppose that the current configuration

of L^I assigns all the vertices in V into these K instances. For each instance label $i \in I$, we define,

$$\psi(l_j^I = i) = -\frac{\exp\left[-\frac{1}{2}(\mathbf{e}_j - \boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i^{-1}(\mathbf{e}_j - \boldsymbol{\mu}_i)\right]}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}_i|}} - \log\left[\sum_k 1(l_k^I = i)\right] \tag{10}$$

where $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ respectively denote the mean and covariance matrix of embeddings assigned to the label i, and $1(\cdot)$ is an indicator.

The term $\sum_k 1(l_k^I = i)$ in (10) represents the area of instance i and is used to favour large instances. We have found that this term could help to remove tiny instances caused by noise in the point cloud.

The pairwise potential of instance labels $\psi(l_j^I, l_k^I)$ captures geometric properties of surfaces in object instances and is defined as a mixture of Gaussians of the locations, normals, and colour of vertices v_j and v_k. In particular,

$$\psi(l_j^I, l_k^I) = \omega_{j,k} \exp\left(-\frac{\|\mathbf{l}_j - \mathbf{l}_k\|_2^2}{2\lambda_1^2} - \frac{\|\mathbf{n}_j - \mathbf{n}_k\|_2^2}{2\lambda_2^2} - \frac{\|\mathbf{c}_j - \mathbf{c}_k\|_2^2}{2\lambda_3^2}\right) \tag{11}$$

where $\omega_{j,k}$ is presented in (9).

The term $\phi(s, i)$ in (6) associates the semantic-based potentials with instance-based potentials and encourages the consistency between semantic and instance labels. For instance, if two vertices are assigned to the same object instance, they should be assigned to the same object class. Technically, if we compute a histogram h_i of frequencies of semantic labels s for all vertices of object instance i, we can define $\phi(s, i)$ based on the mutual information between s and i as,

$$\phi(s, i) = -h_i(s) \log h_i(s) \tag{12}$$

where $h_i(s)$ is the frequency that semantic label s occurs in vertices whose instance label is i.

As shown in (12), given an instance label i, the sum of $\phi(s, i)$ over all semantic labels $s \in S$ is the information entropy of the labels s w.r.t. the object instance i, *i.e.*, $\sum_{s \in S} \phi(s, i) = -\sum_{s \in S} h_i(s) \log h_i(s)$. A good labeling, therefore, should minimise such entropy, leading to low variation of semantic labels within the same object instance. Since the energy $E(L^S, L^I|V)$ in (6) sums over all semantic labels s and instance labels i, it would favour highly consistent labelings.

3.3. Variational Inference

The minimisation of $E(L^S, L^I|V)$ in (6) is equivalent to the maximisation of the posterior conditional $p(L^S, L^I|V)$ which is intractable to be solved using a naive implementation. In this paper, we adopt mean field variational approach to solve this optimisation problem [43]. In general, the idea of mean field variational inference is to approximate the probability distribution $p(L^S, L^I|V)$ by a variational distribution $Q(L^S, L^I)$ that can be fully factorised over all random variables in (L^S, L^I), *i.e.*, $Q(L^S, L^I) = \prod_j Q_j(l_j^S, l_j^I)$.

However, the factorisation of $Q(L^S, L^I)$ over all pairs in (L^S, L^I) induces a computational complexity of $|S| \times |I|$ per vertex. In addition, since our proposed MV-CRF model is fully connected, message passing steps used in conventional implementation of mean field approximation require quadratic complexity in the number of random variables (*i.e.*, $2N$). Fortunately, since our pairwise potentials, defined in (8) and (11), are expressed in Gaussians, message passing steps can be performed efficiently via applying convolution operations with Gaussian filters on downsampled versions of Q, followed by upsampling [19]. Truncated Gaussians can be also be used to approximate these Gaussian filters to further speed up the message passing process [29].

We first assume that L^S and L^I are independent in the joint variational distribution $Q(L^S, L^I)$, and hence $Q(L^S, L^I)$ can be decomposed as,

$$Q(L^S, L^I) = \left[\prod_{j=1}^N Q_j^S(l_j^S)\right]\left[\prod_{j=1}^N Q_j^I(l_j^I)\right] \tag{13}$$

The assumption in (13) allows us to derive mean field update equations for semantic and instance variational distributions Q^S and Q^L.

Since the term $\sum_{s \in S}\sum_{i \in I} \phi(s, i)$ in (6) is not expressed in relative to the index j, for convenience to the computation of mean field updates, for each vertex v_j, we define a new term m_j as,

$$m_j = \frac{\sum_{s \in S} h_{l_j^I}(s) \log h_{l_j^I}(s)}{\sum_{v_k \in V} 1(l_k^I = l_j^I)} \tag{14}$$

By using m_j, the term $\sum_{s \in S}\sum_{i \in I} \phi(s, i)$ in (6) can be rewritten as,

$$\sum_{s \in S}\sum_{i \in I} \phi(s, i) = \sum_{v_j \in V} m_j \tag{15}$$

We then obtain mean field updates,

$$Q_j^S(l_j^S = s) \leftarrow \frac{1}{Z_j} \exp\left[-\varphi(l_j^S = s) - \sum_{s' \in S}\sum_{k \neq j} Q_k^S(l_k^S = s')\varphi(l_j^S, l_k^S) - m_j\right], \tag{16}$$

and

$$Q_j^I(l_j^I = i) \leftarrow \frac{1}{Z_j} \exp\Big[-\psi(l_j^I = i) - \sum_{i' \in I} \sum_{k \neq j} Q_k^I(l_k^I = i')\psi(l_j^I, l_k^I) - m_j \Big] \tag{17}$$

where Z_j is the partition function that makes $Q(L^S, L^I)$ a probability mass function during the optimisation.

4. Experiments

4.1. Experimental Setup

Our MT-PNet was implemented in PyTorch. We trained our network using the SGD optimiser. The learning rate was set to 0.01 and decay rate was set to 0.5 after every 50 epochs. The training took 10 hours on a single NVIDIA TITAN X graphics card.

For the joint optimisation of semantic and instance labeling, we initialised the semantic and instance labels for 3D vertices as follows. Semantic labels with associated classification scores were obtained directly from MT-PNet. Embeddings for all 3D vertices were also extracted. Initial instance labels were then determined by applying the mean shift algorithm [4] on the embeddings. The bandwidth of mean shift was set to the margin of the push force δ_d in (4). We set $\delta_d = 1.5$ and found this setting achieved the best performance. In addition, when setting the bandwidth to lower values, our performance will drop due to over-segmentation. We note that the number of clusters generated by the mean shift algorithm may be much larger than the true number of instances since we allow over-segmentation. After the joint optimisation step, we only maintain instances that pertain at least one vertex.

Input of our MT-PNet is a point cloud of 4,096 points. To handle large-scale scenes, an input point cloud was divided into overlapping windows, each of which roughly contains 4,096 points. Each window was fed to our MT-PNet to extract instance embeddings. The embeddings from all the windows were merged using the `BlockMerging` procedure in SGPN [44]. Joint optimisation was then applied on the entire scene. Finally, we employ non-maximal suppression to yield the final semantic-instance predictions.

4.2. Datasets

We conducted all experiments on two datasets: S3DIS [1] and SceneNN [15]. S3DIS is a 3D scene dataset that includes large-scale scans of indoor spaces at building level. On this dataset, we performed experiments at the provided disjoint spaces, which were typically parsed to about 10–80 object instances. The objects were annotated with 13 categories. We followed the original train/test split in [1]. Since S3DIS does not include normals of 3D vertices, we simplified (11) with only location and colour.

SceneNN [15] is a scene meshes dataset of indoor scenes with cluttered objects at room scale. Their semantic segmentation follows NYU-D v2 [37] category set, which has 40 semantic classes. On this dataset, we followed the train/test split by Hua *et al*. [16]. Similar to S3DIS, the semantic and instance segmentation were done on overlapping windows.

4.3. Evaluation and Comparison

In this section, we provide a comprehensive evaluation of our method and its variants, and comparisons with existing methods in both semantic and instance segmentation tasks. Several results of our method are shown in Figure 3.

Ablation study. We study the effectiveness of joint semantic-instance segmentation compared with its individual tasks. This study is done by investigating the role of potentials of the energy of our MV-CRF defined in (6). Specifically, for semantic segmentation, we investigate the use of unary potentials in (7) only and traditional CRFs combining (7) and (8). Similarly, for instance segmentation, we compare the use of (10) only and the combination of (10) and (11). We also measure the performance of the joint task, *i.e*., the whole energy of MV-CRF. Table 1 compares MV-CRF and its variants in both semantic and instance segmentation on S3DIS. Metrics include micro-mean accuracy (mAcc)[1] [38] for semantic segmentation and mAP@0.5 for instance segmentation.

Semantic segmentation		Instance segmentation	
Method	mAcc	Method	mAP@0.5
(7)	86.7	(10)	24.9
(7) + (8)	86.9	(10) + (11)	27.4
MV-CRF	**87.4**	MV-CRF	**36.3**

Table 1. Comparison of our MV-CRF and its variants.

Semantic segmentation. Table 2 and Table 4 show the performance of our proposed method in semantic segmentation on S3DIS and SceneNN dataset, respectively.

In this task, we evaluate the stand-alone performance of MT-PNet, marked as "Ours (MT-PNet)", and when running the full pipeline with MV-CRF, marked as "Ours (MV-CRF)". We also compare our method with other state-of-the-art deep neural networks including PointNet [32], PointwiseCNN [16], and SEGCloud [40]. The evaluation metric is per-class accuracy and micro-mean accuracy.

[1]Micro-mean takes into account the size of classes in calculating the average accuracy and thus is often used for unbalanced data. In our context, micro-mean accuracy is equivalent to the overall accuracy that is often used in semantic segmentation.

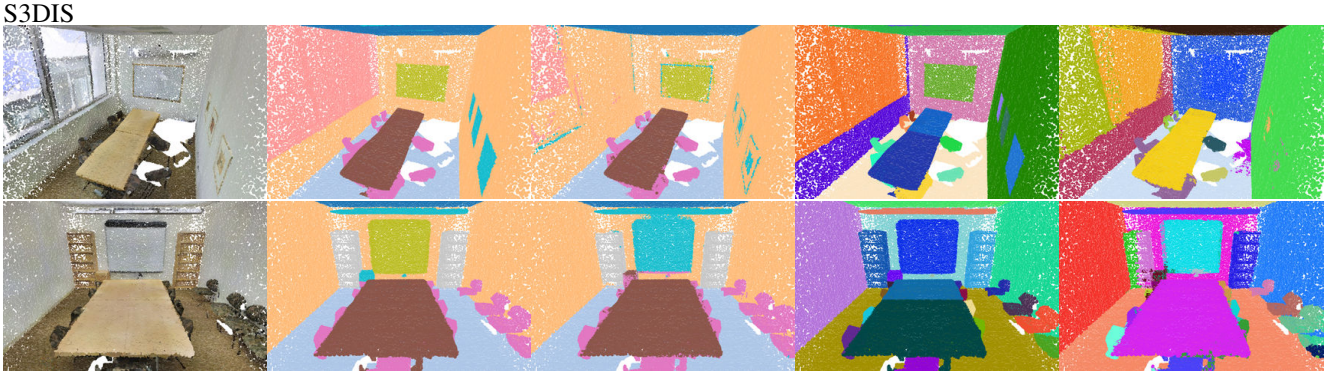
Method	mAcc	ceiling	floor	wall	window	door	table	chair	sofa	bookcase	board	clutter
PointNet [32]	78.6	88.8	97.3	69.8	46.3	10.8	52.6	58.9	40.3	5.9	26.4	33.2
Pointwise [16]	81.5	97.9	99.3	92.7	49.6	**50.6**	74.1	58.2	0	39.3	0	61.1
SEGCloud [40]	80.8	90.1	96.1	69.9	38.4	23.1	75.9	70.4	**58.4**	40.9	13	41.6
Ours (MT-PNet)	86.7	97.4	99.6	92.7	**60.1**	26.4	**80.8**	83.7	23.7	61.1	**55.2**	**70.6**
Ours (MV-CRF)	**87.4**	**98.4**	**99.6**	**94.4**	59.7	24.9	80.6	**84.9**	30	**63.0**	52.5	70.5

Table 2. Semantic segmentation results on S3DIS. Here we also show the stand-alone performance of MT-PNet, and when running the full pipeline with MV-CRF.

Method	mAP	ceiling	floor	wall	window	door	table	chair	sofa	bookcase	board	clutter
Armeni *et al*. [1]	-	71.6	88.7	72.9	25.9	54.1	46	16.2	6.8	54.7	3.9	-
SGPN [44]	54.4	**79.4**	66.3	**88.8**	**66.6**	**56.8**	**46.9**	**40.8**	6.4	**47.6**	11.1	-
Ours (MT-PNet)	24.9	71.5	78.4	28.3	24.4	3.5	12.1	36.2	10	12.6	34.5	12.8
Ours (MV-CRF)	36.3	76.9	**83.6**	32.2	51.4	7.2	16.3	23.6	**16.7**	21.8	**52.1**	**13.4**

Table 3. Instance segmentation results on S3DIS. Here we also show the stand-alone performance of MT-PNet, and when running the full pipeline with MV-CRF. Note that results from Armeni *et al*. are on 3D bounding boxes instead of point clouds.

S3DIS



SceneNN

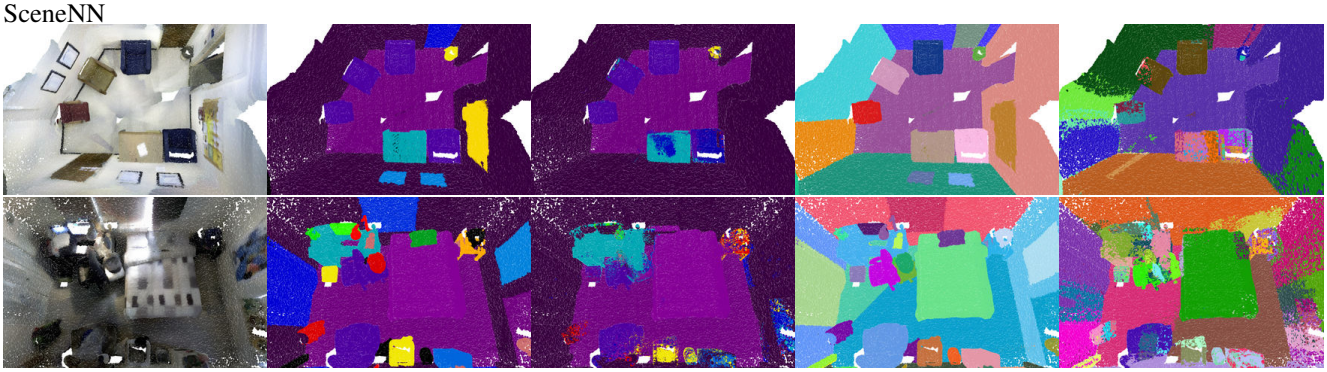


Figure 3. Semantic and instance segmentation results. From left to right: input point cloud, ground truth of semantic segmentation, our semantic segmentation result, ground truth of instance segmentation, our instance segmentation result. For semantic segmentation, different colours represent different categories. For instance segmentation, different colours represent different instances.

Experimental results show that our proposed MT-PNet significantly outperforms its original architecture (*i.e*., PointNet [32]), and the improvement comes from the multi-task architecture. To confirm this, we performed an experiment where we trained our MT-PNet with the instance embedding branch disabled. The disabled-embedding branch network obtained the same performance with the vanilla PointNet on semantic segmentation task.

As shown in Table 2 and Table 4, our MV-CRF also well improves the base results from MT-PNet and achieves state-of-the-art performance on semantic segmentation. This proves that multi-task learning and joint optimisation can be beneficial. Figure 4 shows a close-up example to illustrate the potential of our MV-CRF in semantic segmentation.

Method	wall	floor	cabinet	bed	chair	sofa	table	desk	tv	prop
Pointwise [16]	93.8	88.6	1.5	11.6	58.6	5.5	23.5	29.5	7.7	5.8
Ours (MT-PNet)	94.2	91.5	9.2	58.4	81.4	10.9	37.3	54.0	**33.3**	**13.2**
Ours (MV-CRF)	**96.0**	**92.4**	**10.0**	**74.6**	**83.0**	**11.0**	**44.5**	**61.7**	24.4	11.1

Table 4. Semantic segmentation results on SceneNN. Here we only show a subset of representative classes of NYUv2, as some of the classes are not presented in SceneNN.

Method	wall	floor	cabinet	bed	chair	sofa	table	desk	tv	prop
Ours (MT-PNet)	13.1	27.3	0.0	15.0	**21.2**	0.0	0.7	0.0	6.0	**2.0**
Ours (MV-CRF)	**13.9**	**44.5**	0.0	**32.9**	12.9	0.0	**5.7**	**10.8**	0.0	0.8

Table 5. Instance segmentation results on SceneNN. Here we only show a subset of representative classes of NYUv2, as some of the classes are not presented in SceneNN.

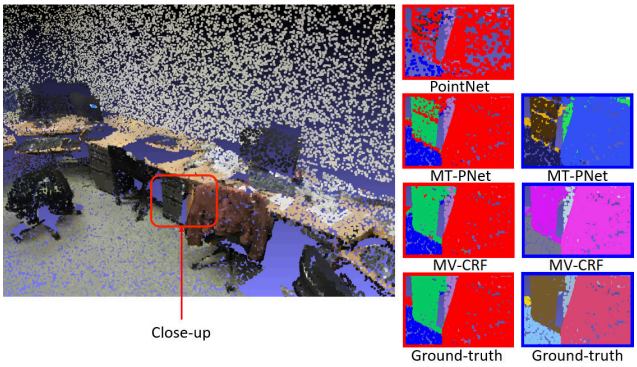


Figure 4. A close-up example of our method. Left: input, middle: semantic segmentation, right: instance segmentation.

Instance segmentation. We consider instance segmentation as object detection and thus evaluate this task using average precision (AP) with IoU threshold at 0.5. To generate object hypotheses, each instance j is granted a confidence score f_j calculated as,

$$f_j = \frac{1}{|V_j|} \log \left\{ \prod_{v_k \in V_j} \left[Q_k^S(l_k^S = s_j) Q_k^I(l_k^I = j) \right] \right\} \quad (18)$$

where V_j is the set of points that have instance label j, and Q_j^S and Q_j^L are defined in (16) and (17) respectively.

Table 3 and Table 5 report the instance segmentation performance of our method on S3DIS and SceneNN dataset, respectively. We refer to the results obtained by applying the mean shift algorithm directly on embeddings from MT-PNet as "Ours (MT-PNet)" and the results of the full pipeline with MV-CRF as "Ours (MV-CRF)". Similarly to semantic segmentation, experimental results show that our MV-CRF significantly boosts up the segmentation performance in comparison to MT-PNet. Figure 4 shows a qualitative comparison of our MV-CRF and other methods in instance segmentation.

We also compare our method with other existing methods including SGPN [44], a recent method for instance segmentation of point clouds, and additional results from Armeni *et al.* [1]. Compared with the state-of-the-art, our method shows clear improvement on some categories, *e.g.*, floor, sofa, board, and clutter. However, it produces low precision segmentation results on other categories such as door. We have found that this is mainly due to the low semantic segmentation accuracy in these categories.

5. Conclusion

Semantic and instance segmentation of point clouds are crucial and fundamental steps in 3D scene understanding. This paper proposes a semantic-instance segmentation method that jointly performs both of the tasks via a novel multi-task pointwise network and a multi-value conditional random field model. The multi-task pointwise network simultaneously learns both the class labels of 3D points and their embedded representations which enable clustering 3D points into object instances. The multi-value conditional random field model integrates both 3D and high-dimensional embedded features to jointly perform both semantic and instance segmentation. We evaluated the proposed method and compared it with existing methods on different challenging indoor datasets. Experimental results favourably showed the advance of our method in comparison to state-of-the-art, and the joint semantic-instance segmentation approach outperformed its individual components.

Acknowledgement. This research project is partially supported by an internal grant from HKUST (R9429) and the MOE SUTD SRG grant (SRG ISTD 2017 131).

References

[1] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3D semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1534–1543, 2016. 2, 3, 6, 7, 8

[2] Anurag Arnab, Sadeep Jayasumana, Shuai Zheng, and Philip H. S. Torr. Higher order conditional random fields in deep neural networks. In *European Conference on Computer Vision (ECCV)*, pages 524–540, 2016. 3, 4

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis & Machine Intelligence (PAMI)*, 40(4):834–848, 2018. 1

[4] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis & Machine Intelligence (PAMI)*, (5):603–619, 2002. 6

[5] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3D reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5828–5839, 2017. 2

[6] Angela Dai, Daniel Ritchie, Martin Bokeloh, Scott Reed, Jürgen Sturm, and Matthias Nießner. Scancomplete: Large-scale scene completion and semantic segmentation for 3D scans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4578–4587, 2018. 2

[7] Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3150–3158, 2016. 3

[8] Bert De Brabandere, Davy Neven, and Luc Van Gool. Semantic instance segmentation with a discriminative loss function. *arXiv preprint arXiv:1708.02551*, 2017. 3, 4

[9] Zhuo Deng and Longin Jan Latecki. Amodal detection of 3D objects: Inferring 3D bounding boxes from 2D ones in rgb-depth images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5762–5770, 2017. 3

[10] Tian Feng, Quang-Trung Truong, Duc Thanh Nguyen, Jing Yu Koh, Lap-Fai Yu, Alexander Binder, and Sai-Kit Yeung. Urban zoning using higher-order markov random fields on multi-view imagery data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 614–630, 2018. 3

[11] S Fidler, Jian Yao, and Raquel Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 702–709, 2012. 3, 4

[12] F. Groh*, P. Wieschollek*, and H. P. A. Lensch. Flex-convolution (million-scale point-cloud learning beyond grid-worlds). In *Computer Vision - ACCV 2018 - 14th Asian Conference on Computer Vision*, 2018. *equal contribution. 2

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2961–2969, 2017. 1, 3

[14] Alexander Hermans, Georgios Floros, and Bastian Leibe. Dense 3D semantic mapping of indoor scenes from rgb-d images. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2631–2638, 2014. 1, 2, 3

[15] Binh-Son Hua, Quang-Hieu Pham, Duc Thanh Nguyen, Minh-Khoi Tran, Lap-Fai Yu, and Sai-Kit Yeung. Scenenn: A scene meshes dataset with annotations. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 92–101, 2016. 2, 6

[16] Binh-Son Hua, Minh-Khoi Tran, and Sai-Kit Yeung. Pointwise convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 984–993, 2018. 1, 2, 3, 6, 7, 8

[17] Qiangui Huang, Weiyue Wang, and Ulrich Neumann. Recurrent slice networks for 3D segmentation of point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2626–2635, 2018. 1, 2

[18] Roman Klokov and Victor Lempitsky. Escape from cells: Deep kd-networks for the recognition of 3D point cloud models. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 863–872, 2017. 1, 2

[19] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in Neural Information Processing Systems (NIPS)*, pages 109–117, 2011. 5

[20] Abhijit Kundu, Yin Li, Frank Dellaert, Fuxin Li, and James M Rehg. Joint semantic segmentation and 3D reconstruction from monocular video. In *European Conference on Computer Vision (ECCV)*, pages 703–718, 2014. 3

[21] L'ubor Ladickỳ, Paul Sturgess, Karteek Alahari, Chris Russell, and Philip H. S. Torr. What, where and how many? combining object detectors and crfs. In *European Conference on Computer Vision (ECCV)*, pages 424–437, 2010. 3, 4

[22] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4558–4567, 2018. 1, 2

[23] Jiaxin Li, Ben M Chen, and Gim Hee Lee. So-net: Self-organizing network for point cloud analysis. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9397–9406, 2018. 1, 2

[24] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. In *Advances in Neural Information Processing Systems (NIPS)*, pages 828–838, 2018. 1, 2

[25] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep continuous fusion for multi-sensor 3D object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 641–656, 2018. 3

[26] Daniel Maturana and Sebastian Scherer. Voxnet: A 3D convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 922–928, 2015. 1, 2

[27] John McCormac, Ankur Handa, Andrew Davison, and Stefan Leutenegger. Semanticfusion: Dense 3D semantic mapping with convolutional neural networks. In *2017 IEEE International Conference on Robotics and automation (ICRA)*, pages 4628–4635, 2017. 2, 3

[28] Duc Thanh Nguyen, Binh-Son Hua, Khoi Tran, Quang-Hieu Pham, and Sai-Kit Yeung. A field model for repairing 3D shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5676–5684, 2016. 1, 2

[29] Sylvain Paris and Frédo Durand. A fast approximation of the bilateral filter using a signal processing approach. In *European Conference on Computer Vision (ECCV)*, pages 568–580, 2006. 5

[30] Quang-Hieu Pham, Binh-Son Hua, Duc Thanh Nguyen, and Sai-Kit Yeung. Real-time progressive 3D semantic segmentation for indoor scenes. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1089–1098, 2019. 1, 2, 3, 4

[31] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3D object detection from rgb-d data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 918–927, 2018. 1, 3

[32] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 652–660, 2017. 2, 3, 6, 7

[33] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems (NIPS)*, pages 5099–5108, 2017. 1, 2

[34] Xiaojuan Qi, Renjie Liao, Jiaya Jia, Sanja Fidler, and Raquel Urtasun. 3D graph neural networks for rgbd semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 5199–5208, 2017. 3

[35] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. Octnet: Learning deep 3D representations at high resolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3577–3586, 2017. 2

[36] Hayko Riemenschneider, András Bódis-Szomorú, Julien Weissenberg, and Luc Van Gool. Learning where to classify in multi-view semantic segmentation. In *European Conference on Computer Vision (ECCV)*, pages 516–532, 2014. 2

[37] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision (ECCV)*, pages 746–760, 2012. 2, 6

[38] Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427–437, 2009. 6

[39] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1746–1754, 2017. 2

[40] Lyne Tchapmi, Christopher Choy, Iro Armeni, JunYoung Gwak, and Silvio Savarese. Segcloud: Semantic segmentation of 3D point clouds. In *2017 International Conference on 3D Vision (3DV)*, pages 537–547, 2017. 2, 6, 7

[41] Julien P.C. Valentin, Sunando Sengupta, Jonathan Warrell, Ali Shahrokni, and Philip H. S. Torr. Mesh based semantic modelling for indoor and outdoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2067–2074, 2013. 2, 3

[42] Vibhav Vineet, Ondrej Miksik, Morten Lidegaard, Matthias Nießner, Stuart Golodetz, Victor A Prisacariu, Olaf Kähler, David W Murray, Shahram Izadi, Patrick Pérez, et al. Incremental dense semantic stereo fusion for large-scale semantic scene reconstruction. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 75–82, 2015. 1, 2, 3

[43] Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008. 5

[44] Weiyue Wang, Ronald Yu, Qiangui Huang, and Ulrich Neumann. Sgpn: Similarity group proposal network for 3D point cloud instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2569–2578, 2018. 3, 6, 7, 8

[45] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *arXiv preprint arXiv:1801.07829*, 2018. 1, 2

[46] Daniel Wolf, Johann Prankl, and Markus Vincze. Fast semantic segmentation of 3D point clouds using a dense CRF with learned parameters. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4867–4873, 2015. 1, 2, 3

[47] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3D shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920, 2015. 1, 2

[48] Yifan Xu, Tianqi Fan, Mingye Xu, Long Zeng, and Yu Qiao. Spidercnn: Deep learning on point sets with parameterized convolutional filters. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 87–102, 2018. 1, 2

[49] Shichao Yang, Yulan Huang, and Sebastian Scherer. Semantic 3D occupancy mapping through efficient high order crfs. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 590–597, 2017. 2, 3

[50] Hongyuan Zhu, Jiangbo Lu, Jianfei Cai, Jianming Zheng, and Nadia M Thalmann. Multiple foreground recognition and cosegmentation: An object-oriented CRF model with robust higher-order potentials. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 485–492, 2014. 3