

Learning 3D Semantics from Pose-Noisy 2D Images with Hierarchical Full Attention Network

Yuhang He

Department of Computer Science
University of Oxford
Oxford, UK
Email: yuhang.he@cs.ox.ac.uk

Lin Chen

Institute of Photogrammetry and
GeoInformation, Leibniz University Hannover
Hannover, Germany
Email: chen@ipi.uni-hannover.de

Junkun Xie and Long Chen

School of Computer Science and Engineering
Sun Yat-Sen University, China.
xiejk3@mail2.sysu.edu.cn
long.chen@ia.ac.cn

Abstract—We propose a novel framework to learn 3D point cloud semantics from 2D multi-view image observations containing pose error. On the one hand, directly learning from the massive, unstructured and unordered 3D point cloud is computationally and algorithmically more difficult than learning from compactly-organized and context-rich 2D RGB images. On the other hand, both LiDAR point cloud and RGB images are captured in standard automated-driving datasets. This motivates us to conduct a “task transfer” paradigm so that 3D semantic segmentation benefits from aggregating 2D semantic cues, albeit pose noises are contained in 2D image observations. Among all difficulties, pose noise and erroneous prediction from 2D semantic segmentation approaches are the main challenges for the “task transfer”. To alleviate the influence of those factor, we perceive each 3D point using multi-view images and for each single image a patch observation is associated. Moreover, the semantic labels of a block of neighboring 3D points are predicted simultaneously, enabling us to exploit the point structure prior to further improve the performance. A hierarchical full attention network (HiFANet) is designed to sequentially aggregates patch, bag-of-frames and inter-point semantic cues, with hierarchical attention mechanism tailored for different level of semantic cues. Also, each preceding attention block largely reduces the feature size before feeding to the next attention block, making our framework slim. Experiment results on Semantic-KITTI show that the proposed framework outperforms existing 3D point cloud based methods significantly, it requires much less training data and exhibits tolerance to pose noise. The code is available at <https://github.com/yuhanghe01/HiFANet>.

I. INTRODUCTION

Directly learning from 3D point cloud is difficult. Challenges derive from four main aspects: First, 3D point cloud is massive and a typical Velodyne HDL-64E scan leads to millions of points. Processing such large data is prohibitively expensive for many algorithms and computation sources. Second, 3D point cloud is unstructured and unordered as well. It record neither the physical 3D world texture nor object topology information, which have often been used as important priors by image based environment perception methods [28, 35, 15]. Third, data imbalance issue. Due to the 3D physical world layout that particular categories conquer most of the space, captured 3D point cloud is often dominated by classes such as road, building and sidewalk. Other categories (*i.e.* traffic sign, poles, pedestrian) with minor point cloud presence but vital importance for self-driving driving scenario understanding and high-quality map construction are often overwhelmed by

dominating classes. Lastly, capturing 3D point cloud is a dynamic process, resulting in inconsistent and nonuniform data sampling. Distant objects are much more sparsely sampled than close objects.

The aforementioned difficulties largely restricted 3D point cloud segmentation progress. 3D point cloud processing with deep neural network [33, 34, 18, 45, 22] has thus emerged much later than counterpart task in 2D images [7, 11, 35, 27, 9, 15]. Meanwhile, most self-driving data collection platforms collect 3D point cloud and RGB images simultaneously, with the LiDAR scanner and camera being pre-calibrated and synchronized to perceive the scene. Therefore, we are naturally motivated to transfer 3D point cloud segmentation to its 2D image based counterpart (we call “task transfer”) so that the segmentation of point cloud can largely benefit from various matured 2D image semantic segmentation networks. Specifically, we exploit features arising from 2D image semantic segmentation result to predict 3D point cloud semantics.

The feasibility of such “task transfer” basically lies in the fact that, given the LiDAR-Camera pose, we can project a 3D point to the 2D image plane to get its 2D pixel correspondence. However, such seemly-fascinating “task transfer” comes with a price: In real-scenario, LiDAR-Camera pose is often noisy so accurate 3D-2D correspondences are non-guaranteed. In addition, view-angle change easily results in distorted image observation. Moreover, 2D semantic segmentation method may also give erroneous predictions.

To tackle the aforementioned challenges, we first propose to perceive each 3D point from multi-view images so that bag-of-frame observations for each single 3D point are obtained. Multi-view image observation reduces the impact of the unfavoured view-angle as it introduces extra semantic cues. Moreover, instead of looking into single-pixel of an image, we focus on a small patch-area around the pixel. The patch observation strategy mitigates 3D-2D correspondence error led by pose noise and further enables neural network to learn pose noise tolerant representation in a data-driven way. Moreover, we process a local group of spatially or temporally close 3D points at the same time, so that we can exploit 3D points structure prior (*i.e.* two points’ spatial location). Actually, the local 3D point group and the corresponding 2D observation can be treated as seq2seq learning problem [37], where one sequence

is 2D image and the other is 3D point cloud. To accommodate these different data representation properties, we propose a hierarchical fully attention network (HiFANet) to sequentially and hierarchically aggregate the patch observation, bag-of-frame observation and inter-point structural prior to infer the 3D semantics. Such hierarchical attention blocks design enables the neural network to learn to efficiently aggregate semantics at different levels. Moreover, the preceding attention block naturally reduces the feature representation size before feeding it to the next attention block, so the whole framework is slim by design.

In sum, our contribution is three fold: first, we propose to transfer 3D point semantic segmentation problem to its counterpart in 2D images. Second, to counteract the pose noise impact, we propose to associate each single 3D point with multi-view patch observation so that the neural network can learn to tolerate pose inaccuracy. Third, we formulate it as a seq2seq problem so that we can best exploit the structural prior arising from both 3D point cloud and 2D images to improve the performance.

II. RELATED WORK

3D semantic segmentation can be divided into three main categories: point-based, voxel-based and 2D projection based methods [16, 43].

Point based methods compute the features from points and can be categorized into three sub-classes [16]: Multi Layer Perceptron (MLP), point convolution and graph convolution based methods. MLP based method apply MLP directly on points to learn features, such as PointNet [33], HRNN [46], PointNet++ [34], PointWeb [50]. In comparison, point convolution based methods apply convolution on individual point. Representative works in this group are PointwiseCNN [20], PCNN [40], PointConv [42], RandLA-Net [19] and PolarNet [48]. In the third class, the points are connected with graph structure, graph convolution is further applied to capture more meaningful local information. Example works include DeepGCNs [25], AGCN [44], HDGCN [26] and 3DContextNet [47].

In voxel based methods, voxels divide 3D space into volumetric grids, which are used as input for 3D convolutional neural networks. The voxel used is either uniform [21, 8, 30] or non-uniform [36, 13]. Methods in this group are restricted by the fact that the computation burden fast grows with the scale of scene. Consequently, the usage of those methods in large scale becomes impractical.

In projection based methods, point cloud is projected into synthetic but multi-view image planes and then 2D CNNs are used by each view, finally semantic results from multiple views are aggregated [23, 14, 6, 17]. However, this idea is restricted by misinterpretation stem from sparse sampling of 3D points. Our work shares the similar idea to convert 3D point cloud to 2D plane, but we exploit 2D RGB images to assist 3D semantic segmentation and we rely on 2D semantic segmentation to predict 3D semantics.

In 2D semantic segmentation, FCN [29] is one of the first works using deep neural network for semantic segmentation by replacing the fully connected layer with fully convolution layers. The following works, e.g., SegNet [1] and [32], use more sophisticated way to encode the input image and decode the latent representation so that images are better segmented. Obtaining features at multiple scale is manipulated either at convolution kernel level or through pyramid structure. The former leads to the method of using dilated convolution and representative works are DeepLabV2 [4] and DeepLabV3[5]. The latter is implemented in PSPN [49] and [12]. Also, attention mechanisms are used to weight features softly for semantic segmentation task in [3]. In this paper, we make use of the network proposed in [51] as our base feature extractor, since it uses synthetic predicting to scale up training data and the trained label is also robust, benefiting from the usage of the boundary relaxation strategy proposed in that paper.

This paper utilizes features from multi-view patches sampled from camera images, which are not accurately aligned with 3D point cloud, to benefit the semantic segmentation of 3D point cloud. In this context, the central issue is how to aggregate multi view image features in a sophisticated way so that 3D points can be better separated in the feature space spanned by those aggregated features.

III. PROBLEM FORMULATION

We have a sequence of N 3D point cloud frames $\mathbf{P} = \{P_1, P_2, \dots, P_N\}$, and framewise associated 3D point semantic label \mathbf{C} and RGB image \mathbf{I} . Such data is collected by platform where LiDAR scanner and camera are carefully synchronized and pre-calibrated with noisy pose information $P_o = [R|t]$ (rotation matrix R and translation t). Moreover, the relative pose between any two neighboring point cloud frames can be obtained via IMU system. With the noisy pose, we can theoretically project any 3D point to any image plane. Off-the-shelf image semantic segmentation method [51] is adopted to get semantic result \mathbf{S} for each image, each pixel of which consists of categorical semantic label and semantic-aware representation r . Our goal is to train a model \mathcal{F} parameterized by θ to predict point cloud semantics from images $\mathbf{C} = \mathcal{F}(\mathbf{I}, \mathbf{S} | P_o, \theta)$.

IV. HIERARCHICAL FULL ATTENTION NETWORK

The fundamental idea of designing our framework is two-fold: “task transfer” which learns 3D point cloud semantics from 2D images; further address accompanying challenges brought by the “task transfer” through a “learning” perspective by fully exploiting the potential of deep neural networks in a hierarchical way. Specifically, given the pose information between any 3D point cloud frame and any 2D image, we can obtain N patch observations $\{\mathcal{P}_1, \dots, \mathcal{P}_N\}$ for each 3D point by projecting it to its neighboring image frames (we call bag-of-frames), where a patch observation \mathcal{P}_i indicates a $k \times k$ squared patch centered at the pixel $[u_x, u_y]$ of the 3D point’s i -th observation image frame. $[u_x, u_y]$ corresponds to the 3D point projection location with noisy pose information. In

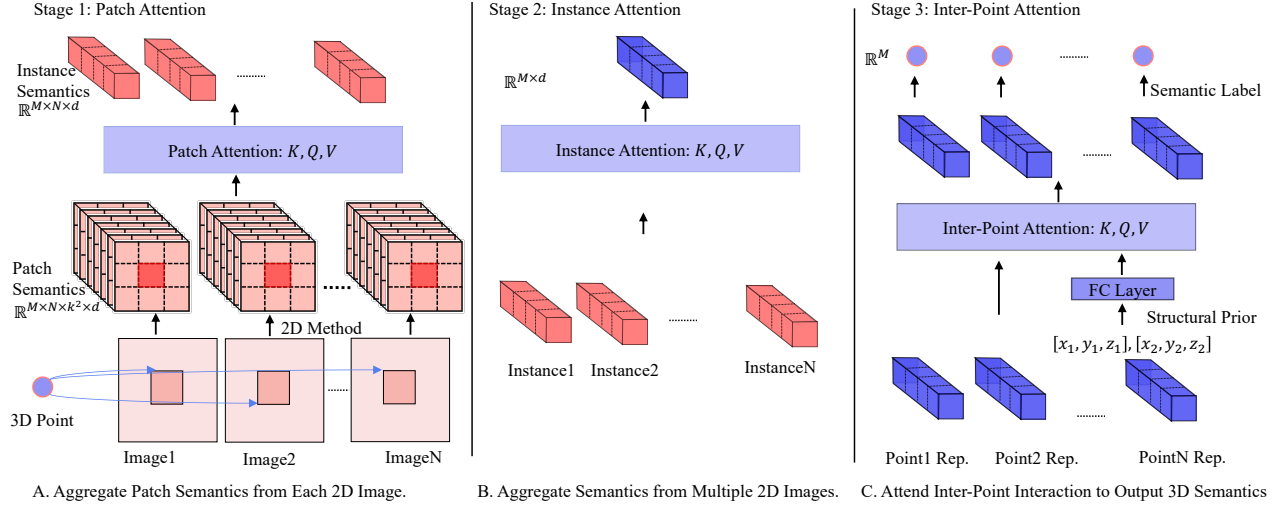


Fig. 1. HiFANet pipeline: Given the pose, we project M 3D points to their nearest top- N RGB images to get $k \times k$ patch observations. Off-the-shelf 2D image segmentation model is trained to get each patch’s semantic feature representation as well as categorical semantic labels. HiFANet is a three-stage hierarchical fully attentive network. It first learns to aggregate patch representation into an instance representation (left image), then aggregates multiple image instances into one point-wise representation (middle image), and finally an inter-point attention module to attend structural and feature interaction among 3D points to output per-point elegant semantic feature representation, which is then used to predict the ultimate semantic label.

the meantime, a pre-trained 2D image semantic segmentation model is available, so we can get both the categorical semantic label s_j and the semantic-guided feature representation r_j for the j -th pixel in the patch. The feature representation r_j can be easily obtained by taking the penultimate layer activation of the model trained on 2D images. So the patch observation can be expressed as,

$$\mathcal{P} = \{(s_1, r_1)_i, \dots, (s_k, r_k)_i\}_{i=1}^N \quad (1)$$

Introducing patch observation instead of single-pixel observation in 2D image is to address pose noise challenge, which we will give detailed discussion in next section. Instead of learning 3D semantic for each 3D point separately, we model M neighboring 3D points simultaneously, which benefits us to use 3D points structure prior to escalate the performance. For example, an intuitive spatial prior is that two spatially-close 3D points are much more likely to share the same semantic label than those lie far apart. In sum, our model takes M 3D points’ Cartesian coordinates as well as each 3D point’s N patch observations as input and outputs each 3D point’s semantic label. It is worth noting that M 3D points forms a point sequence and $M \times N$ image observation forms another image sequence, the whole framework can be treated as a seq2seq task, either spatially or temporally. The framework input simply consists of image-learned semantic information (categorical label or feature presentation), no extra constraint is involved and we do not directly process 3D point cloud.

With “task transfer”, the main task of our framework is to efficiently aggregate semantic clues arising from bag of 2D image frames. To this end, we propose a hierarchical full attention three-stage aggregation mechanism, in which we first learn to aggregate patch observation into an instance

observation (i.e., single pixel observation in an image), and then learn to aggregate multiple instances in the bag-of-frames for each 3D point into 3D point wise observation, and finally attend all the structure prior and interaction between 3D points to output the target semantic label for each single 3D point. Our framework is fully attentive and invariant to images observation order permutation. The hierarchical attention mechanism design has two advantages: it first enables the neural network to fully learn specified attention tailed for different semantic representation, second it aggressively reduces the feature size so that we keep the whole framework slim.

A. Patch Attention for Patch Aggregation

Patch attention tends to aggregate the patch observation into a single-pixel observation. Within each $k \times k$ patch, we call the centered point the principle point and the remaining points are neighboring points. The basic idea behind the patch attention is to attend all points in the patch with a trainable weight before weighted-adding them together to generate one feature. Since the principle point records the most-confident 3D point semantic related feature representation, we add a short-cut connection between the principle point feature and the attended to feature representation. Specifically, given the feature representation $r_i \in \mathbb{R}^{k \times k \times d}$, the principle points lies in $[\frac{k}{2}, \frac{k}{2}]$ and has feature representation f_p of length d , the output feature f_{pa} after patch attention can be expressed as,

$$f_{pa} = \sum_{j=1}^{k \times k} w_j \cdot V_j + f_p \quad (2)$$

where w_j is the learned weight for the j -th point in the patch. To learn the attention weight w , we draw inspiration

from self-attention module [39] to learn a patch Key $K = \mathbb{R}^{k \times k \times d_1}$ and patch Query $Q = \mathbb{R}^{k \times k \times d_1}$ and a patch Value $V = \mathbb{R}^{k \times k \times d}$. The three parts can be efficiently learned via 1×1 2D convolution on the patch observation. To reduce the computation cost (usually $d_1 \ll d$), we set $d_1 = 64$ and $d = 256$. With K and Q we can further compute the scaled dot-product attention where the attended weight w can be obtained by,

$$w = \text{softmax}\left(\frac{Q_p K}{\sqrt{d_1}}\right) \quad (3)$$

Q_p is the principle point query. With Eqn. (3), we can get the weight of each point to the principle point. The patch attention is a self-attention module, it requires no extra supervision and can efficiently attend the final single-pixel observation in with paralleling computation.

B. Instance Attention for Image Aggregation

Instance attention module takes $\mathbb{R}^{M \times N \times d}$ semantic feature as input, and aims to aggregate bag-of-frames features to get 3D point wise feature. We call the aforementioned patch-attention aggregated pixel-wise semantic representation in each image frame as an instance, because it represents an independent observation towards a 3D point. The multiple instances arising from bag-of-frames form an *Instance Set* [41, 24], which means these instances are orderless, the final accurate semantic label may derive from an individual instance or multiple instances combination. To satisfy the instance set property, the instance attention module has to be order-permutation invariant. Commonly seen set-operators include max-pooling and average-pooling. In HiFANet, we first apply a self-attention layer like the patch attention block does to attend each instance by all the remaining instances. Finally, we apply average pooling to merge multiple instances into one instance representation.

C. Inter-point Attention for 3D Points Aggregation

Inter-point attention take $\mathbb{R}^{M \times d}$ semantic feature learned by instance attention module as input. Unlike the previous two attention modules that just focus on per-point semantic feature learning, inter-point attention module fully considers the interaction between 3D points, including the spatial structure interaction and semantic feature interaction. We adopt a Transformer [39] multi-head self-attention like network to construct the inter-point attention module. Specifically, the input feature is fed to learn per-point Key $K = \mathbb{R}^{M \times d_2}$ and per-point Query $Q = \mathbb{R}^{M \times d_2}$ as well as per-point Value $V = \mathbb{R}^{M \times d}$. To involve structural prior, we encode the relative Cartesian position difference between any two 3D points $p_i - p_j$. The Cartesian position difference is further fed to two consecutive fully connection layers to get the structural prior encoding K_{pe} , which is the same size of K . The original Key K is then updated by adding K_{pe} ,

$$K = K + K_{pe} \quad (4)$$

The updated K in Eqn.(4) naturally contains the structural prior. With the Q and updated K , we can compute the attention weight for each single 3D point w.r.t the remaining 3D points, as is shown in Eqn.(3). The attention weight is further applied to combine value V to get the final per-point semantic representation, which is further concatenated with a classification layer for semantic classification.

In sum, HiFANet sequentially and hierarchically aggregates patch semantics, instance semantics and inter-point semantics to learn semantic representation for each 3D point. It is fully attentive and learns compartmentalized and certain attention blocks w.r.t. different aggregation granularity separately. The preceding attention layer largely reduce the feature size before feeding it to the next layer, so the whole neural network is slim. Detailed HiFANet pipeline is shown in Fig. 1.

V. DISCUSSION ON HiFANET DESIGN MOTIVATION

The feasibility of such “task transfer” lies in the availability of the pose information between LiDAR scanner and the camera, which enables us to project 3D point cloud onto the image plane to get each 3D point’s correspondence in the image plane. We hereafter call such correspondence as a 2D observation. The “task transfer” poses three main challenges that may jeopardize the performance.

- 1) Pose noise. Sensor calibration often suffers from internal and external noise. Noisy pose information leads to inaccurate 2D observations. This stays as the most prominent challenge.
- 2) View-angle. Projecting a cluster of point cloud belonging to a specific category (*i.e.* car) to an image plane often leads to distorted 2D observation. In severe cases, it leads to wrong observation due to the occlusion caused by view-angle difference.
- 3) Void projection. While LiDAR scanner scans in 360° , pinhole camera simply captures the forward-facing view. This mismatch of perception field inevitably leads to void projection in which point cloud cannot find observation in one image.

Addressing the above three challenges leads to our proposed framework. To mitigate the pose noise impact, we propose to use patch observation to replace pixel observation. Pixel-wise observation is fragile and sensitive to pose noise, a small change leads to totally different observation. Patch-wise, on the contrary, becomes much more resilient to pose noise because it covers possible observations potentially led by noisy pose. Moreover, introducing patch-wise observation avoids us directly optimizing $[R|t]$ in an iterative way. To address the view-angle and void projection issue, we propose to involve multiple observation arising from different view-angles. With the multi-view observations, we naturally obtain multiple clues for each 3D point.

A. Pose Noise and Patch Observation

The pose between LiDAR scanner coordinate system and camera coordinate system can be formulated as a rotation matrix R and translation T . A 3D point $[x, y, z]$ projects onto

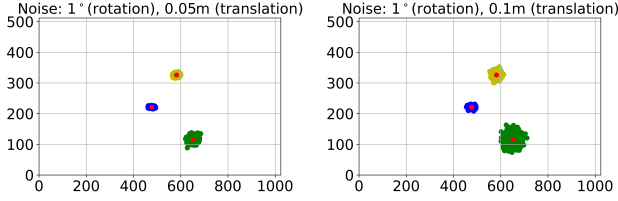


Fig. 2. The influences of pose noise to the projected point coordinates on image plane (in pixel) for 3D world points that are various distant from the camera plane (green: 5m; yellow: 10m; blue: 20m). The simulated noises of rotation angles are 1° (for each rotation angle) for both cases and the translation noise are 0.05m (left), 0.1m (right) for each of the three axes in world coordinate system.

an image plane, the corresponding observation location $[u, v]$ in the 2D image plane is computed by,

$$[u, v, 1] = K[R|T] \cdot [x, y, z, 1]^T \quad (5)$$

Please note that the projected pixel location is normalized by its 3rd dimension. The pose noise of sensor calibration (between laser scanner and camera) renders the location of true projected point uncertain. However, in our approach, a patch is extracted and then the attention is learned to focus on the pixel closest to the true projected points.

In order to investigate the influence of the pose noise on the location of projected points, a toy simulation experiment is provided and illustrated in Fig. 2. As can be observed in Fig. 2, given the translation noise for the calibration between the LiDAR scanner and camera as 10cm and the rotation angle noise as 1° , the projection error on the image plane (1024×512 pixels) is around 40 pixels for near camera object points. Since the patch extracted on each camera view is within a $k \times k$ patch in the downsized feature maps (normally at $1/16$ or $1/32$ resolution), the information encoded in the image is then well preserved for the attention module to discover, although the pose noise exist.

B. View-angle and Void Projection

View-angle easily leads to titled, occluded and even erroneous observation. A 3D point that is observed in one viewpoint (an RGB image) can be obstructed in another neighboring viewpoint. Traditional 3D reconstruction framework like structure-from-motion (SfM [10]) suffer from the same dilemma. The void projection jeopardizes the “task transfer” proposal because it causes large number of 3D points being 2D image unobserved.

To mitigate the two challenges, we propose to observe a single 3D point from multiple view-angles. On the one hand, it reduces the risk of one 3D point being observed at an unfavored view angle. On the other hand, it maximally ensures each 3D point cloud to be observed by at least one 2D image. Moreover, this strategy brings us the advantage of aggregating semantic clues arising from multiple images to better estimate semantics. Multiple view-angles observation can be efficiently aggregated in parallel in HiFANet.

VI. EXPERIMENTS AND RESULTS

We conduct experiment on the Semantic-KITTI dataset [2]. Since we need the inter-frame odometry information to project each 3D point to multiple RGB frames but the official provided test dataset (sequence 11-20) does not provide such information, so we do not follow the official split but instead create the train/test/val split by ourselves and further train the comparing methods with the split dataset from scratch. The same problem applies to other relevant datasets such as Waymo and CityScapes [7], so we just run experiment on Semantic-KITTI dataset in this paper.

Data Preparation We run experiment on sequence 00-10 because the inter-frame odometry information is available for the 11 sequences, with which we can register all point cloud frames from a sequence to a uniform 3D coordinate so that each 3D point can be freely projected to any image plane. There are 13 semantic categories in total: *road*, *side-walk*, *building*, *fence*, *pole*, *traffic sign*, *vegetation*, *terrain*, *person*, *bicyclist*, *car*, *motorcycle* and *bicycle*. Some categories like *road*, *building*, *vegetation* and *terrain* dominate most of the points, whereas the others’ portion is very small. An extra *unlabelled* background category is added. Sequence 06 is selected as test set as it contains all semantic categories and account for 20% data of the whole dataset. Sequence 08 is selected for validation and the remaining 9 sequences serve as training set. To get each 3D point’s N neighboring image observations, we project it to its closest N image planes. N is set as 5 because it then covers 64% of the whole point cloud dataset with patch size $k = 5$ and 3D points number size $M = 10$. Those 3D points that fail to find N image observations are discarded during test but left for training point cloud based models. The image based semantic representation and semantic label are obtained from VideoProp [51] model pre-trained on KITTI dataset [11]. The semantic representation is a 256-d feature. Therefore, the size of patch semantics representation feeds to HiFANet is $5 \times 5 \times 256$. For the evaluation metric, we adopt the standard mIoU and average accuracy [2].

Methods to Compare The first method category we tend to compare is pure 3D point cloud based semantic segmentation method. It helps us to gain an understanding of how far our proposed “task transfer” strategy goes, comparing with directly learning from 3D points. The second method category we compare with is the semantic result giving by deterministically aggregating the category semantic labels predicted by 2D image aggregation method, it gives us an understanding of how good image based semantic prediction methods can perform, by varying the observation number like image number and patch size. The third category is multi-view learning method which means designing neural network to learn from image semantic representations, as our proposed HiFANet does.

Ablation Study we want to figure out the impact of the involvement of patch feature representation, structural prior on the performance. We thus test two HiFANet variants: reduce the patch size to 1 so no patch attention module is

TABLE I
QUANTITATIVE RESULT ON SEMANTIC-KITTI[2] DATASET. B, K AND M MEAN BILLION, THOUSAND AND MILLION, RESPECTIVELY.

Method Category	Method	Train Dataset	Param Num	mIoU (\uparrow)	Average Accuracy (\uparrow)
Point Based Methods	PointNet [33]	2.8 B	3.53 M	0.036	0.105
	PointNet++ [34]	2.8 B	0.97 M	0.055	0.156
	RangeNet++(CRF) [31]	2.8 B	50.38 M	0.500	0.878
	RangeNet++(KNN) [31]	2.8 B	50.38 M	0.512	0.899
	KPConv[38]	2.8 B	18.34 M	0.466	0.868
	RandLANet [18]	2.8 B	1.24 M	0.578	0.913
Image Aggregation Methods	BoF Num = 1	23 K	137 M	0.422	0.845
	BoF Num = 3	23 K	137 M	0.437	0.852
	BoF Num = 5	23 K	137 M	0.436	0.852
	Patch Size = 1	23 K	137 M	0.436	0.850
	Patch Size = 3	23 K	137 M	0.436	0.851
	Patch Size = 5	23 K	137 M	0.436	0.852
Multi-View Learning Methods	AvgPool_FC	0.5 M	0.04 M	0.451	0.872
	HiFANet_noPA	0.5 M	2.5 M	0.537	0.891
	HiFANet_noSP	0.5 M	2.7 M	0.561	0.920
	HiFANet	0.5 M	2.7 M	0.620	0.933

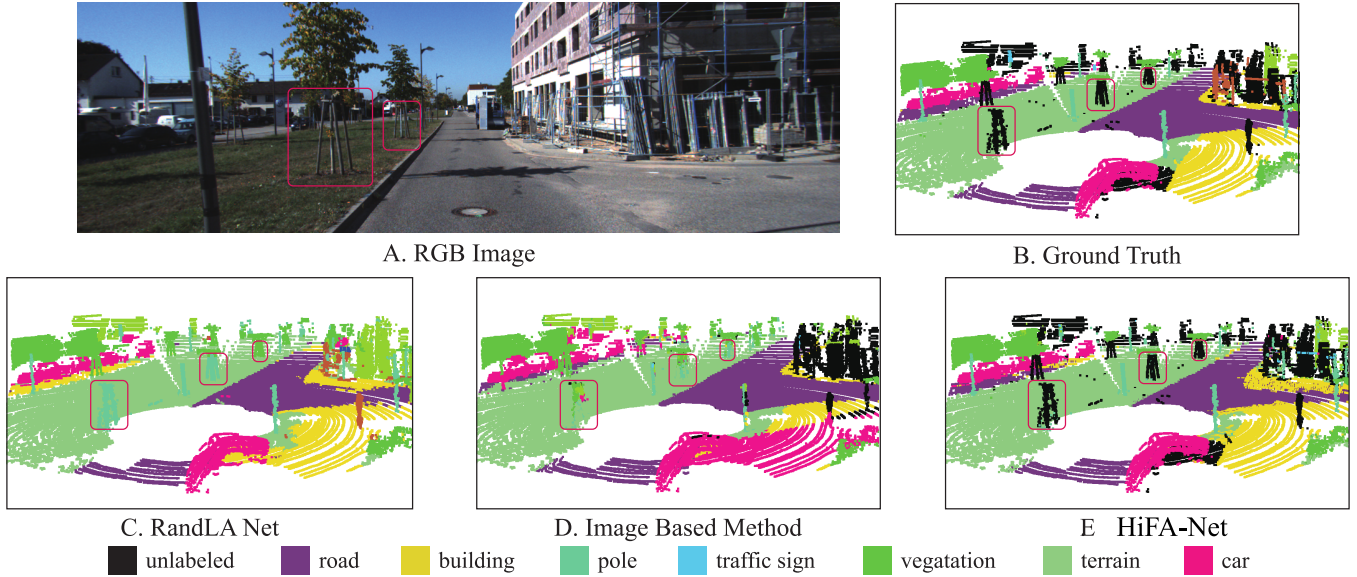


Fig. 3. Close-up visualization of various methods on unlabelled tree stake. While point based method erroneously classifies them as pole and image based method as terrain, HiFANet accurately recognizes it by fully combining 2D image based semantics and 3D structural priors.

applied (**HiFANet_noPA**), no structural prior involvement in inter-point attention module (**HiFANet_noSP**). Moreover, to test the effectiveness of our proposed full attention network, we train another simple semantic aggregation network, in which we simply average-pool all the input feature (patch and instance feature) to get per-point feature, and further concatenate two full connection layer (of size 256, 128) to directly predict the semantic label (**AvgPool_FC**). Please note that AvgPool_FC is a simple neural network and it is order-permutation invariant.

Five most recent 3D point cloud based methods: PointNet [33], PointNet++ [34], RangeNet [31] (two variants, with KNN and CRF), KPConv [38] and RandLANet [18] are selected for comparison study. For image aggregation methods, we simply deterministically choose the semantic label

with maximum occurrence times. Within multi-view learning methods, all HiFANet variants are trained with the same hyper-parameter setting as HiFANet.

Quantitative Result is shown in Table I. We can observe that point cloud based methods training requires much larger number of training dataset than both image aggregation methods and our proposed multi-view learning methods. This shows the advantage of learning semantics from 2D images. The compactly-organized and topology-preserving RGB images enables neural network to learn meaningful semantic representations with much fewer training samples. Within image aggregation methods, involving extra bag-of-frame observations increases the performance, but the performance gain is not prominent due to the view-angle and occlusion challenges. Moreover, expanding the patch size also improves

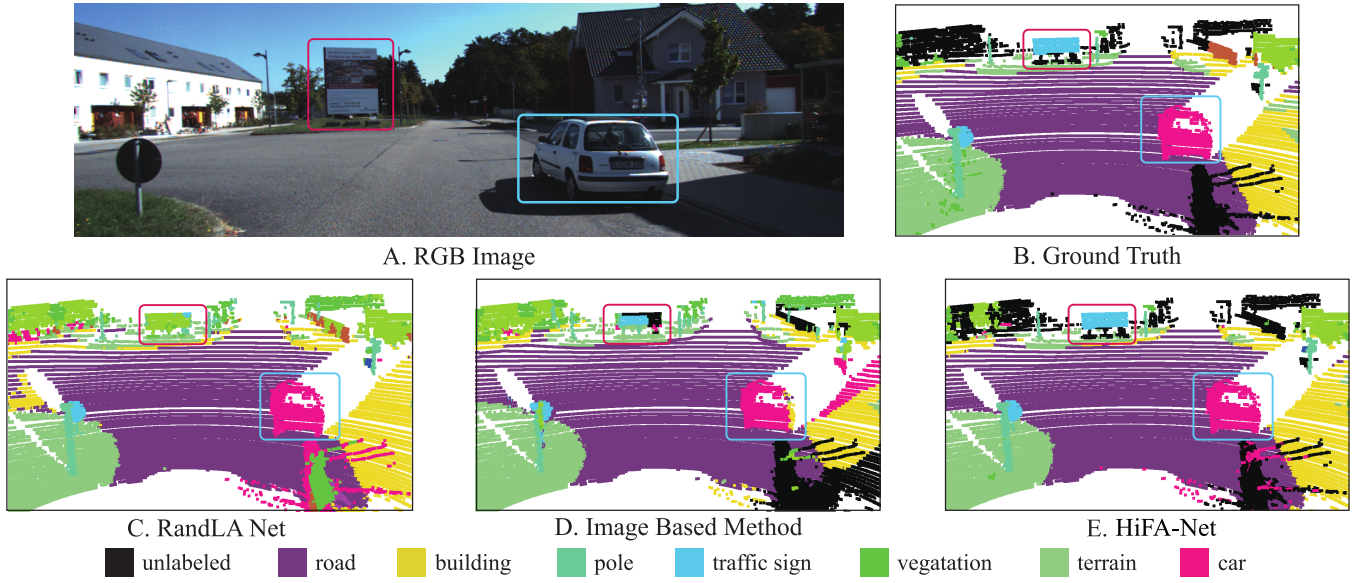


Fig. 4. Global visualization of various methods comparison. While point based method fails to classify traffic sign and image based method generates spatially distributed prediction, HiFANet successfully avoids these dilemmas and gives the right semantics.

the performance, which shows capability of introducing patch-wise observation in mitigating the dilemma caused by observation uncertainty. In sum, aggregating image-predicted semantics can achieve comparable performance than point based methods. It further shows the potential of designing neural network to learn from image learned semantic representations, instead of simply voting them.

Within multi-view learning methods, we can observe that all methods outperform image aggregation methods, showing the advantage of neural network learning over deterministic semantic aggregation. Simply adding several fully connection layers (AvgPool_FC) generates inferior performance than the other three HiFANet variants. This result shows that more advanced semantic aggregation strategy is needed to better aggregate semantic cues arising from multiple image observations. At the same time, either removing the patch attention module or the structural prior module inevitably reduces the performance. Patch observation introduces extra semantic cues in a pose noise sensitive way and structural prior regularizes the whole network training. Finally, HiFANet generates the best performance over all methods, far outweighing other methods by a large margin.

Qualitative Result is shown in Fig. 3 and Fig. 4. In the close-up comparison of tree stakes in Fig. 3, as it is a category falls out of our consideration, it should be regarded as *unlabelled* category. However, 3D point based method RandLANet [18] (sub-figure B.) mixes it with *pole* due to their point cloud representation similarity. Image aggregation method (sub-figure D.) directly predicts it as terrain because of its color similarity and connection with the tree leaves. HiFANet (sub-figure E.), however, fully exploits 3D point structural prior information to predict the correct semantics. For example, the tilted angle of tree stakes over the ground

makes it unlikely to be a pole (which is usually vertical to ground), nor terrain (no angle information).

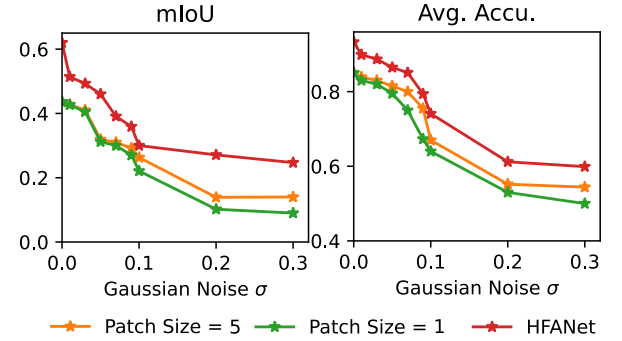


Fig. 5. Pose noise test: performance variation trend under various Gaussian pose noise level.

The global comparison of various methods is shown in Fig. 4. We can observe that point based method (C. RandLANet) failed to predict the large traffic sign (red box in the RGB image) because such samples are rarely seen in training dataset. At the same time, due to the pose noise, image based method distributes car 3D points to large area (see the largely distributed red points in sub-figure D., near the light blue). Our proposed HiFANet can maximally avoid these dilemmas. It obtains semantic representation from RGB images, so it does not require massive training dataset and large presence of all classes. The hierarchical attention design and the involvement of 3D structural prior equip HiFANet with capability to dynamically alleviate the erroneous prediction led by pose noise. In sum, our proposed HiFANet achieves promising performance with relatively small training dataset. It also exhibits pose noise tolerance capability, which is a

TABLE II
DETAILED IOU SCORE FOR EACH CATEGORY ON SEMANTIC-KITTI[2] DATASET

Method	road	side-walk	building	fence	pole	traffic-sign	vegetation	terrain	person	rider	car	motor-cycle	bicycle
PointNet [33]	0.031	0.069	0.113	0.043	0.036	0.022	0.041	0.054	0.000	0.000	0.052	0.002	0.003
PointNet++ [34]	0.066	0.023	0.079	0.042	0.112	0.014	0.036	0.183	0.000	0.002	0.133	0.010	0.000
RangeNet++(CRF) [31]	0.878	0.745	0.742	0.232	0.252	0.313	0.612	0.875	0.088	0.356	0.853	0.375	0.176
RangeNet++(KNN) [31]	0.895	0.769	0.819	0.258	0.333	0.291	0.648	0.896	0.114	0.414	0.856	0.178	0.183
KPCConv [38]	0.738	0.574	0.653	0.244	0.469	0.400	0.533	0.767	0.249	0.696	0.739	0.360	0.000
RandLANet [18]	0.883	0.760	0.883	0.323	0.537	0.319	0.731	0.910	0.216	0.572	0.909	0.470	0.003
Image Based BoF=5	0.888	0.710	0.378	0.154	0.189	0.362	0.598	0.889	0.210	0.055	0.563	0.533	0.146
HiFANet	0.910	0.790	0.903	0.349	0.540	0.374	0.755	0.912	0.247	0.577	0.933	0.547	0.169

TABLE III
HiFANet NETWORK ARCHITECTURE. FC INDICATES FULLY-CONNECTION LAYER, K,Q INDICATES THE KEY AND QUERY IN THE SELF-ATTENTION MODULE.

layer	filter num	output size
Input: [B, 10, 5, 5, 5, 256]		
Patch Attention Module		
K,Q	64, head num = 4	[B, 10, 5, 256]
FeedForward Net	256	[B, 10, 5, 256]
Instance Attention Module		
K,Q	64, head num = 4	[B, 10, 256]
FeedForward Net	256	[B, 10, 256]
InterPoint Attention Module		
K,Q	64, head num = 4	[B, 10, 256]
FeedForward Net	256	[B, 10, 256]
InterPoint Attention: Structural Prior		
FC	128	[B, 10, 128]
FC	256	[B, 10, 256]
Classification Head		
FC	512	[B, 10, 512]
FC	512	[B, 10, 512]
FC	class num	[B, 10, class num]

common challenge in real scenario.

A. More Experimental Result

We report the detailed mIoU and mAP score for each individual class in Table II. We can see from the table that our proposed HiFANet achieves the best performance on most categories. Image based method (with BoF=5) obtains inferior performance on some categories such as car, rider and traffic sign, due to the pose noise. Our proposed HiFANet maximally resists the negative impact of pose noise and thus is capable of obtaining promising performance.

B. Discussion on Pose Noise

We further want to test our proposed HiFANet performance under various pose noise level. To this end, we add Gaussian pose noise to the point-to-image projection matrix in Eqn.5. The Gaussian noise level is controlled by the Gaussian deviation σ (the mean value is set 0). We compare HiFANet with two image aggregation variants: with patch size 1 and 5. Since the introduction of patch observation is to handle pose noise, it helps us to understand patch observation (patch size = 5) resistance to pose noise against the original observation (patch size = 1), and against HiFANet.

The Gaussian pose noise σ is linearly spaced from 0 to 0.3. The result is shown in Fig. 5, from which we can observe that adding more pose noise reduces the performance of all methods. The variant with patch size 1 suffers most while HiFANet maximally mitigates the pose noise impact. It thus shows the advantage of involving patch observation in tackling pose noise and our carefully designed HiFANet is capable of learning pose noise tolerant feature representation.

C. Implementation Detail and Source Code

In HiFANet, we set image observation number as 5, the number of 3D points as 10 and the patch size as 5, so the input size is [10, 5, 5, 5, 256]. The multi-head attention module head number is 4. The total training dataset is more than 100 million, we randomly subsample 0.5 million points. We implement in PyTorch and train with SGD optimizer, the initial learning rate is 0.1 and decays with factor 0.5 every 30 epochs. Batchsize is 64. The network is trained 100 epochs in total. The network architecture is shown in Table III. We use the same hyper-parameter setting to train all other HiFANet variants in our ablation study. For comparing methods, we use their released source code with default or recommended training strategy.

VII. CONCLUSION AND LIMITATION DISCUSSION

We propose a three-stage hierarchical fully attentive network, HiFANet, to label the point cloud semantically. The patch observation strategy and bag-of-frames multi-view observation enable HiFANet to handle point-image projection pose noise. Compared to point cloud based methods, HiFANet requires significantly less amount of data and outperforms point based methods by a large margin. The downside our method is that HiFANet's good performance still depends relatively on the LiDAR-camera pose accuracy. If the pose accuracy drops significantly, HiFANet's performance reduces accordingly. Designing more pose-noise tolerant method thus forms a potential future research direction. Another point is that HiFANet only builds on 2D image observations, a joint learning from both the image and point cloud may further improve the performance.

REFERENCES

- [1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12): 2481–2495, 2017.
- [2] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In *Proc. of the IEEE/CVF International Conf. on Computer Vision (ICCV)*, 2019.
- [3] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016.
- [4] 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 and machine intelligence*, 40(4):834–848, 2017.
- [5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.
- [6] Long Chen, Yuhang He, Jianda Chen, Qingquan Li, and Qin Zou. Transforming a 3-d lidar point cloud into a 2-d dense depth map through a parameter self-adaptive framework. *IEEE Transactions on Intelligent Transportation Systems (TITS)*, 2017.
- [7] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [8] 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.
- [9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
- [10] Yasutaka Furukawa and Jean Ponce. Accurate, Dense, and Robust Multi-View Stereopsis. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, 2007.
- [11] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361, 2012.
- [12] Golnaz Ghiasi and Charles C Fowlkes. Laplacian pyramid reconstruction and refinement for semantic segmentation. In *European conference on computer vision*, pages 519–534. Springer, 2016.
- [13] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9224–9232, 2018.
- [14] Joris Guerry, Alexandre Boulch, Bertrand Le Saux, Julien Moras, Aurélien Plyer, and David Filliat. Snapnet-r: Consistent 3d multi-view semantic labeling for robotics. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 669–678, 2017.
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [16] Yong He, Hongshan Yu, Xiaoyan Liu, Zhengeng Yang, Wei Sun, Yaonan Wang, Qiang Fu, Yanmei Zou, and Ajmal Mian. Deep learning based 3d segmentation: A survey. *arXiv preprint arXiv:2103.05423*, 2021.
- [17] Yuhang He, Long Chen, and Ming Li. Sparse depth map upsampling with rgb image and anisotropic diffusion tensor. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, 2015.
- [18] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [19] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11108–11117, 2020.
- [20] 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.
- [21] Jing Huang and Suya You. Point cloud labeling using 3d convolutional neural network. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2670–2675. IEEE, 2016.
- [22] Loic Landrieu and Martin Simonovski. Large-scale Point Cloud Semantic Segmentation with Superpoint Graphs. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [23] Felix Järemo Lawin, Martin Danelljan, Patrik Tostberg, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Deep projective 3d semantic segmentation.

- In *International Conference on Computer Analysis of Images and Patterns*, pages 95–107. Springer, 2017.
- [24] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosior, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 3744–3753, 2019.
 - [25] Guohao Li, Matthias Muller, Ali Thabet, and Bernard Ghanem. Deepgcns: Can gcns go as deep as cnns? In *Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR)*, pages 9267–9276, 2019.
 - [26] Zhidong Liang, Ming Yang, Liuyuan Deng, Chunxiang Wang, and Bing Wang. Hierarchical depthwise graph convolutional neural network for 3d semantic segmentation of point clouds. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8152–8158. IEEE, 2019.
 - [27] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014.
 - [28] Wei Liu, Dragomir Anguelov, D. Erhan, Christian Szegedy, S. Reed, Cheng-Yang Fu, and Alex Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision (ECCV)*, 2016.
 - [29] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
 - [30] Hsien-Yu Meng, Lin Gao, Yu-Kun Lai, and Dinesh Manocha. Vv-net: Voxel vae net with group convolutions for point cloud segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8500–8508, 2019.
 - [31] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. RangeNet++: Fast and Accurate LiDAR Semantic Segmentation. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.
 - [32] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015.
 - [33] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2017.
 - [34] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017.
 - [35] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.
 - [36] 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.
 - [37] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2014.
 - [38] Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschard, Beatriz Marcotegui, François Goulette, and Leonidas J. Guibas. Kpconv: Flexible and deformable convolution for point clouds. *Proceedings of the IEEE International Conference on Computer Vision (CVPR)*, 2019.
 - [39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
 - [40] Shenlong Wang, Simon Suo, Wei-Chiu Ma, Andrei Pokrovsky, and Raquel Urtasun. Deep parametric continuous convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2589–2597, 2018.
 - [41] Yun Wang, Juncheng Li, and Florian Metze. A comparison of five multiple instance learning pooling functions for sound event detection with weak labeling. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.
 - [42] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9621–9630, 2019.
 - [43] Yuxing Xie, Jiaojiao Tian, and Xiao Xiang Zhu. Linking points with labels in 3d: A review of point cloud semantic segmentation. *IEEE Geoscience and Remote Sensing Magazine*, 8(4):38–59, 2020.
 - [44] Zhuoyang Xie, Junzhou Chen, and Bo Peng. Point clouds learning with attention-based graph convolution networks. *Neurocomputing*, 402:245–255, 2020.
 - [45] Bo Yang, Jianan Wang, Ronald Clark, Qingyong Hu, Sen Wang, Andrew Markham, and Niki Trigoni. Learning Object Bounding Boxes for 3D Instance Segmentation on Point Clouds. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.
 - [46] Xiaoqing Ye, Jiamao Li, Hexiao Huang, Liang Du, and Xiaolin Zhang. 3d recurrent neural networks with context fusion for point cloud semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 403–417, 2018.
 - [47] Wei Zeng and Theo Gevers. 3dcontextnet: Kd tree guided hierarchical learning of point clouds using local and global contextual cues. In *European Conference on Computer Vision (ECCV)*, pages 314–330. Springer, 2019.

2018.

- [48] Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, Boqing Gong, and Hassan Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9601–9610, 2020.
- [49] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.
- [50] Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia. Pointweb: Enhancing local neighborhood features for point cloud processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5565–5573, 2019.
- [51] Yi Zhu, Karan Sapra, Fitsum A. Reda, J. Shih, Kevin, Shawn Newsam, Andrew Tao, and Bryan Catanzaro. Improving semantic segmentation via video propagation and label relaxation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.