Hybrid CNN-LSTM Architecture for LiDAR Point Clouds Semantic Segmentation

Shuhuan Wen, Tao Wang, and Sheng Tao

***Abstract*—The semantic segmentation of LiDAR in the outdoor environment is still an open problem in the field of automatic driving. Although the emerging development and technological advancements, it remains challenging for three reasons: 1) uneven distribution of LiDAR Points in 3D space; 2) the same object can be represented by different point clouds sequences; 3) no unique geometric information compared with the pixels of 2D RGB images. In order to tackle all the challenges mentioned above, we propose a new LiDAR point clouds semantic segmentation algorithm, Hybrid CNN-LSTM, which is composed of an efficient point clouds feature processing method and a novel neural network structure. Inspired by representing the point clouds as a fixed-length vector in PolarNet, we first convert the 3D point clouds into pseudo image. In order to better represent the features of small objects, we input the pseudo image into long short-term memory (LSTM) network according to the spatial filling curve. We design a new neural network structure, which combines the features of different channels generated by convolutional neural network and long short-term memory network. Experiments show that our method has higher semantic segmentation accuracy in comparison with state-of-the-art algorithms on SemanticKITTI dataset. Theoretically, we provide an analysis to understand why our network can better segment the small objects with sparse point clouds features.**

***Index Terms*—LiDAR point clouds semantic segmentation, deep learning, autonomous driving, LSTM, CNN.**

I. Introduction

THE autonomous vehicle is high-speed moving object, which requires real-time and accurate perception of the surrounding dynamic environment. LiDAR has the advantages of long measuring distance, high measurement resolution (high precision), and it can also present a three-dimensional point clouds image. Therefore, the environment sensing algorithm of LiDAR [1] point clouds has been widely used in autonomous driving. Autonomous driving generally has three modules [2]: environment perception, behavior decision, and motion control. Environmental perception is a prerequisite for autonomous vehicles to achieve obstacle avoidance and path planning. Therefore, we study the environment sensing algorithm about LiDAR in this paper. With the development of 3D LiDAR sensor technology, many research results have been obtained to segment 3D LiDAR point clouds and identify ground and obstacles. Traditional methods of ground segmentation usually rely on artificially designed features and rules [3]. In recent years, with the development of deep learning in image vision, a class of point clouds target detection methods based on deep learning models has been proposed to extract point clouds features to approximate the end-to-end processing [4]. The research on the semantic segmentation of point clouds based on various deep learning algorithms is still in its infancy. Because the LiDAR point clouds are discrete and sparse in space, quantization causes irregular and inconsistent edges in 2D coordinate system. Today, the research focus is to keep the original details of the point clouds as much as possible without losing speed.

A. Motivation

In the second II, we introduce the mainstream algorithm of LiDAR semantic segmentation in the field of automatic driving, and introduce the inspiration of our algorithm. Therefore, the major goal of this paper is to propose a neural network algorithm based on PointNet [5] that is more suitable for dealing with discrete LiDAR point clouds. Because LiDAR obtains environmental information through circular rotation, so it is easy to cause the objects far from the LiDAR to be blocked by other objects. Therefore, in order to compensate for the lack of LiDAR Points on the occluded object, we propose to use the memory function of the long short-term memory network for feature extraction. In order to deal with the uneven distribution of LiDAR point clouds and reduce the computational power consumption of the algorithm, we also carry out a random sampling process for the point clouds. We design a new network structure, which combines the features from CNN network and LSTM network. We validate our approach on SemanticKITTI dataset [18], which is widely used in the field of automatic driving. We also analyze why the neural network structure proposed in this paper can outperform others.

B. Contribution

This paper proposes a new algorithm to solve the LiDAR semantic segmentation in a pseudo image fashion. The key contributions of our work are summarized as follows:

Manuscript received November 14, 2021; accepted January 23, 2022. Date of publication February 24, 2022; date of current version April 5, 2022. This letter was recommended for publication by Associate Editor P. Castillo and Editor P. Pounds upon evaluation of the reviewers' comments. The work was supported in part by the National Natural Science Foundation of China (NSFC) under Project 61773333, in part by the National Natural Science Foundation of China and the Royal Society of Britain (NSFC-RS) under Project 62111530148, and in part by the China Scholarship Council (CSC) under Project 201908130016. *(Corresponding author: Shuhuan Wen)*

Shuhuan Wen, Tao Wang, and Sheng Tao are with the Engineering Research Center of the Ministry of Education for Intelligent Control System and Intelligent Equipment, Yanshan University, Qinhuangdao 066004, China, and also with the Key Laboratory of Industrial Computer Control Engineering of Hebei Province, Yanshan University, Qinhuangdao 066004, China (e-mail: swen@ysu.edu.cn; 82368157@qq.com; 13785071098@163.com).

Digital Object Identifier 10.1109/LRA.2022.3153899

We propose a novel LiDAR point clouds processing method that considers the relationship between regions. We use PointNet to process the LiDAR point clouds in each grid to get the pseudo image. Since LiDAR point clouds are sparse in space and discontinuous due to occlusion, quantization creates irregular and inconsistent edges in 2D representations. Therefore, we divide the pseudo image into patches and input them into LSTM network according to Hilbert curve sequence to extract features more continuously and accurately.

We propose a novel semantic segmentation networks framework for LiDAR point clouds, which combines spatial context and time-series information to solve the disadvantage of losing details in the pooling layer of CNN network. The experimental results show that our network has significantly improved the accuracy of LiDAR point clouds semantic segmentation compared with other methods.

II. Related Work

A. Geometric Relationship-Based Algorithm

Traditional traversable area detection uses visual information, such as semantic segmentation based on neural networks [6]. Although vision sensors are closer to human environmental perception and can provide abundant environmental information, cameras rely on environment light and vehicle lighting. They are vulnerable to weather and lighting conditions. LiDAR can emit laser beams to sense the environment, which makes it more adaptable to the environment. For the use of LiDAR data in autonomous driving systems, one way is to use the geometric information of range and the scan lines of single-scan LiDAR to compute the unevenness field. The authors in [3] use height difference and slope along a column to detect obstacles, pitch, and depression in the ground. This system has been tested at NIST and has successfully detected obstacles and regions of concealment while driving cross country at speeds of 35 km/h. Satish *et al.* [7] apply the geometry of the rings both in radial and transverse directions to determine traversable and obstacle regions. It is robust against slight variations in the pitch and roll of the robot and terrain slopes. However, the disadvantage of this algorithm based on geometric information is that it can only segment terrain and obstacles but lacks the semantic information required for autonomous driving. Compared with geometric information, semantic information contains more detailed environmental information, such as car, person and so on. Therefore, semantic information is the basis of realizing higher-level automatic driving. Therefore, we mainly study the semantic segmentation algorithm of LiDAR.

B. Clustering-based Algorithm

Other traditional methods that use LiDAR to judge the traversable area, such as [8], generally have the following steps: segmentation of ground points, point clouds clustering, feature extraction, and classification. These methods usually rely on artificially designed rules based on the geometric features between points, such as setting some thresholds and surface normal. The scalability of these algorithms is relatively poor, and the multi-stage processing flow means that compound errors may occur.

C. Projection-based Network

In recent years, many semantic segmentation methods of LiDAR point clouds based on deep learning have been proposed [9]. However, before processing point clouds using the neural network, how to represent disordered point clouds is a research hotspot. For semantic segmentation of LiDAR point clouds, the current popular 3-D point clouds representation methods include three-dimensional voxelization [10], and bird's-eye-view projection [11]. The authors in [12] use convolutional neural network (Convolutional neural network, CNN) combined with Conditional Random Field (CRF) structure. This method projects a 3D LiDAR point clouds onto a spherical surface and uses 2D CNN network to predict the point-by-point label of the point clouds. Moreover, the conditional random field is used to correct the label mapping output by the CNN network. Although this algorithm can process point clouds quickly, the segmentation accuracy is still low and requires a lot of training data. Wu *et al.* [4] proposed SqueezeSegV2 by improving the basic SqueezeSeg [12] model, using a novel region location module to reduce the sensitivity to noise loss, and replacing the original model's cross-entropy loss as focal loss [13]. Chen *et al.* [14] combines visual and LiDAR point clouds information. The network consists of two sub-networks for generating 3D object proposals and the other for multi-view feature fusion. Vocalizing the point clouds will change the original features of the data, cause unnecessary data loss, and increase the amount of calculation.

D. Voxel-based Networks

The authors in [10] divide the point clouds uniformly into 3D voxels, and then use the voxel feature coding layer to convert them into standard feature representations. Like the deep learning method in image vision, it does not require artificially designed target features. However, the disadvantages of this algorithm are poor convergence of network parameters and high computational cost.

E. Point-based Networks

Qi *et al.* [5] proposed PointNet to process point clouds that uses the original point clouds as the input to retain the spatial features of the point clouds. The network maps each point from low-dimensional to high-dimensional, and then represents the global feature through the maximum pool. The method of operating on all points at once ignores the local information of the environment. The authors in [15] improved the network structure of PointNet, first extracting local features from the geometric structure in a small range, then expanding the field, and extracting higher-level features based on these local features. Because of the uneven distribution of the point clouds, multi-scale grouping (MSG) and multi-resolution grouping (Multi-resolution grouping, MRG) solutions are also used in this method. The author in [16] proposes a novel 3D object detection model, Sparse-PointNet, which detects 3D objects from multi-modal sensor data. For a given LiDAR point clouds scan, Zhang *et al.* [17] quantize the points into grids using polar BEV coordinates so that the points with the same ground truth label can be evenly assigned to the same grid.

F. Practical Considerations

As a result, how to design a model to reliably and quickly implement semantic segmentation for LiDAR point clouds is still a problem to be solved. In this paper, we propose a new neural network structure, which combines LSTM network and CNN network for LiDAR semantic segmentation. Different from the existing approaches, the new network structure we proposed is more suitable to represent the relationship between local features and global features. Experiments show that the network structure is more suitable for semantic segmentation of sparse and unevenly distributed LiDAR point clouds.

The rest of the paper is structured as follows. Section III provides the proposed architecture, including the deep learning algorithm and space-filling curve. Then experimental results are reported in Section IV. Finally, a conclusion is given in Section V.

III. Methodology

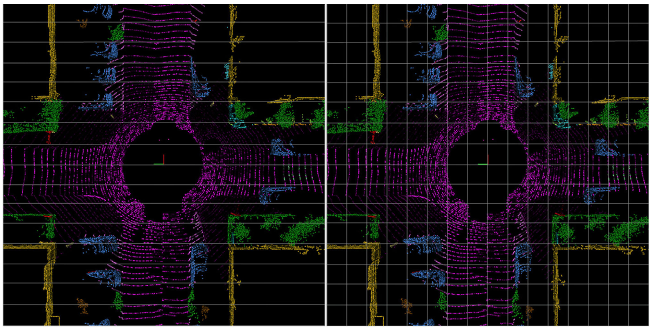
We first give the content of this paper, which is to improve the accuracy of LiDAR semantic segmentation. Then, in order to better represent the LiDAR features, we propose a method to input the patches of pseudo images into LSTM network according to the sequence of Hilbert curves. Finally, the framework of our proposed method is given in section III. D.

A. Problem Statement

In SemanticKITTI dataset [18], N represents the LiDAR scan of a given training set [17], $\{P_i, L_i | i = 1, \ldots, N\}$, $\mathrm{P_i} \in \mathrm{R}^{\mathrm{i}\times 4}$ is the *i*-th point of the current scan, which contains four dimensions of information (*x, y, z,* reflection). The *(x, y, z)* is the Cartesian coordinate of the point relative to the LiDAR sensor. The reflection is the intensity of returning laser beam. $\mathrm{L_i}$ represents the label of $\mathrm{P_i}$. Our goal is to learn a network model that takes the original point clouds as input so that the point clouds label of the network output matches the ground truth label of the dataset as much as possible [19].

B. Input the Patches of Pseudo Images Into LSTM Network

Recurrent neural network (RNN) [20] is used to process sequence data. Compared with the general neural network, it can process the data of the sequence change. LSTM is a special RNN, mainly to solve the problem of gradient disappearance and gradient explosion during long sequence training [21]. Compared to RNN network, LSTM network can perform better in longer sequences. At present, CNN network is commonly used in the field of image semantic segmentation. However, the max pooling layer of CNN network has a significant disadvantage that ignores the relationship between the local and the global, and so loses valuable information. Now the commonly used method is to design a multi-layer convolution structure. Convolution layers with different sizes have different perception fields. Finally, the features of each layer are combined to perform semantic segmentation.



(a) Traditional patching method (b) Our patching method

Fig. 1. Different image patching methods for SemanticKITTI [18].

Because the input received by the LSTM network is one-dimensional sequence data, it is necessary to reduce the dimensionality of the two-dimensional pseudo-image. The method of dividing a LiDAR point clouds scan into patches in this paper is shown in Fig. 1 [18]. The traditional LSTM network method is to input each row of pixels of the image as a node into LSTM network in turn, as shown in Fig. 1(a) [21]. However, this blocking method cannot express the relation between the parts of the image very well, and an object is easily assigned to different patches by mistake. Our blocking method is shown in Fig. 1(b) [21]. We use the output of PointNet as a pseudo-image, and the pseudo-image size is $\mathrm{L_{Pseudo\ Image}} \times \mathrm{W_{Pseudo\ Image}} \times \mathrm{H_{Pseudo\ Image}}$(We use 512∗512∗32). $\mathrm{L_{Pseudo\ Image}}$ is the length, $\mathrm{W_{Pseudo\ Image}}$ is the width, $\mathrm{H_{Pseudo\ Image}}$ is the number of channels to allow an object to be allocated to an LSTM network node as much as possible. Integrate the features of each patch as a node of LSTM network. To enable the pseudo image processed by LSTM network to be combined with a specific convolutional layer in the CNN network, suppose the length and width of this convolutional layer are $\mathrm{L_{CNN}} \times \mathrm{W_{CNN}}$ (We choose to make the value of $\mathrm{L_{CNN}}$ equal to $\mathrm{W_{CNN}}$), The length of the LSTM network sequence $\mathrm{Seq_{LSTM}}$ as shown in equation (1):

$$\mathrm{Se\ q_{LSTM}} = \mathrm{L_{CNN}}\ \times \mathrm{W_{CNN}} \qquad (1)$$

The length (width) of each patch is as shown in equation (2):

$$\mathrm{L_{Patch}} = \frac{\mathrm{L_{Pseudo\ Image}}}{\mathrm{L_{CNN}}\ (\mathrm{W_{CNN}})}\ . \qquad (2)$$

We input the divided data into LSTM network according to a specific sequence to strengthen the connection between the local and the global. The discussion of the space-filling curve is in Section III. C.

C. Hilbert Curve

We input the data of each patch into LSTM network according to a kind of spatial curve. The space-filling curve can map the

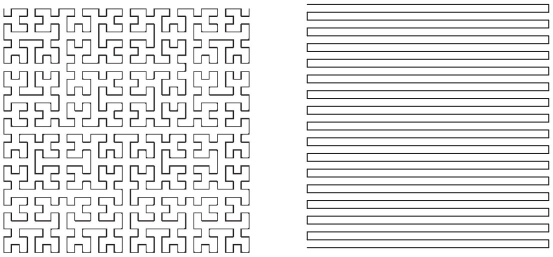

(a) Hilbert curve [24]
(b) Snake curve [23]

Fig. 2. Two different space-filling curves.

data that are not in good order from the high-dimensional space to the one-dimensional space so that the spatial locality of the patches will be better preserved [22]. Traditional space-filling curves include Snake curve [23] and the Hilbert curve [24]. The Hilbert curve can linearly connect each discrete patch in two or higher dimensions. Its main advantage is to preserve the consistency between adjacent patches [25]. So, we follow the Hilbert curve and input the pseudo-image into LSTM network [26] in blocks. Fig. 2(a) shows the working process of the Hilbert curve. In order to prove the advantages of the Hilbert curve, we designed a controlled experiment with the same grid size and input the data into LSTM network in a different order, as shown in Fig. 2(b). We analyze the difference between the two space-filling curves in Section IV. D. The test results proved that Hilbert curve show excellent spatial continuity, and the corresponding network also achieved higher mIoU.

D. Features Fusion Network

For a LiDAR scan, our purpose is to perform semantic segmentation on the point clouds to complete environment perception. The proposed framework is based on [21] and is shown in Fig. 3. Our network can be divided into four parts. 1) A learnable simplified PointNet, and 2) LSTM network with Hilbert curve sequence, and 3) convolutional encoder, and 4) decoder network. One LiDAR scan is encoded by a PointNet [5] to obtain a 2D pseudo image. Convolutional encoder-decoder network and Long Short-Term Memory networks use the pseudo image to learn spatial features. The LiDAR point clouds are then segmented into 19 categories using the label of the SemanticKITTI dataset [18].

For the first part (The original LiDAR point clouds are processed by PointNet to obtain the pseudo image.), we use a simplified PointNet to transform points in each grid into a fix-length representation vector as a pseudo image. We then divide pseudo images extracted using PointNet into patches. For the second part (Bidirectional LSTM network and CNN network.), we use the Hilbert curve (discussed in Section. III-C) to determine the ordering of the patches to feed into LSTM cells. Because the current output of LSTM cell is related to the previous state, in order to better represent the correlation between each patch, we use LSTM network to learn the feature of each patch of the pseudo image. Then, the feature maps are generated from the LSTM cell output, which will be combined with the feature maps from the CNN network. The convolutional encoder consists of max-pooling, convolution layer, batch normalization, and activation function. For the third part (Decoder network), the feature map is fed to a decoder that can provide a more refined representation of different objects in the environment. The decoding network consists of upsampling layers and convolution layers. In the upsampling layer, we use bilinear interpolation. We perform end-to-end training to classify each pixel of the pseudo image.

Algorithm 1: The Hybrid CNN-LSTM Architecture.

```
Input: LiDAR scan with size:
    $\mathrm{L}_{\text{Piont Clouds}} \times \mathrm{W}_{\text{Piont Clouds}} \times \mathrm{H}_{\text{Piont Clouds}}$
    Ground truth label with size:
    $\mathrm{L}_{\text{Piont Clouds}} \times \mathrm{W}_{\text{Piont Clouds}} \times \mathrm{H}_{\text{Piont Clouds}}$
    Hilbert curve sequence (5th order).
Output: Predicted labels for all points.
For each LiDAR scan do:
    According to PointNet [5], the original point clouds
    data are processed to obtain the pseudo image:
    $L_{Pseudo\ Image} \times W_{Pseudo\ Image} \times H_{Pseudo\ Image}$
    Input the pseudo image into the CNN encoding
    network to get: $F_{\mathrm{CNN}}$
    Input the patches of the pseudo image into LSTM
    network (in Hilbert curve sequence) to get: $\mathrm{F}_{\mathrm{LSTM}}$
    Concatenate $\mathrm{F}_{\mathrm{CNN}}$ and $\mathrm{F}_{\mathrm{LSTM}}$ to get: $\mathrm{I}_{\mathrm{Decoder}}$
    Input $\mathrm{I}_{\mathrm{Decoder}}$ into the decoding network to obtain
    predicted labels for all points
End
Return Labels
```

We compute pixel-based segmentation loss, which is then minimized by utilizing the backpropagation algorithm. After optimization, we find the optimal set of parameters for the network used for the semantic segmentation of the LiDAR point clouds for a given test set. The procedure of the novel neural network structure we proposed is presented in Algorithm 1. We express the output of LSTM network as $\mathrm{S}_{\mathrm{LSTM}}$. The result obtained by sampling under CNN network is $\mathrm{S}_{\mathrm{CNN}}$. Then the input of the decoding channel of the neural network $\mathrm{I}_{\mathrm{Decoder}}$ is [21]

$$\mathrm{I}_{\mathrm{Decoder}} = \mathrm{S}_{\mathrm{LSTM}} + \mathrm{S}_{\mathrm{CNN}}. \tag{3}$$

IV. Experiments

We present the dataset used in our paper, the experiments settings, and the ablation experiments result in this section. In the current research field of LiDAR semantic segmentation, the mainstream dataset is SemanticKITTI [18]. Because they annotated all sequences of the SemanticKITTI Vision Odometry Benchmark and provided dense point-wise annotations for the complete 360° field-of-view of the employed automotive LiDAR, it has 22 annotated sequences the odometry benchmark of the SemanticKITTI Vision Benchmark [19] consisting of over 43000 scans. There are 19 challenging classes in total. So, we use the SemanticKITTI dataset to verify the superiority of our method in some respects.

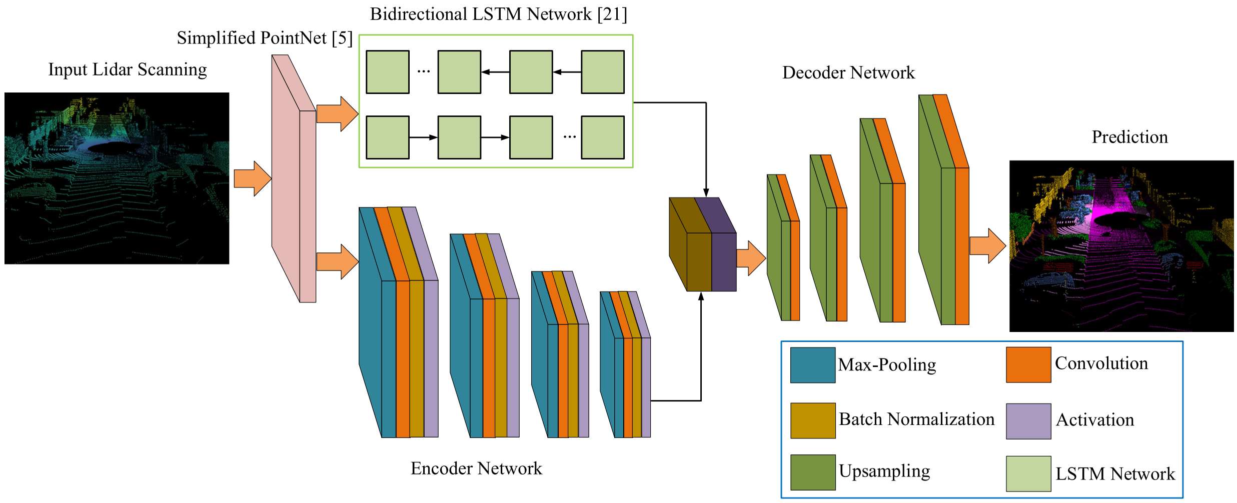


Fig. 3. Proposed CNN-LSTM network architecture.

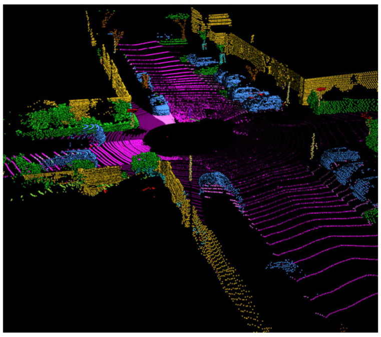


Fig. 4. The dataset used in this paper [18].

A. Dataset

We use the SemanticKITTI datasets in our experiments. SemanticKITTI has 23201 full 3D scans for training and 20351 for testing showing urban traffic, residential areas, highway scenes, and countryside roads around Karlsruhe, Germany. To compare with other benchmark methods, we use the same sequence segmentation method as SemanticKITTI, 00 to 10 as the training set, 11 to 21 as the test set. It has more than 100 000 points per scan on average, and each scan is collected by a single Velodyne HDL-64E laser scanner shown in Fig. 4. By analyzing the related files of the SemanticKITTI dataset, we found that the proportions of different categories in the total number of point clouds vary greatly. The class vegetation accounts for 26.68% of the total number of point clouds, while motorcyclist only accounts for 0.00000055% of the total number of point clouds. Obviously, the distribution of point clouds is extremely uneven, which is a challenging for semantic segmentation algorithms. We show several SemanticKITTI semantic segmentation benchmarks to verify our algorithm.

After analyzing the spatial distribution of points could in the SemanticKITTI training split, we fixed the hybrid CNN-LSTM network grid spaces to be [distance: 0∼50m, z: −3∼9m], which include more than 99% of points for each scan on average. Points exceeding this range are assigned to the closest BEV grid cell. In addition, we set the grid sizes as [512, 512, [32].

B. Baselines and Metric

RandLA [27]: Hu *et al.* proposed a point clouds semantic segmentation algorithm for large scenes (Outdoor environment) and used random sampling. A local feature sampler is proposed to reduce the information lost in random sampling, including local spatial encoding (LocSE) and attention pooling.

PolarNet [17]: Zhang *et al.* proposed polar bird's-eye-view representation and ring convolution and proved by experiments that the distribution of point clouds in polar coordinates is more uniform than that in Cartesian coordinates improved the accuracy of semantic segmentation.

C. SemanticKITTI Segmentation Experiment

We report the *IoU* and *mIoU* on the entire validation split. The class *i*'s intersection over union $\mathrm{IoU_i}$, which refers to the intersection of the class prediction and ground truth divided by their union [17]:

$$\mathrm{Io\ U_i} = \frac{\mathrm{P_i} \cap \mathrm{G_i}}{\mathrm{P_i} \cup \mathrm{G_i}}\ . \tag{4}$$

P: Prediction.
G: Ground Truth.

mIoU is the mean overall semantic classes of class intersection over union [28]:

$$\mathrm{mIoU} = \frac{1}{\mathrm{k}} \sum_{\mathrm{i}=1}^{\mathrm{k}} \frac{\mathrm{P_i} \cap \mathrm{G_i}}{\mathrm{P_i} \cup \mathrm{G_i}} \tag{5}$$

Given the unique features of real-time requirements for LiDAR application under the automatic driving background, we also

TABLE I
SEGMENTATION RESULTS ON TEST SPLIT OF SEMANTICKITTI [18]

Model	Size	FPS	mIoU (%)	Per class IoU (%)																		
				car	bicycle	motorcycle	truck	bus	person	bicyclist	motorcyclist	road	parking	sidewalk	other-ground	building	fence	vegetation	trunk	terrain	pole	traffic-sign
PointNet [5]	50K pts	2	14.6	46.3	1.3	0.3	0.1	0.8	0.2	0.2	0.0	61.6	15.8	35.7	1.4	41.4	12.9	31.0	4.6	17.6	2.4	3.7
PointNet++ [15]	50K pts	0.1	20.1	53.7	1.9	0.2	0.9	0.2	0.9	1.0	0.0	72.0	18.7	41.8	5.6	62.3	16.9	46.5	13.8	30.0	6.0	8.9
TangentConv [30]	50K pts	0.3	35.9	86.8	1.3	12.7	11.6	10.2	17.1	20.2	0.5	82.9	15.2	61.7	9.0	82.8	44.2	75.5	42.5	55.5	30.2	22.2
LatticeNet [31]	50K pts	7	52.9	92.9	16.6	22.2	26.6	21.4	35.6	43.0	**46.0**	90.0	59.4	74.1	22.0	88.2	58.8	81.7	63.6	63.1	51.9	48.4
RandLA [27]	50K pts	5	53.9	**94.2**	26.0	25.8	40.1	**38.9**	49.2	48.2	7.2	90.7	60.3	73.7	20.4	86.9	56.3	81.4	61.3	66.8	49.2	47.7
RangeNet++ [32]	64×2048	11	52.2	91.4	25.7	34.4	25.7	23.0	38.3	38.8	4.8	**91.8**	**65.0**	75.2	**27.8**	87.4	58.6	80.5	55.1	64.6	47.9	55.9
SqueezeSeg [12]	64×2048	**49**	29.5	68.8	16.0	4.1	3.3	3.6	12.9	13.1	0.9	85.4	26.9	54.3	4.5	57.4	29.0	60.0	24.3	53.7	17.5	24.5
SqueezeSegV2 [4]	64×2048	37	39.7	81.8	18.5	17.9	13.4	14.0	20.1	25.1	3.9	88.6	45.8	67.6	17.7	73.7	41.1	71.8	35.8	60.2	20.2	36.3
SqueezeSegV3 [33]	64×2048	6	55.9	92.5	38.7	36.5	29.6	33.0	45.6	46.2	20.1	91.7	63.4	74.8	26.4	89.0	59.4	82.0	58.7	65.4	49.6	**58.9**
PolarNet [17]	[480, 360, 32]	16	54.3	93.8	40.3	30.1	22.9	28.5	43.2	40.2	5.6	90.8	61.7	74.4	21.7	**90.0**	**61.3**	84.0	**65.5**	67.8	51.8	57.5
Ours	[512, 512, 32]	11	**56.9**	92.6	**45.7**	**49.6**	**48.6**	30.2	**53.8**	**74.6**	9.2	90.7	23.3	**75.7**	17.6	**90.0**	51.3	**87.1**	60.8	**75.4**	**63.9**	41.5

report models' maximum frames-per-second with the largest possible batch size (FPS) to evaluate the real-time performance of the algorithm and the input size of each model. We implement the new network architecture proposed in this paper in PyTorch [29].

Table I shows the performance comparison between our approaches and multiple baselines. In general, our neural network model has better segmentation accuracy. The results show that the hybrid CNN- LSTM network based on U-net [34] is better than the state-of-the-art method for point could semantic segmentation network, especially for the classes with fewer data. The segmentation accuracy of our algorithm in the classes: "bicycle", "motorcycle", "truck", "person", "bicyclist" and "pole" is much higher than other neural network models. These classes have a common feature: they are relatively smaller compared to class "car", "bus", and roads. Because of the uneven distribution of LiDAR point clouds, the number of points on objects of different sizes varies greatly, which results in the difficulty of semantic segmentation due to the lack of descriptions of enough points for smaller objects. Because we first divide the original point clouds into grids and then extract features, so the small object may be assigned to different grids.

We input the pseudo image patches into LSTM network to extract features according to the Hilbert curve, so that the features of small objects distributed on different grids can be better expressed by the neural network we proposed. Therefore, our network model can better represent small-volume objects, thereby improving the accuracy of semantic segmentation, which matches with the range and details preserving properties of the hybrid CNN-LSTM network.

At the same time, our network also improves the segmentation accuracy of class "vegetation", "terrain" and "pole". Similar to other algorithms, we also notice low performance on "motorcyclist", "other-ground", and "parking". It is very difficult to identify the class motorcyclist because it is often largely occluded and rarely appears in the verification set. According to the definition of SemanticKITTI, "other ground" is essentially sidewalk/terrain-like ground but used for other purposes, such as traffic islands. For the class parking, our method is worse than other networks. Because of the use of LSTM network and Hilbert curve, the features of class parking are closely to the features of surrounding objects, such as road, which makes it difficult for neural network to identify them.

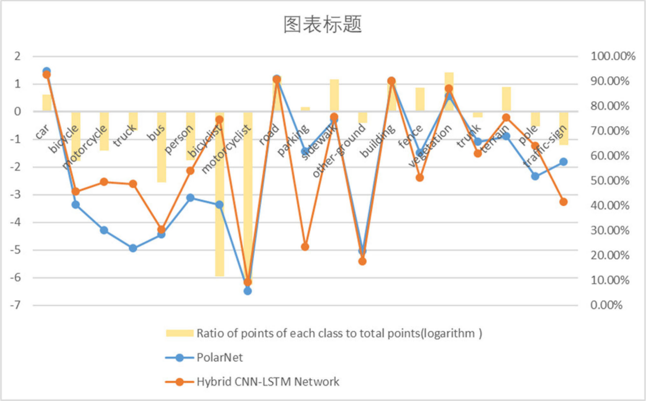
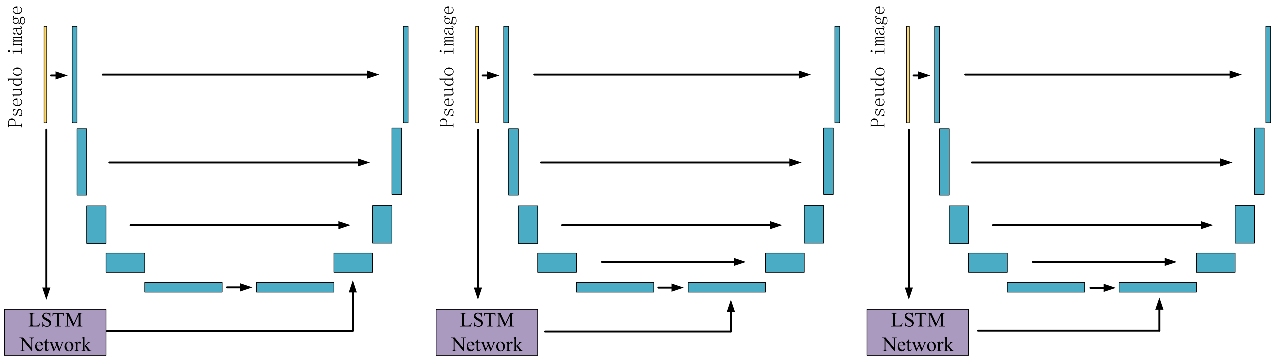


Fig. 5. Analyze the segmentation results and the proportion of various classes in the total point clouds together.

In addition, although our algorithm improves the accuracy of semantic segmentation, the running speed of the algorithm decreases. As shown in FPS in Table I. Compared with PolarNet, because our neural network adopts double flow structure, the large number of parameters leads to higher latency.

In Fig. 5, we analyze the proportion of each class in SemanticKITTI to the total number of point clouds and the segmentation effect of different networks. The histogram represents the logarithm of the ratio of point clouds occupied by a certain class. The orange curve represents the segmentation IoU of our hybrid CNN-LSTM network for each category, and the blue curve represents the segmentation IoU of PolarNet for each category. PolarNet and our hybrid CNN-LSTM network have great differences in the segmentation results of objects in the outdoor environment. Our model is far better than PolarNet in some classes, such as motorcycle, truck, person, bicycle, terrain, pole. According to Fig. 5, motorcycle, truck, person, and bicyclist occupy a smaller number of point clouds in the dataset,



a.Hybrid CNN-LSTM Network with Hilbert Curve and Shallow Layer

b.Hybrid CNN-LSTM Network with Snake Curve

c.Hybrid CNN-LSTM Network with Hilbert Curve

Fig. 6. Three different neural network structures [34] in the ablation experiment.

TABLE II
ABLATION EXPERIMENT OF OUR NEURAL NETWORK

Model	Size	FPS	mIoU (%)	Per class IoU (%)																		
				car	bicycle	motorcycle	truck	bus	person	bicyclist	motorcyclist	road	parking	sidewalk	other-ground	building	fence	vegetation	trunk	terrain	pole	traffic-sign
Model of Fig.6a	[512, 512, 32]	12	51.4	91.1	22.9	31.0	44.9	30.0	**56.8**	61.7	0.0	89.7	13.4	74.0	0.1	87.3	44.6	87.1	59.7	74.9	63.2	43.3
Model of Fig.6b	[512, 512, 32]	**15**	54.2	**92.9**	26.0	44.1	**60.9**	28.4	49.7	73.6	0.0	84.7	22.4	70.3	0.0	88.8	50.8	**87.8**	**65.2**	**77.5**	61.7	**44.1**
Model of Fig.6c	[512, 512, 32]	11	**56.9**	92.6	**45.7**	**49.6**	48.6	**30.2**	53.8	**74.6**	**9.2**	**90.7**	**23.3**	**75.7**	**17.6**	**90.0**	**51.3**	87.1	60.8	75.4	**63.9**	41.5

which means that our model has a better segmentation effect for classes with few samples and smaller objects. However, our model still maintains good segmentation accuracy in other classifications, such as cars, buses, roads, sidewalks, buildings, vegetation and so on.

Although LSTM network is more suitable for learning time series, the results of this experiment prove that LSTM network also has a better effect in LiDAR point clouds segmentation. Although the CNN network has good performance in extracting spatial features, it will lose some associated regional features during the downsampling and maximum pooling process. Therefore, we are using LSTM network to process pseudo images to improve this defect. At the same time, we input the segmented pseudo image into the LSTM network in the order of the Hilbert curve. Because of the continuity of the Hilbert curve, the connection between the regions is enhanced, and the network can better segment small objects.

D. Ablation Experiment

We have designed three hybrid neural network architectures based on CNN network and LSTM network. The simple schematic diagrams of the three networks are shown in Fig. 6, in which we learned from [34]. In Fig. 6(a), the pseudo image is divided into blocks and input into LSTM network according to the Hilbert curve because the features extracted by the LSTM network are down-sampled by U-net four times, some low-dimensional information will be lost. This causes the segmentation edge to be rough, so we put the features extracted by LSTM network in the position after upsampling once of the upsampling channel of the decoding network. In Fig. 6(b), LSTM network is used to extract the features of each patch of the pseudo image according to the Snake curve [23], and then the feature concatenated with U-net after four times of downsampling. Then these features are sent to the upsampling channel of the decoding network. In Fig. 6(c), the pseudo image is divided into blocks and input into LSTM network according to the Hilbert curve, and then concatenated together with the features of encoder network after four times of downsampling as a new feature, it is fed to the upsampling channel of the decoding network.

The results of the ablation experiment are shown in Table II. Based on the analysis of Fig. 6(a), although the model hybrid CNN-LSTM network with Hilbert Curve and Shallow Layer considers the shallow information in the neural network, it only makes some improvements on the class “person”. Our analysis shows that the reason for this result is that U-net [34] like structure itself has been able to combine the information of the original data of different depths. Therefore, the network structure in Fig. 6(a) does not improve the segmentation accuracy of the network model. As shown in Fig. 6(b), the semantic segmentation accuracy of the hybrid CNN-LSTM network with Snake Curve network model is significantly improved compared to hybrid CNN-LSTM network with Hilbert Curve and Shallow Layer. Especially in the category “truck”, IoU is as high as 60.92%, and the segmentation accuracy for this category is much higher than RandLA (40.1%) and PolarNet (22.9%). As shown in Fig. 6(c), the hybrid CNN-LSTM network framework proposed in this paper has better segmentation effects in most classes than other models in the ablation experiment. The mIoU of the network model proposed in this paper has better segmentation effects paper exceeds the latest methods.

V. Conclusion

We propose a new neural network framework for LiDAR semantic segmentation, which combines the advantages of CNN network and LSTM networks to improve the accuracy of neural network for smaller object. Compared with other methods, the new framework proposed in this paper is more suitable for LiDAR point clouds with uneven distribution. We use LSTM network to process the pseudo image obtained from PointNet, divide the pseudo image into patches, and input it into LSTM network in Hilbert curve sequence. The use of the Hilbert curve strengthens the correlation between patches. The features extracted by LSTM network are combined with the features sampled under the CNN network and then applied to the decoding network.

Our hybrid CNN-LSTM network is compared with other methods on the SemanticKITTI dataset. Experiments show that our method has a better segmentation effect for small objects with fewer points and has higher mIoU. However, the real-time performance of the algorithm is affected because we sort the patches according to the Hilbert curve, then input it to LSTM network, and finally restore it to the original sequence. In the future, we are mainly prepared to adjust the structure of our model to improve the running speed and the classes with poor segmentation performance.

References

[1] K. Yuan, Z. Guo, and Z. J. Wang, “RGGNet: Tolerance aware LiDAR-Camera online calibration with geometric deep learning and generative model,” *IEEE Robot. Automat. Lett.*, vol. 5, no. 4, pp. 6956–6963, Oct. 2020.

[2] I. D. Miller *et al.*, “Any way you look at it: Semantic crossview localization and mapping with LiDAR,” *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 2397–2404, Apr. 2021.

[3] T. Chang, S. Legowik, and M.N. Abrams, “Concealment and obstacle detection for autonomous driving,” in *Proc. Robot. Appl. Conf.*, 1999, pp. 28–30.

[4] B. Wu, X. Zhou, S. Zhao, X. Yue, and K. Keutzer, “SqueezeSegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a LiDAR point cloud,” in *Proc. Int. Conf. Robot. Automat.*, 2019, pp. 4376–4382.

[5] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, “PointNet: Deep learning on point sets for 3d classification and segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 77–85.

[6] J. V. Dueholm, M. S. Kristoffersen, R. K. Satzoda, T. B. Moeslund, and M. M. Trivedi, “Trajectories and Maneuvers of surrounding vehicles with panoramic camera arrays,” *IEEE Trans. Intell. Veh.*, vol. 1, no. 2, pp. 203–214, Jun. 2016.

[7] Reddy. Satish and Pal. Prabir, “Computing an unevenness field from 3D laser range data to obtain traversable region around a mobile robot,” *Robot. Auton. Syst.*, vol. 84, pp. 48–63, 2016.

[8] J. Yang, Z. Kang, and P. H. Akwensi, “A skeleton-based hierarchical method for detecting 3-D pole-like objects from mobile LiDAR point clouds,” *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 5, pp. 801–805, May 2019.

[9] X. Chen *et al.*, “Moving object segmentation in 3D LiDAR data: A learning-based approach exploiting sequential data,” *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 6529–6536, Oct. 2021.

[10] Y. Zhou and O. Tuzel, “VoxelNet: End-to- end learning for point cloud based 3D object detection,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4490–4499.

[11] L. Luo, S. -Y. Cao, B. Han, H. -L. Shen, and J. Li, “BVMatch: LiDAR-Based place recognition using bird’s-eye view images,” *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 6076–6083, Jul. 2021.

[12] B. Wu, A. Wan, X. Yue, and K. Keutzer, “SqueezeSeg: Convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D LiDAR point cloud,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 1887–1893.

[13] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.

[14] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, “Multi-view 3D object detection network for autonomous driving,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6526–6534.

[15] C. R. Qi, Li. Yi, H. Su, and L. J. Guibas, “PointNet++: Deep hierarchical feature learning on point sets in a metric space,” in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5099–5108.

[16] L. Wang and B. Goldluecke, “Sparse-PointNet: See further in autonomous vehicles,” *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 7049–7056, Oct. 2021.

[17] Y. Zhang *et al.*, “PolarNet: An improved grid representation for online LiDAR point clouds semantic segmentation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit .*, 2020, pp. 9598–9607.

[18] J. Behley *et al.*, “SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9296–9306.

[19] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? The KITTI vision benchmark suite,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.

[20] Z. Han *et al.*, “SeqViews2SeqLabels: Learning 3D global features via aggregating sequential views by RNN with attention,” *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 658–672, Feb. 2019.

[21] J. H. Bappy, C. Simons, L. Nataraj, B. S. Manjunath, and A. K. Roy-Chowdhury, “Hybrid LSTM and encoder–decoder architecture for detection of image forgeries,” *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3286–3300, Jul. 2019.

[22] E. Motovilova and S. Y. Huang, “Hilbert curve-based metasurface to enhance sensitivity of radio frequency coils for 7-T MRI,” *IEEE Trans. Microw. Theory Techn.*, vol. 67, no. 2, pp. 615–625, Feb. 2019.

[23] G. Schrack and L. Stocco, “Generation of spatial orders and space-filling curves,” *IEEE Trans. Image Process.*, vol. 24, no. 6, pp. 1791–1800, Jun. 2015.

[24] C. Böhm, M. Perdacher, and C. Plant, “A novel Hilbert curve for cache-locality preserving loops,” *IEEE Trans. Big Data*, vol. 7, no. 2, pp. 241–254, Jun. 2021.

[25] L. Yousefi and O. M. Ramahi, “Artificial magnetic materials using fractal Hilbert curves,” *IEEE Trans. Antennas Propag.*, vol. 58, no. 8, pp. 2614–2622, Aug. 2010.

[26] X. Shu, L. Zhang, Y. Sun, and J. Tang, “Host–Parasite: Graph LSTM-in-LSTM for group activity recognition,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 2, pp. 663–674, Feb. 2021.

[27] Q. Hu *et al.*, “RandLA-Net: Efficient semantic segmentation of large-scale point clouds,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11105–11114.

[28] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena Martinez, and J. Garcia-Rodriguez, “A review on deep learning techniques applied to semantic segmentation,” 2017, *arXiv:1704.06857*.

[29] A. Paszke *et al.*, “Automatic differentiation in pytorch,” in *Proc. NIPS Autodiff Workshop, Future Gradient-based Mach. Learn. Softw. Techn.*, 2017, pp. 1–6.

[30] M. Tatarchenko, J. Park, V. Koltun, and Q. Zhou, “Tangent convolutions for dense prediction in 3D,” in *Proc.IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3887–3896.

[31] R. A. Rosu, P. Schutt, J. Quenzel, and S. Behnke, “Latticenet: Fast point cloud segmentation using permutohedral lattices,” *Autonomous Robots*, vol. 46, no. 1, pp. 45–60, Jan. 2022.

[32] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, “Rangenet++: Fast and accurate lidar semantic segmentation,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 4213–4220.

[33] C. Xu *et al.*, “Squeezesegv3: Spatially-adaptive convolution for efficient point-cloud segmentation,” in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 1–19.

[34] Z. Tang, X. Peng, K. Li, and D. N. Metaxas, “Towards efficient U-Nets: A coupled and quantized approach,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2038–2050, Aug. 2020.