IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received October 12, 2021, accepted November 5, 2021, date of publication November 10, 2021,
date of current version November 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3127234

POAT-Net: Parallel Offset-Attention Assisted Transformer for 3D Object Detection for Autonomous Driving

JINYANG WANG[1], **(Member, IEEE), XIAO LIN**[2], **AND HONGYING YU**[3]
[1]Rhinoceros Intelligent Robots Technology Company Ltd., Shanghai 200000, China
[2]Department of Computer Science, Shanghai Normal University (SHNU), Shanghai 200234, China
[3]Department of Electrical and Control Engineering, North University of China (NUC), Taiyuan 030051, China

Corresponding author: Jinyang Wang (yangnuc@outlook.com)

This work was supported by the National Natural Science Foundation of China under Grant 61775139, Grant 61772164, and Grant 62072126.

ABSTRACT 3D object detection is playing a key role in the perception process of autonomous driving and industrial robots automation. Inherent characteristics of point cloud raise an enormous challenge to both spatial representation and association analysis. Unordered point cloud spatial data structure and density variations caused by gradually varying distances to LiDAR make accurate and robust 3D object detection even more difficult. In this paper, we present a novel transformer network POAT-Net for 3D point cloud object detection. Transformer is credited with the great success in Natural Language Processing (NLP) and exhibiting inspiring potentials in point cloud processing. Our method POAT-Net is inherently insensitive to element permutations within the unordered point cloud. The associations between local points contribute significantly to 3D object detection or other 3D tasks. Parallel offset-attention is leveraged to highlight and capture subtle associations between local points. To overcome the non-uniform density distribution of different objects, we exploit Normalized multi-resolution Grouping (NMRG) strategy to enhance the non-uniform density adaptive ability for POAT-Net. Quantitative experimental results on KITTI3D dataset demonstrate our method achieves the state-of-the-art performance.

INDEX TERMS 3D object detection, non-uniform density, parallel offset-attention, point cloud, transformer.

I. INTRODUCTION

Robust and accurate 3D object detection from point cloud is becoming an urgent in autonomous driving and industrial robots automation. For instance, in autonomous driving, walking pedestrians, cyclists, lanes, overtaking and lane-changing vehicles are all considered to be real-time 3D objection detection targets for the running self-driving vehicles [1]. By far, most of the environment perception data is collected from 3D LiDAR sensor mounted on autonomous driving vehicles. And the output of 3D LiDAR is unordered and spatial discrete points set. What's more, the density of point cloud varies according to the distance between 3D LiDAR and different target objects. In other words, 3D LiDAR data is a kind of non-uniform sampling sensor to the surroundings to some extent [2].

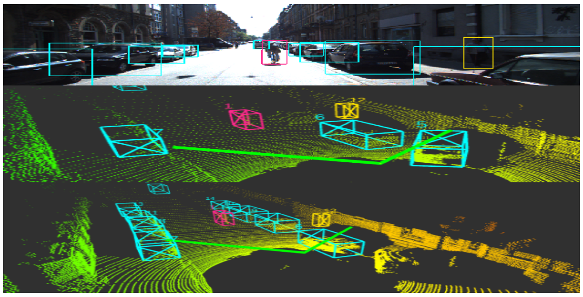


FIGURE 1. Comparative 3D object detection results of POAT-Net. Camera image and 2D bounding boxes at top row are provided for reference. Middle row is single offset-attention version without normalized multi-resolution grouping (NMRG), failing to detect cars with large occlusions on right and left side. Bottom row is the full version of POAT-Net.

Non-uniform density distribution of point clouds of different distances poses a challenging problem for 3D object detection [2].

The associate editor coordinating the review of this manuscript and approving it for publication was Charalambos Poullis.

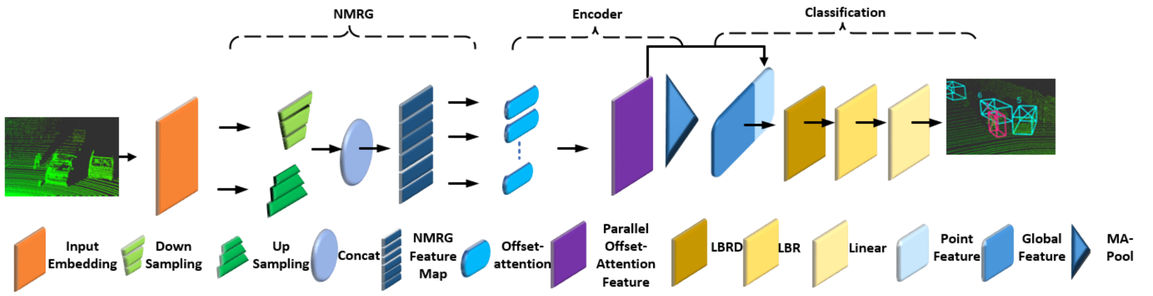


FIGURE 2. **Architecture of POAT-Net. The NMRG in this paper mainly consists of three groups of down-sampling and three groups of up-sampling modules. The encoder comprises six stacked offset-attention modules that extract varying scales of offset-attention features. Numbers above each module denote for number of output features. Max-Pool and Average-Pool are concatenated together to form MA-Pool. LBRD is for Linear, BatchNorm, ReLU and Dropout layer.**

Transformer vanilla assumes that all the elements within the point cloud set are uniformly sampled [3]. Therefore, feature extraction module of transformer vanilla performs poorly on non-uniform density point cloud, which we would explain and compare in Section IV. And this is also the reason that we do not use ModelNet40 or ShapeNet Datasets to verify our method as they are uniform density models without autonomous driving backgrounds. If extracting associations of local points at different sampling levels, we obtain more robust local information with density variations. We present a novel adaptive density structure named Normalized multi-resolution Grouping (NMRG), which enables our system to be robust to non-uniform density sampling.

Feature extraction using transformer is insensitive to permutations of elements within the point cloud. Transformer for Natural Language Processing (NLP) presents a novel representation for the input, which constructs the feature of each input by linear transformation of all the inputs [4]. Transformer adopts attention mechanism to calculate weight scores of every input. Point cloud data could be fed to transformer network directly even point cloud is not highly regularized data format. Inspired by [2] and [3], we do not transform point cloud data to a canonical data space, such as voxelizing the whole point cloud data into cubes or projecting it to multiple 2D images from different views. We prefer feeding point cloud data to the 3D coordinate based input embedding layer of POAT-Net to reduce feature dimensions and prepare for the association mining between points for later works. Through exploiting inherent characteristics of transformer, our system is more computational efficient since POAT-Net needs not to preprocess every point cloud frame. This is quite useful for practical autonomous driving perception applications with limited vehicular computing resources.

Parallel offset-attention module is derived from self-attention of transformer vanilla. Self attention is the core innovation of transformer, which extracts distinctive features from the local associations [5]. The input of self-attention is the sum of embedding and positional encoding of raw point cloud. The intermediate products of self-attention include three vectors q, k, v via linear operation for every component of input respectively. The attention weight matrix is calculated using dot-product between the key k and query q vectors of two arbitrary components. The final attention feature is produced by attention weighted sum of all value vectors v. Calculated using the linear transformation of all input information, the attention mechanism is able to capture the distinctive information. Parallel Offset-attention (POA) is an upgraded version of self-attention along with preceding module of NMRG. Parallel offset-attention takes the output multi-scale vectors of NMRG as input features and utilizes the subtraction between input and attention feature as the output of each scale level. The subtraction operation sharpens the attention features just as removing direct-current (DC) part in Control Theory. And concatenating different offset-attention features from varying scales enhances the robustness of POAT-Net.

We conduct quantitative experiments on KITTI3D benchmark to verify the correctness and efficiency of our system. The experimental results demonstrate that POAT-Net achieves state-of-the-art performance compared with existing mainstream approaches. We also visualize the feature capturing results of various 3D objects in ModelNet40 with parallel offset-attention to inspect the ability of POAT-Net to capture distinctive features using comparative query points. The main contributions of our work are listed as below:

- **Parallel Offset-Attention**. We propose this novel method to facilitate POAT-Net the ability of capturing the distinguishing features at different scale levels. There are two advantages for 3D object detection using parallel offset-attention. Firstly, relative 3D geometry positions between points are much more robust than absolute coordinates in real world especially when point cloud transforms with a tiny rigid translation or rotation. Secondly, parallel offset-attention benefits from the offset subtraction) operation, which is analogous to the Laplacian matrix proved to be effective by [18]. The Laplacian matrix in [18] is the difference between adjacency matrix and degree matrix. The attention map of POAT-Net could be regarded as adjacency matrix and respectively the degree matrix is equivalent to identity matrix if we normalize attention map to the sum of every row to one. Therefore, the parallel offset-attention can be understood as Laplacian process. In section IV,

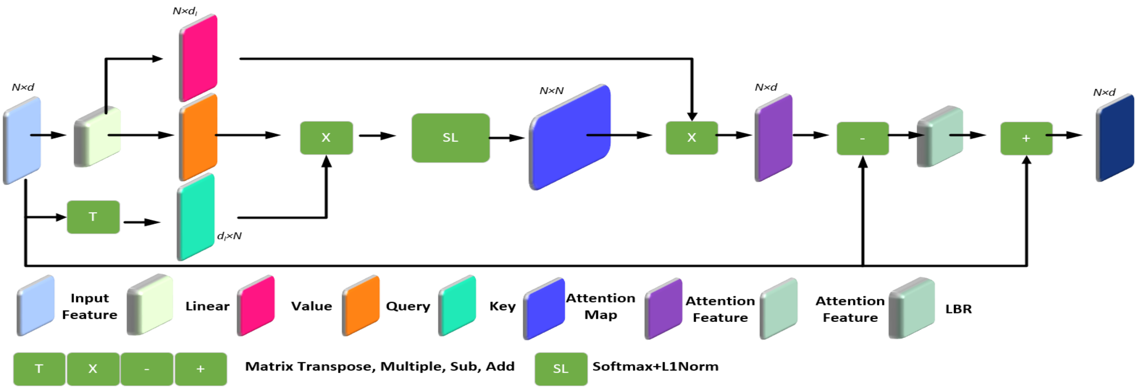


FIGURE 3. Architecture of Parallel Offset-Attention (POA). Query, Key and Value are three terminologies derived from Natural Language Processing (NLP). Query and Value are produced by linear transformation of output feature of NMRG module. L1 normalization is leveraged to sparse the multiplication result between Key *k* and Query *q*. Offset-attention feature is obtained by subtraction between original attention and NMRG feature. The final addition of offset-attention and NMRG is kind of encoder of low-level features of multi-resolution grouping and high-level distinctive features.

we perform quantitative and comparative experiments to verify the effectiveness of parallel offset-attention module.

- **Normalized multi-resolution Grouping (NMRG)**. NMRG aims at enabling our system to be adaptive to non-uniform density distribution of 3D object point cloud. NMRG constructs two pyramids of different scales regarding the scale coefficient of raw point cloud as dividing line. Thereafter, NMRG concatenates features from down-sampling pyramid and up-sampling pyramid together and transforms the information of the complete pyramid into normalized feature space, which is more suitable for subsequent detection works when scale problem occurs.
- **Invariant to initial states of point cloud**. Leveraging encoder and decoder structure of transformer, POAT-Net is insensitive to the permutations of point cloud fed to 3D coordinate based input embedding layer. Through incorporating T-net into the structure of POAT-Net, our system tolerates any initial rigid translation, rotation of the raw point cloud or the order of the points fed to POAT-Net.

The rest of our paper is organized as follows: Section II discusses the related works of 3D object detection from point cloud using transformer. Section III describes our method in detail, including parallel offset-attention, normalized multi-resolution grouping, etc. Section IV performs experiments to verify efficiency and robustness of POAT-Net on KITTI3D benchmark dataset and visualize the features learned in POAT-Net.

II. RELATED WORKS

A. TRANSFORMER FOR NLP

Transformer is achieving dominant position after Devlin *et al.* [6] proposes Bidirectional Encoder Representation from Transformer (BERT). Unlike previous language representation models, BERT pre-trains its bidirectional representations by jointly conditioning unlabeled text on left and right context in all layers. And BERT is convenient to be applied to language inference and question answering scenarios [7].

B. TRANSFORMER FOR 2D OBJECT DETECTION

Transformer has been applied to computer vision since its success in Natural Language Processing (NLP). Carion *et al.* [8] adopt encoder-decoder structure of transformer to reason about the relationships between the objects and the global image context and then output the final set of predictions in parallel. Ding *et al.* [9] proposes ROI (region of interest) transformer to overcome the highly complex backgrounds, variant appearance of objects problem. They feed spatial transformer with oriented bounding box annotations to learn transformer parameters. Srivinas *et al.* [10] incorporates global self-attention transformer with multiple computer vision tasks. They explain why ResNet bottleneck blocks with self-attention can be regarded as transformer blocks and the importance in image processing. Zhu *et al.* [11] proposes deformable transformer to put attention mainly on key sampling points around the reference annotations. What's more, deformable transformer achieves better performance than their previous DETR method [8].

C. TRANSFORMER FOR 3D DETECTION FROM POINT CLOUD

As the key idea of transformer, attention mechanism has been introduced into many deep learning frameworks for point cloud tasks. Zheng *et al.* [12] focuses on single-stage point cloud 3D object detection in an anchor-free manner. They fit the sparse feature maps to dense based on object regions through the deformable convolution tower and supervised mask-guided attention. Yuan *et al.* [13] uses the temporal-channel encoder of the transformer to encode the information of different channels and frames and the spatial decoder of the transformer to decode the information for each location. Yin *et al.* [14] adopts attentive Spatiotemporal Transformer GRU to aggregate the features, which is encoded by its feature encoder structure. However, We exploit Parallel Offset-Attention (POA) to capture the distinctive features from normalized multi-resolution grouping mechanism. Guo *et al.* [3] leverages farthest point sampling and nearest neighbor search to enhance input embedding and capture better local information of point cloud.

Inspired by [3], [15], [16], we present parallel offset-attention assisted transformer POAT-Net to overcome permutations issue of elements within point cloud and normalized multi-resolution grouping to mitigate non-uniform density distribution of point cloud.

III. SYSTEM DESCRIPTION

In this section, firstly, we clarify the problem discussed in this paper and describe the end-to-end framework of POAT-Net through the whole data flow from input-embedding to 3D object detection output layer in Fig. 2. Thereafter, we explain offset-attention mechanism in detail about how to better extract local features. And then normalized multi-resolution grouping (NMRG) is interpreted on how to overcome density distribution variations of different objects.

A. PROBLEM STATEMENT

In autonomous driving, real-time 3D object detection is an indispensable and fundamental function providing the perception information for the vehicle. The input of POAT-Net is raw point cloud from 3D LiDAR $PC_{raw} = \{p_1, p_2, \cdots, p_N\}$. Given consideration that we leveraging distance metric $D_{eu} \in \mathbb{R}^n$ inherited from Euclidean space, the metric space could be written as $\chi = \{PC_{raw}, D_{eu}\}$. The goal of our system is to optimize a set of 3D bounding boxes $B \in \{b_1, b_2, \cdots, b_{NI}\}$ for one frame of point cloud and assign a class vector $C \in \{c_1, c_2, \cdots, c_M\}$ to every 3D bounding box, where NI is the number of instances within the FOV (field of view) of LiDAR and M is the number of classes.

B. SYSTEM ARCHITECTURE

As shown in Fig. 2, the unified framework of POAT-Net consists of four parts, input embedding, NMRG, encoder and detection. Input embedding is fed with raw point cloud $PC_{raw} = \{p_1, p_2, \ldots, p_N\} \in \mathbb{R}^{3+d_e}$, where N is the total number of points within one frame of point cloud raw data. Therefore, d_e is the number of other properties, such as reflectivity, aligned RGB color or normal vectors and so on. Input embedding layer transforms the raw point cloud PC_{raw} into higher dimensional space $F_{in\_em} \in \mathbb{R}^{N*(d_{em})}$, where d_{em} is determined by the 3D coordinate based input embedding. Then normalized multi-resolution grouping (NMRG) and parallel offset-attention (POA) of the whole pipeline are explained in detail sequentially.

C. NORMALIZED MULTI-RESOLUTION GROUPING (NMRG)

Input embedding layer transforms raw point cloud into higher dimensional space $F_{in\_em} \in \mathbb{R}^{N*d_e}$ to obtain associations between discrete points based on combination of raw positions and input embedding. However, it does not take non-uniform density sampling into account. Therefore, we present NMRG to overcome this issue as its network structure is shown in Fig. 4. NMRG network structure consists of three modules, down-sampling, up-sampling and post-processing. Down-sampling and up-sampling process point cloud in parallel and produce density groups of different scales. Post-processing includes normalization and concatenation. Given consideration to efficiency and accuracy, the scale coefficient we adopt for later experiments between two adjacent levels is 2 and total number of levels is 5. We normalize features from different scale groups separately in case of the interaction between adjacent sampling levels. Six groups of normalized features are concatenated together to form an integral scaling association.

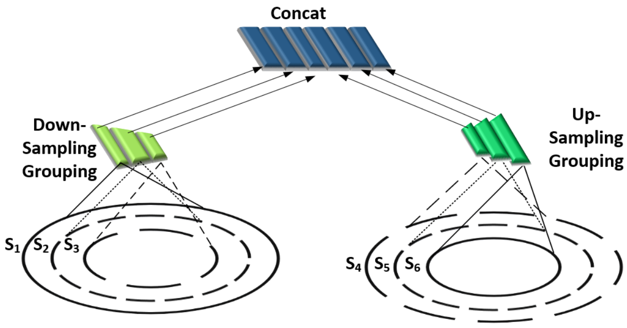


FIGURE 4. Schematic diagram of Normalized multi-resolution Grouping. We adopt scaling pyramids $S \in \{s_1, s_2, \cdots, s_6\}\}$ strategy for down-sampling and up-sampling modules based on experimental results. The output features of down-sampling and up-sampling are normalized and then concatenated together as NMRG feature for later processing.

As explained earlier, non-uniform density $D_{t,sid,x,y,z,r}$ of point cloud at different parts of one object or multiple objects challenge the accuracy and robustness of perception system. Features we extracted from sparse density areas could not be generalized to that from dense areas. Therefore, it is not reasonable to aggregate all the features directly from each point without consideration of non-uniform point cloud density. Models trained regardless of non-uniform density may not achieve satisfactory results both on sparse and dense part of the point cloud data obtained from vehicular 3D LiDAR sensors.

Theoretically, the denser the driving environment is sampled, the more abundant local information we obtain. However, sampling deficiency at low-density areas corrupts the local patterns compared with the same scale of the dense areas. In these circumstances, we concentrate on multi-resolution grouping to capture local features of different scales with density adaptive NMRG module (Fig. 4). NMRG extracts local features from different scaling levels and then concatenates them together. Therefore, POAT-Net is gathering environment information with gradually varying scales in essence rather than a single scale of various density distributions.

Structure of down-sampling and up-sampling modules are designed utilizing almost the same idea as shown in Fig. 4. Input embedding $F_{N,de}$ based on 3D coordinate is fed to down-sampling and up-sampling in parallel with $N \in \{m, n\}$, $d_{em} \in \{x, y, z, r\}$, where $\{x, y, z\}$ is the 3D coordinate and r is reflectivity of the point. In practical applications, $N \in \{m, n\} = \{len, 1\}$, where len is usually the total number of points within one data frame of 3D LiDAR. For down-sampling, each adjacent feature map size is calculated as $S_{dwn,n+1} = 0.5 \times S_{dwn,n}$. Similarly, each feature map in up-sampling size is $S_{up,n+1} = 2.0 \times S_n$ as the convolution kernel is twice the size larger than that of the previous level. However, this down-sampling or up-sampling is quite computationally expensive when high-accuracy 3D LiDAR is used. The number len of points within one data frame is quite large and its time complexity is $T(N) = O(N)$. And it is not reasonable to resize the raw point cloud input PC_{raw} to a smaller size directly as irreversible information loss occurred during this process. In terms of computation efficiency and detection accuracy, we selectively concatenate feature vectors together at some scales S_n to avoid expensive computation cost but still preserve the adaptive ability of non-uniform density distributions.

Thereafter, We encounter a problem that whether it is the best way to combine every output of down-sampling with that of up-sampling. Obviously and experimentally direct aggregation of their outcomes corrupts the robustness of detection and prolonged computation time. We propose a regularization method to accelerate the large computation cost brought by NMRG and guarantee the converge speed by using regularization in the subsequent selective concatenation. Each pair of features produced by down-sample F_{dwni} and up-sampling F_{up_i} within one combination $F_{comb} = Concat(F_{dwni}, F_{up_j}), i \neq j; i, j \in \{1, 2, \cdots, s-1\}$ at different scales should have almost the same Information Entropy metric IEN_{unif}. This is reasonable that the dense part and the sparse part of the point cloud complement each other to form approximate uniform density distribution. Therefore, the combination of NMRG is composed of down-sampling, up-sampling, normalization, combination and random drop layer. The regularization of selective combination choice strategy could be written as below:

$$Loss_{reg} = \|IEN_{unif} - IEN(F_{dwni}, F_{up_j})\|^2_{F_{comb}} \tag{1}$$

where IEN_{unif} denotes for the information entropy of one point cloud data frame with uniform density distribution. $D(F_{dwni}, F_{up_j})$ is the information entropy of down-sampling or up-sampling point cloud data frame concatenated with F_{dwni} and F_{up_j}.

D. PARALLEL OFFSET-ATTENTION

The Network Structure of parallel offset-attention is shown in Fig. 3. The input of parallel offset-attention is the output of NMRG. The output of each thread of parallel offset-attention is the difference between its input and features processed by self-attention. Instead of extracting global features only once, we propose parallel offset-attention to refine distinctive features with gradually changing scales in parallel.

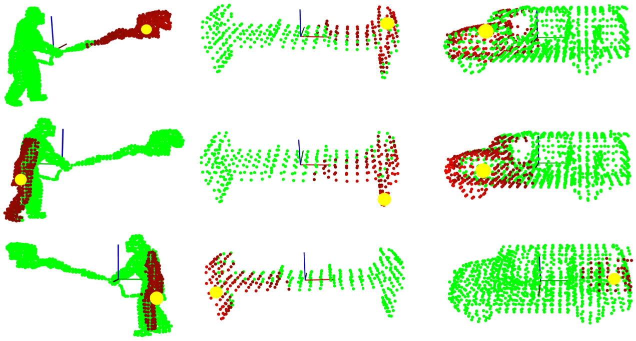


FIGURE 5. Visualization of attention map produced by parallel offset-attention module for person, bench and car with different spatial query points. Query point of each attention map is indicated with yellow circle (●).

Parallel offset-attention is designed to capture distinctive local features at different scales so as to generalize POAT-Net well when different 3D objects own almost the same point cloud representation. The amount of points representing the same 3D object varies significantly for the 3D LiDAR when the distance changes between autonomous driving vehicle and various 3D objects within the Field of View (FOV). Sometimes, the distinctive parts of different objects is not so easy to be obtained at one scale level. For example, on the street distinguishing a standing pedestrian and charging pile at a long distance is a great challenge as the different parts of the two objects are quite difficult to grab. Features of one level show poor generalization abilities on 3D objects at another scale. Therefore, we present parallel offset-attention mechanism to enhance the scale generalization ability of our system.

Offset-attention is a kind of variation of multi-head attention mechanism in transformer vanilla, which backbones POAT-Net without gradient vanishing and processing the data one by one [17]. The core idea of transformer attention is to learn global context vector U and capture the most important parts of the whole target. For example, we could implement soft attention mechanism via applying linear operation to three learnable weight matrix Q, K, V and the raw input point cloud PC_{raw}. Thereafter, we to obtain q, k, v weight coefficient vectors, which will be used in self-attention (SA) calculation. Parallel offset-attention is produced by element-wise subtraction between self-attention (SA) features and input features. The key idea of offset-attention is inspired by [Spectral networks and locally connected networks on graphs], which replacing adjacency matrix $\boldsymbol{E}$ with a Laplacian matrix $\boldsymbol{La} = \boldsymbol{D} - \boldsymbol{E}$. Where $\boldsymbol{D}$ is the diagonal degree matrix. Therefore, the process of offset-attention (OA) $F_{O\_A_{out}}$ could be formulated as below:

$$F_{O\_A_{out}} = O\_A(F_{in}) = LBR(F_{in} - F_{sa}) + F_{in} \tag{2}$$

$F_{in} - F_{sa}$ is an offset operator analogous to the [Spectral networks and locally connected networks on graphs]. The mathematical proof is as below:

$$F_{in} - F_{sa} = F_{in} - A_W \times V$$

$$
\begin{aligned}
&= F_{in} - A_W \times F_{in} \times W_v \\
&= F_{in} \times (I - A_W * W_v) \qquad (3)
\end{aligned}
$$

where A_W is attention weights coefficient matrix. V denotes for value matrix following the terminology of Natural Language Processing (NLP). Since the W_v is the linear layer's weight matrix, we can regard it as a unit matrix I. Therefore, $F_{in} - F_{sa}$ could be rewritten as below:

$$
\begin{aligned}
F_{in} - F_{sa} &= F_{in} \times (I - A_W * W_v) \\
&\approx F_{in} \times (I - A_W) \qquad (4)
\end{aligned}
$$

where I is unit identity matrix equivalent to the diagonal degree matrix D of Laplacian matrix calculation and A_W represents attention weight matrix corresponding to adjacency matrix E [18].

We enhance the attention weight $\widetilde{A}$ calculation for parallel offset-attention module by refined normalization in case of the mutual interference between features from different scales. $\widetilde{A}$ of transformer vanilla is calculated as below:

$$\widetilde{A} = Q \cdot K^T \qquad (5)$$

where V, Q and K are query matrix produced by linear operators of output features of NMRG F_{NMRG} and learnable $\{W_V, W_Q, W_K\}$ as:

$$
\begin{aligned}
\{V, Q, K\} &= F_{NMRG} \cdot (W_V, W_Q, W_K) \\
&= \{F_{NMRGV}, F_{NMRGQ}, F_{NMRGK}\} \\
\{Q, K\} &\in \mathbb{R}^{N \times d_a}, V \in \mathbb{R}^{N \times d_e} \\
\{W_Q, W_K\} &\in \mathbb{R}^{d_e \times d_a}, W_V \in \mathbb{R}^{d_e \times d_e} \qquad (6)
\end{aligned}
$$

Thereafter, after normalization $\widetilde{A}$ could be rewritten as:

$$\widetilde{A} = \frac{exp(\widetilde{\alpha}_{i,j})}{\sum_1^K exp(\widetilde{\alpha}_{k,j})} \qquad (7)$$

Compared with transformer vanilla only scaling the first dimension by $1/\sqrt{d_a}$ and normalizing the second dimension by softmax, we leverage softmax operator to process the first dimension and L1-norm to normalize the attention map of the second dimension. Via such an approach, POAT-Net mitigates the noise and make the attention weights more distinctive, which could be proved in Fig. 5 that the attention map weights vary significantly and are more semantically meaningful when the location of query point changes.

IV. EXPERIMENTS

We conduct experiments to verify the robustness and efficiency of POAT-Net on KITTI3D datasets. KITTI3D contains more than seven thousand training and test images and their corresponding point clouds, including eighty different objects. KITTI3D has more than one hundred thousand images and eighty thousand LiDAR point clouds for varying conditions and traffic densities in urban traffic scenes. We adopt single variable method to test the effectiveness of NMRG and parallel offset-attention on the datasets.

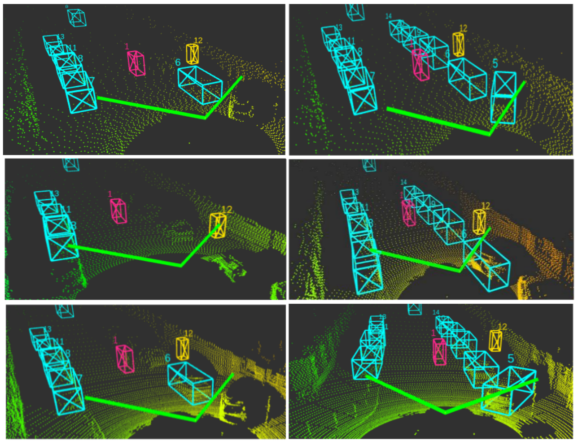


FIGURE 6. Experiments on NMRG. The left three rows are POAT-Net without NMRG module. The density from the top row to the bottom row is 12.5%, 25%, 50% of the original point cloud respectively. Right three rows are full POAT-Net with NMRG module.

TABLE 1. Experimental results of comparison with state-of-the-art methods on KITTI3D Car Test Dataset. Superscript 1 denotes for the method of one-stage and 2 is the method of two-stage. "Moda.," "Mod.," "M1," "M2" denotes for LiDAR, LiDAR+RGB respectively.

Method	Moda.	Easy	Mod.	Hard	mAP
PointRCNN^2 [20]	M1	86.96	75.64	70.70	77.77
3D IoU Loss^2 [21]	M1	86.16	76.50	71.39	78.02
UberATG-MMF^2 [22]	M2	88.40	77.43	70.22	78.68
STD^2 [23]	M1	87.95	79.71	75.09	80.92
CLOCs PVCas^2 [24]	M2	88.94	80.67	77.15	82.25
De-PV-RCNN^2 [25]	M1	88.25	**81.46**	76.96	82.22
PointPillars^1 [26]	M1	82.58	74.31	68.99	75.29
TANet^1 [27]	M1	84.39	75.94	68.82	76.38
Point-GNN^1 [28]	M1	88.33	79.47	72.29	80.03
SA-SSD^1 [29]	M1	88.75	79.79	74.16	80.90
CIA-SSD^1 [30]	M1	89.59	80.28	72.87	80.91
POAT-Net(ours)^1	**M1**	**90.39**	81.34	**76.99**	**82.91**

A. IMPLEMENTATION DETAILS

1) DATA PREPROCESSING

POAT-Net adopts mere one modality of environment information 3D LiDAR point cloud as input. POAT-Net does not voxelize or project objects within raw point cloud to multiple views. We truncate range of the three dimensions $\{x, y, z\}$ of raw point cloud into $[0, 60.5]$, $[-30, 30]$ and $[-2.5, 1]$ respectively, where the range unit is meter. Empirically, we set d_e as four with $\{x, y, z, r\}$ (see Part C of Section III). We leverage three kinds of data augmentation: (1) Rigid spatial transformation, which includes random translation between $[-2, +2]$ meters and rotation between $[-\pi/4, +\pi/4]$ radians on the whole point cloud frame; (2) Cross frame Addition, which randomly copies ground-truth objects within 3D bounding box of another point cloud frame into current data frame without interference with objects existing within the current frame; (3) Local jittering, which rotates and translates ground-truth 3D objects within the current point cloud frame between $[-0.5, +0.5]$ in meter and $[-\pi/4, +\pi/4]$ in radian.

2) DETAILS OF TRAINING

Cosine annealing strategy for learning rate and ADAM (Adaptive Moment Estimation) optimizer for gradient descent are used in POAT-Net. The initial learning rate, batch size and the number of epoches are 0.01, 32 and 1000. All the initial values of W_V, W_Q, W_K matrix are $1.0/len$, where len is 123000, the approximate average number of points within the original point cloud data frame of KITTI3D tested. The number of up-sampling and down-sampling is 3. POAT-Net runs six threads in parallel when it is started.

B. COMPARISON WITH STATES-OF-THE-ARTS ON KITTI3D

The experimental results are produced by submitting prediction data of POAT-Net to KITTI server and we compare them with the mainstream state-of-the-art methods in Table 1. Our method ranks 1st among all difficulty levels except moderate level. Our POAT-Net improves 1.4% compared with the best two-stage method CLOCs PVCas, which uses two modalities of RGB and 3D point cloud, and 0.8% compared with best one-stage method CIA-SSD on the easy level. POAT-Net improves 2.83% compared with the best one-stage method SA-SSD on the hard level and 2% of mAP compared with CIA-SSD.

C. ABLATION STUDY

Next we perform an ablation study of POAT-Net to investigate the contribution and effectiveness of each module we proposed using KITTI val split. Table 2 lists the ablation results of input embedding (IEM), NMRG (normalized multi-resolution grouping) and POA (parallel offset-attention). NMRG of POAT-Net overcomes the non-uniform density distribution problem via regularized normalization of multi-resolution grouping of point clouds. Objects in 3D Point clouds samples of KITTI3D are located from far hundreds of meters to near tens of centimeters. Therefore, the variations of distances of 3D objects to LiDAR provide us real non-uniform density distribution conditions to verify the effectiveness of POAT-Net. We replace input embedding based on 3D coordinates with the voxelizing method since we cannot remove it directly. The data we reported are produced with 40 recall points.

1) EFFECTIVENESS OF NMRG

As the third and first rows show in Table 2, NMRG we proposed improves the hard AP (Average Precision) by about 0.8 percentages. This significant improvement indicates that the proper combination of different scaling features complements the weakness brought by non-uniform density distribution. In contrast, we observe a tiny increase in easy AP. The reason in our view is that the point cloud density distribution is nearly uniform. Therefore NMRG strategy contributes less to the easy AP improvement. We also perform comparative experiments to check the effects with NMRG and without NMRG as shown in Fig. 6 with different densities.

TABLE 2. Ablation study of NMRG, Parallel offset-attention (POA) and Input Embedding (IE) modules we designed.

IEM	NMRG	POA	Easy	Moderate	Hard
-	-	-	80.26	72.20	66.82
-	-	✓	88.71	81.22	74.55
-	✓	-	88.71	81.22	74.55
✓	-	-	88.71	81.22	74.55
✓	✓	-	88.71	81.22	74.55
✓	✓	✓	90.39	81.34	76.99

TABLE 3. Experimental results of study effectiveness of POA on 3D bounding box regression and classification. The data are reported based on moderate difficulty level.

POA	3D IOU	Recall Rate	mAP	Precision
-	61.2%	81.96%	81.22%	63.92%
✓	66.8%	83.71%	81.34%	65.88%

2) EFFECTIVENESS OF POA

As the first and second rows show in Table 2, POA boosts AP rates of the moderate level and the hard level by about 0.6 and 0.9 respectively. The AP increase on the hard level is larger than that of NMRG, thus indicating the effectiveness of POA with more occlusions. The information of the occluded cars or pedestrians of the hard level is not complete as that in the easy level. This phenomenon shows that POA owns a measure of shape prediction ability for the occluded point cloud but it is limited and not perfect.

What's more, we further study the effectiveness of POA for 3D bounding box regression and classification confidence. As shown in Table 3, the improvement from the confidence module is larger than that from 3D bounding box regression module. We argue that POA may assist confidence optimization with mitigation the misalignment between classification confidence and localization accuracy.

3) EFFECTIVENESS OF IEM

In Table 2, the fourth and first rows indicate that IEM (Input Embedding) brings about 0.3 points improvement for the hard level and about 0.2 points for the moderate level. The AP increase in hard and moderate levels shows that IEM refines the associations between limited existing points and the global distinctive information. We think this is because relative 3D geometry positions the IEM leveraging between points are more robust than voxelizing method.

D. RUNTIME ANALYSIS

The inference time of POAT-Net is about 29ms on average, including 3.01ms for input preprocessing, 5.83ms for NMRG, 7.45ms for POA, 12.71ms for 3D bounding box regression and 4.03ms for classification. All the experiments are done on Intel Xeon Gold CPU and 4 TITAN GPU.

V. CONCLUSION

In this paper, we propose a 3D point cloud objects detection framework for Autonomous Driving. POAT-Net is insensitive to the permutations of point cloud, leveraging the inherent characteristics of transformer. POAT-Net mitigates the impact of non-uniform density caused by 3D sampling sensor. By combining normalized multi-resolution Grouping (NMRG) and Parallel offset-attention (POA), POAT-Net successfully improves the detection rate of occluded objects. NMRG mechanism normalizes and concatenates feature groups of down-sampling and up-sampling to assist POA with capturing local associations at different scales. Overall, the experiment results verify the effectiveness and robustness of our POAT-Net.

ACKNOWLEDGMENT

The authors would like to thank Rhinoceros Robotics for providing the Autonomous Mobile Robot (AMR) for the related experiments.

REFERENCES

[1] Y. Zhang, J. Wang, X. Wang, and J. M. Dolan, "Road-segmentation-based curb detection method for self-driving via a 3D-LiDAR sensor," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 12, pp. 3981–3991, Dec. 2018, doi: 10.1109/TITS.2018.2789462.

[2] L. Qi, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5099–5108.

[3] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu, "PCT: Point cloud transformer," *Comput. Vis. Media*, vol. 7, no. 2, pp. 187–199, Apr. 2021.

[4] W. Yuan, D. Held, C. Mertz, and M. Hebert, "Iterative transformer network for 3D point cloud," 2018, *arXiv:1811.11209*.

[5] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," 2018, *arXiv:1803.02155*.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[7] W. Liu, P. Zhou, Z. Zhao, Z. Wang, Q. Ju, H. Deng, and P. Wang, "K-bert: Enabling language representation with knowledge graph," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, Apr. 2020, pp. 2901–2908.

[8] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Aug. 2020, pp. 213–229.

[9] J. Ding, N. Xue, Y. Long, G.-S. Xia, and Q. Lu, "Learning RoI transformer for oriented object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2849–2858.

[10] A. Srinivas, T.-Y. Lin, N. Parmar, J. Shlens, P. Abbeel, and A. Vaswani, "Bottleneck transformers for visual recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16519–16529.

[11] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," 2020, *arXiv:2010.04159*.

[12] M. Zheng, P. Gao, R. Zhang, K. Li, X. Wang, H. Li, and H. Dong, "End-to-end object detection with adaptive clustering transformer," 2020, *arXiv:2011.09315*.

[13] Z. Yuan, X. Song, L. Bai, Z. Wang, and W. Ouyang, "Temporal-channel transformer for 3D LiDAR-based video object detection for autonomous driving," *IEEE Trans. Circuits Syst. Video Technol.*, early access, May 21, 2021, doi: 10.1109/TCSVT.2021.3082763.

[14] J. Yin, J. Shen, C. Guan, D. Zhou, and R. Yang, "LiDAR-based online 3D video object detection with graph-based message passing and spatiotemporal transformer attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11495–11504.

[15] J. Komorowski, "MinkLoc3D: Point cloud based large-scale place recognition," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 6098–6107.

[16] C. Kaul, N. Pears, and S. Manandhar, "FatNet: A feature-attentive network for 3D point cloud processing," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 7211–7218.

[17] E. Voita, D. Talbot, F. Moiseev, R. Sennrich, and I. Titov, "Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned," 2019, *arXiv:1905.09418*.

[18] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–14.

[19] S. Shi, X. Wang, and H. Li, "PointRCNN: 3D object proposal generation and detection from point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 770–779.

[20] D. Zhou, J. Fang, X. Song, C. Guan, J. Yin, Y. Dai, and R. Yang, "IoU loss for 2D/3D object detection," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2019, pp. 85–94.

[21] M. Liang, B. Yang, Y. Chen, R. Hu, and R. Urtasun, "Multi-task multi-sensor fusion for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7345–7353.

[22] Z. Yang, Y. Sun, S. Liu, X. Shen, and J. Jia, "STD: Sparse-to-dense 3D object detector for point cloud," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1951–1960.

[23] S. Pang, D. Morris, and H. Radha, "CLOCs: Camera-LiDAR object candidates fusion for 3D object detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 10386–10393.

[24] P. Bhattacharyya and K. Czarnecki, "Deformable PV-RCNN: Improving 3D object detection with learned deformations," 2020, *arXiv:2008.08766*.

[25] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12697–12705.

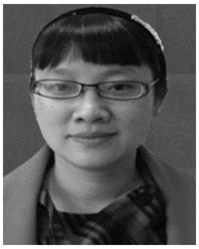
[26] Z. Liu, X. Zhao, T. Huang, R. Hu, Y. Zhou, and X. Bai, "TaNet: Robust 3D object detection from point clouds with triple attention," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, Apr. 2020, pp. 11677–11684.

[27] W. Shi and R. Rajkumar, "Point-GNN: Graph neural network for 3D object detection in a point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1711–1719.

[28] C. He, H. Zeng, J. Huang, X.-S. Hua, and L. Zhang, "Structure aware single-stage 3D object detection from point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11873–11882.

[29] W. Zheng, W. Tang, S. Chen, L. Jiang, and C.-W. Fu, "CIA-SSD: Confident IoU-aware single-stage object detector from point cloud," 2020, *arXiv:2012.03015*.

JINYANG WANG (Member, IEEE) received the B.S. degree in process equipment and control engineering and the M.S. degree in control engineering from the North University of China, Taiyuan, China, in 2010 and 2015, respectively. He is currently the Research and Development Director of Rhinoceros Robotics, Shanghai, China.



XIAO LIN received the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China. She is currently a Full Professor at the Department of Computer Science, Shanghai Normal University (SHNU), Shanghai. She has authorized and co-authorized a set of research papers in international journals and conferences, such as IEEE Transactions on Multimedia, *CAD* (Elsevier), and IEEE ICME. Her main research interests include image processing, computer vision, and machine learning.

HONGYING YU received the Ph.D. degree in artillery automatic weapon and ammunition engineering from the North University of China (NUC), Taiyuan, China. She is currently a Full Professor at the Department of Electrical and Control Engineering, NUC. She has authorized and co-authorized a set of research papers in internal journals and conferences. Her main research interests include industrial automation, robot vision, and machine learning.

• • •