ABSTRACT

Existing deep learning-based 3D object detectors typically rely on the appearance of individual objects and do not explicitly pay attention to the rich contextual information of the scene. In this work, we propose Contextualized Multi-Stage Refinement for 3D Object Detection (CMR3D) framework, which takes a 3D scene as an input and strives to explicitly integrate useful contextual information of the scene at multiple levels to predict a set of object bounding-boxes along with their corresponding semantic labels. To this end, we propose to utilize a context enhancement network that captures the contextual information at different levels of granularity followed by a multi-stage refinement module to progressively refine the box positions and class predictions. Extensive experiments on the large-scale ScanNetV2 benchmark reveals the benefits of our proposed method, leading to an absolute improvement of 2.0% over the baseline. In addition to 3D object detection, we investigate the effectiveness of our CMR3D framework for the problem of 3D object counting. Our source code is available at https://github.com/Dhanalaxmi17/CMR3D.

CCS CONCEPTS

- Computer systems organization → 3D Computer Vision;
  Object Detection; 3D Object Detection; 3D Object Counting; Robotics.

KEYWORDS

3D Object Detection, Context, Point Clouds, 3D Bounding Boxes, Refinement, Counting.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ACM Reference Format:

Dhanalaxmi Gaddam, Jean Lahoud, Fahad Shahbaz Khan, Rao Muhammad Anwer, and Hisham Cholakkal. 2022. CMR3D: Contextualized Multi-Stage Refinement for 3D Object Detection. In ACM Multimedia Asia (MMAsia '22), December 13–16, 2022, Tokyo, Japan. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3551626.3564938

1 INTRODUCTION

3D object detection from point clouds is an emerging computer vision technique that has a diverse set of applications, which include autonomous driving, robotic navigation, augmented reality, among many others. Object detection methods aim to simultaneously localize and classify 3D objects from a 3D point set. Compared to 2D object detection on 2D images, 3D object detection on point clouds is more challenging due to the increased dimensionality of the object search space and the order-less and sparse nature of point clouds.

Recently, several methods [21, 22, 29, 30] propose to detect 3D objects directly from point clouds. These methods generally utilize 3D point processing backbone networks [23, 25] to capture geometric information from raw point clouds followed by a Hough voting mechanism [22] to estimate object centers. Features encoded on these object centers are then used to predict the 3D bounding boxes. Among these voting-based approaches, H3DNet [39] achieves superior performance by introducing an intermediate representation comprising a hybrid and overcomplete set of geometric primitives. The geometric primitives used in H3DNet are bounding box centers, bounding box 2D face centers, and bounding box edge centers. Although H3DNet achieves promising results, it does not have an explicit mechanism to incorporate rich contextual information which is helpful for 3D object detection.

In this work, we introduce a context enhancement network that improves the baseline H3DNet by capturing contextual information at different granularity levels. Our context enhancement network consists of three modules: (1) geometric context module, (2) proposal context module, and (3) hybrid context module, for tightly integrating contextual information into the geometric primitive and proposal generation modules of the baseline. Furthermore, motivated by Cascade R-CNN [2] 2D detector, we introduce a multi-stage refinement strategy composed of a sequence of detectors which are
2 RELATED WORK

We propose a 3D object detection model that integrates the contextual information at different granularity levels and utilizes the multi-stage refinement module which refines the predicted bounding boxes and class predictions.

2.1 3D Object Detection

Given the advancement of deep learning methods on images and the abundance of 2D datasets, some 3D object detection methods choose to learn scene representation by projecting into a 2D plane. MV3D [9] and VoxelNet [40] use 3D data which is reduced to a 2D bird’s eye view and then fed to the network. These approaches depend on 2D detectors, which restricts the model from learning 3D shape information from point clouds. As a solution to this problem, many methods propose to directly process point clouds. 3D-SIS [12] and 3D-BoNet [32] use 3D instance segmentation methods to learn object bounding boxes. PointRCNN [25] presents a two-stage 3D object detector that generates the bounding box proposals first and then refines them to achieve the detection results. Motivated by the Hough Voting mechanism [17] in 2D object detection, VoteNet [22] proposes learning to vote towards object centers. It uses PointNet++ [23] as a backbone for feature extraction and vote estimation, and it then detects bounding boxes by feature sampling and grouping.

VoteNet is used as a building block for many methods. 3D-MPA [9] utilizes the voting mechanism and graph convolution networks for refining object proposals and uses 3D geometric features for aggregating the object detections. MLCVNet [30] improves VoteNet...
by using multi-level contextual information for object candidates. HGNet [4] improves the VoteNet and uses a hierarchical graph network with feature pyramids. H3DNet [39] incorporates the votes to additional 3D primitives such as centers of box edges and surfaces. BRNet [5] uses the back-tracing operation in Hough voting by querying neighboring points around the object candidates. Modeling the relationship between object proposals by graph structure is demonstrated in RGNet [10]. Given the challenging real world scenarios, we believe that knowledge of the complete scene would help in further improving the detection accuracy. This is achieved by understanding the relationship between the objects and their surroundings for better scene understanding.

2.2 Contextual Information
Contextual information is widely used in improving the performance of many tasks, which include 2D object detection [13, 18, 34], 3D point matching [8], point cloud semantic segmentation [33] and 3D scene understanding [38]. The point patch context is used in 3D point clouds for instance segmentation [15] and achieves enhanced performance. Hierarchical context priors are introduced in a recursive auto-encoder network [26] for detecting 3D objects. Self-attention is widely used in natural language processing [28] and learns global context. Inspired by this mechanism, contextual information is being used in many applications, such as image recognition [14], semantic segmentation [11] and point cloud recognition [31]. PCAN [37] proposes to utilize the attention network in 3D point data processing for capturing the contextual information. Instead of searching the whole input point cloud data, Attentional-PointNet [19] suggests searching in regions of interest while detecting 3D objects. MLCVNet [30] adopts a multi-level contextual information module for 3D object detection from point clouds. In this work, we utilize the concept of multi-level contextual information by integrating two self-attention modules and one multi-scale feature fusion module into a baseline method to learn multi-level contextual relationships between the geometric primitives, object centers, and the complete scene with their surroundings. Also, we use the multi-stage refinement strategy for refining the detected 3D bounding boxes and class predictions.

Counting: Counting objects is an interesting problem in the research field of computer vision as it helps in managing many scenarios such as traffic control, crowd management, and many more. Several methods [24, 36] propose to count objects from 2D images; however, to the best of our knowledge, there are few in the 3D domain especially on 3D point cloud datasets. Counting objects by detection is investigated in [3], where per-class object counting is done. Motivated by [6], we perform 3D object counting by detection.

3 APPROACH
The proposed 3D object detector, CMR3D, is shown in Fig. 2. It consists of three contextual modules that are integrated within our baseline [39], along with a multi-stage refinement module. The three contextual modules are the Geometric Context Module, the Proposal Context Module, and the Hybrid Context Module. We describe in detail these contextual modules in following sections.

3.1 Baseline Method
We choose H3DNet [39] as our baseline method. It consists of three main modules: geometric primitive module, proposal generation module, and classification and refinement module. It uses a PointNet++ backbone, which is a common approach adopted by numerous point-based methods, to learn features for a sub-sampled set of points. Features of downsampled points are used as input to a geometric primitive module that predicts the locations of the bounding boxes (BB) centers, faces, and edge centers. The centers are predicted using a Multi-Layer Perceptron (MLP), adapted from VoteNet’s Hough voting. These centers are passed through the proposal generation module, where a distance function is used for optimizing these centers and obtaining local minima which provide
Figure 3: Detailed architecture of the proposed 3D object detector, CMR3D. We introduce a context enhancement network to capture contextual information at three granularity levels and a multi-stage refinement module which progressively refines the box positions and class prediction. Our context enhancement network tightly integrates contextual information to the baseline through our geometric, proposal, and hybrid context modules. Geometric and proposal context modules use the self-attention layers followed by layer normalization, and the hybrid context module concatenates the proposals using multi-scale feature fusion layer. Point clouds are colored for illustration only.

high-fidelity object proposals. The final module takes these object proposals as input and classifies them as objects or not. This is achieved by combining the geometric features in the neighborhood of the object proposals with the help of MLP layers.

Limitations: In the baseline method, the geometric primitives are converted into object proposals by defining a distance function between an object and the geometric primitives [39]. However, we argue that the complete relationship between these primitives is not achieved and can be observed in Fig. 1. To this end, we propose CMR3D to obtain more information regarding the centers, proposals, and scenes at each level respectively.

3.2 CMR3D

To this end, we propose CMR3D, a 3D object detection network that utilizes the contextual information of the scene to better identify the 3D object bounding boxes. This is achieved with the help of three contextual models which are integrated into the baseline method: geometric context module (GCM), proposal context module (PCM), and hybrid context module (HCM). GCM helps the initial primitives (which are taken as input from the backbone) in capturing the information related to its neighboring primitives. This helps in increasing the voting accuracy for obtaining the accurate BB centers, BB face centers, and BB edge centers. PCM is used to get the relationship between the object proposals which are generated from each of the obtained BB centers. Further, HCM helps the network in understanding the relationship between the objects and their surroundings. Finally, the multi-stage refinement module progressively refines the box positions and class predictions. Next, we provide a detailed discussion about our novel modules.

3.2.1 Geometric Context Module (GCM). It is the first level of the context module introduced for retrieving the missing information within the initial primitives by collecting the necessary data from similar primitives, i.e., BB centers, BB face centers, and BB edge centers. Therefore, our GCM sub-module aims at better understanding the relationships between geometric primitives to improve the voting accuracy. The process is repeated with each primitive to enhance the voting accuracy by utilizing the concept of self-attention which can encode long-range dependencies. The squeeze and excitation method [14] shows that correlation between the channels promotes in retrieving the contextual information, especially in the object detection tasks. To this end, we use the concept of compact generalized non-local network (CGNL) [35] as the self-attention mechanism for exploiting the correlations between primitives and from the channels in feature space. After the extraction of points from PointNet++, we get the feature map \( F \). The feature map \( F \in \mathbb{R}^{n \times D} \) represents the hybrid geometric features of \( n \) points, which are subsampled from the input point cloud, and \( D \) is the feature vector dimension. The new feature map \( F' \) encodes the relationship between the geometric primitives, which can be formulated as:

\[
F' = LN \left( f(\theta(F), \phi(F))g(F) \right)
\]

Here, LN is the LayerNorm operation [1] which is applied along the channels of the feature map. \( \theta(F), \phi(F), \) and \( g(F) \) are three different transform functions and \( f(\theta(F), \phi(F)) \) encodes the similarities
between the two primitive positions. The new feature map \( F' \) is obtained after applying self-attention, and each primitive holds its own primitive features along with the information of other primitives. This whole process is repeated for all three centers i.e., BB center, BB face center, and BB edge center. The aqua-colored region inside the geometric primitive module in Fig. 3 shows the GCM module.

### 3.2.2 Proposal Context Module (PCM).

It is the second-level context module introduced for enhancing relationship between the object proposals. In the baseline method, the geometric primitives, BB centers, BB face centers, and BB edge centers are directly taken as input for obtaining the object proposals. We believe that this does not fully learn the relationships between objects. Instead, we propose to first pass the primitives obtained after voting into a self-attention module. By grouping the voted centers for each primitive, we attain a set of clusters for each geometric primitive: \( K_c, K_f, \) and \( K_e \) for BB center, BB face center, and BB edge center respectively. These clusters can be represented as \( K = \{ k_1, k_2, \ldots, k_N \} \) where \( N \) is the number of clusters generated by grouping the predicted centers. We provide the formulation for one \( K \), and the same applies for all three \( K_c, K_f, K_e \). To make \( k = \{ c_1, c_2, \ldots, c_n \} \) as a single vector form representation of cluster, it is fed into an MLP followed by the max-pooling operation. Here, \( c_i \) represents the \( i \)-th vote and \( n \) is the number of votes present in each \( k \). This is shown in the pink region of architecture, Fig. 3.

To consider the relationship between these cluster vectors, we pass the cluster features through a self-attention module. The cluster features \( K \in \mathbb{R}^{N \times D'} \) are fed to the attention module to generate a new feature map \( K_P \in \mathbb{R}^{N \times D'} \) which stores the relation between them, \( D' \) is the dimension of the feature map. This can be formulated as:

\[
k_p = \text{max-pool}(LN(SA(MLP(c_i))))
\]

where \( k_p \) is the new feature vector of new feature map \( K_P \), \( LN \) is the layer norm operation and \( SA() \) is a self-attention similar to equation 1.

### 3.2.3 Hybrid Context Module (HCM).

It is the third-level of context module which is introduced to capture the whole scene information to improve the detection accuracy. We believe that having access to information from the entire scene helps the network to easily detect an object and reduce ambiguity. Motivated from structure inference net [18], we propose the HCM module which enables the network to learn about contextual information and further improve the feature representation for BB proposals.

In order to achieve this, we create a new branch, where geometric primitives and clusters are taken (as shown in the blue region of Fig. 3) from GCM and PCM modules before they are processed through the self-attention layer. Primitives (\( G \)) and cluster features (\( K \)) are then pooled using the max-pool operation, and the resulting feature vectors are concatenated. Using an MLP, the obtained features are combined, and the output is expanded to match the size of the feature map of PCM module. This increases the BB detection performance as it considers the whole scene information. This operation can be formulated as:

\[
K_h = MLP([\text{max-pool}(k); \text{max-pool}(G)]) + K_P
\]

The geometric primitives are \( G \in G = \{ g_1, g_2, \ldots, g_P \} \) and object proposals (or clusters) are \( k \in K = \{ k_1, k_2, \ldots, k_N \} \) where \( P \) and \( N \) are sampled number of primitives and clusters respectively. \( K_P \) is the outcome of PCM, \( K_h \) is calculated for all three centers i.e, BB centers, BB face centers, and BB edge centers, and are used as input to our last module i.e., multi-stage classification and refinement.

### 3.2.4 Multi-stage Refinement Module.

The final module of our network is the multi-stage refinement module which takes object proposals (with rich contextual information) as input and outputs 3D object bounding boxes. This module has both classification and refinement sub-modules. The classification sub-module helps in determining whether the object proposals obtained represent an object or not, and the refinement sub-module determines the offset of detected objects. All the features associated with primitives are combined for each object proposal, as these contain the information which helps in determining the 3D bounding boxes. The baseline method has a single stage for refinement and classification of the predicted BB, whereas we believe that having the multi-stage refinement module enhances the efficiency of the network. Inspired by 2D cascade detection [2], where different threshold values are used to detect objects at multiple level. We intend to apply the same mechanism in 3D, where we use three levels of refinement with different threshold values \( U = \{ 0.5, 0.55, 0.6 \} \) at each level.

### 3.2.5 Extending CMR3D to Object Counting.

To the best of our knowledge, this is the first work that proposes counting of objects in 3D object detection network, especially on ScanNetV2 [7] dataset. We perform counting of objects by detection, and our main idea is to show that contextual information is helping the detector to better identify object occurrences. For counting the objects by detection, we compare our baseline method (H3DNet), Context-H3D, and CMR3D. In order to get the proper count of 3D bounding boxes, we use the objectness score as confidence level for each category. We choose a threshold confidence score of 0.95 for all three detectors. After getting the count of objects for each scan in the validation set of ScanNetV2, we use the evaluation criteria similar to the one used in [6].

### 4 IMPLEMENTATION DETAILS.

Our 3D object detector is implemented in PyTorch [20], and it is trained for 360 epochs using an Adam optimizer [16] with batch size 6. The base learning rate is initialized with \( 1e^{-2} \) and decay...
steps to [80, 140, 200, 240] with decay rates {0.1, 0.1, 0.1, 0.1}. We trained the model with one Quadro RTX 6000 GPU, it takes around 22 hours for training. We noticed that while training, the mAP oscillates with a small frequency; therefore, we report the max and mean results of five runs.

**Dataset** We use the ScanNetV2 [7] 3D dataset, which contains 1,513 scans of indoor scenes and annotations for 40 object categories, of which we consider 18 categories. Similar to VoteNet [22], we employ 1,201 scans for training and 312 validation scans as test set and sub-sample 40,000 points from every scan.

### 4.1 Evaluation Protocol

Similar to the other 3D object detectors, we use average precision (AP) and the mean average precision (mAP) among all semantic classes [27] for various IoU (Intersection over union) values. IoU is calculated as the ratio of the area of intersection to the area of union of the predicted bounding box and the ground truth bounding box. AP represents the average precision value across recall values ranging from 0 to 1.

Evaluation for counting is done by using the root mean square error (RMSE) along with its few variants: non-zero RMSE (nz-RMSE), relative RMSE (rRMSE), and non-zero relative RMSE (nz-rRMSE) [6]. For a given ground-truth counts \( g_{ic} \) and predicted counts \( p_{ic} \), the RMSE is calculated as,

\[
RMSE_{c} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (p_{ic} - g_{ic})^2}
\]

and relative RMSE is calculated as

\[
rRMSE_{c} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \frac{(p_{ic} - g_{ic})^2}{g_{ic} + 1}}
\]

where \( N \) is the number of scans in validation set. The \( m-RMSE \) and \( m-rRMSE \) are obtained by calculating an average error across all categories. Similarly, we obtain the count for non-zero ground-truth instances known as \( m - nz - RMSE \) and \( m - nz - rRMSE \).

### 4.2 Results Analysis

Table 1 shows the results of our 3D object detector, CMR3D, along with other state-of-the-art methods. We partition the table with respect to the number of backbones used and highlight the rows of methods with decay rates.

| Method          | cab  | bed  | chair | sofa  | tabl | door | wnsdw | blschf | pic  | contr | desk | curtin | fridge | showwr | toilet | sink  | bath  | ofurn | mAP   |
|-----------------|------|------|-------|-------|------|------|-------|-------|------|-------|------|--------|--------|--------|--------|-------|-------|-------|-------|
| VoteNet         | 8.1  | 76.1 | 67.2  | 68.8  | 42.4 | 15.3 | 6.4   | 28.0  | 1.3  | 9.5   | 37.5 | 11.6   | 27.8  | 10.0   | 86.5   | 16.8  | 78.9  | 11.7  | 33.5  |
| MLCVNet         | 16.6 | 83.3 | 78.1  | 74.7  | 55.1 | 28.1 | 17.0  | 51.7  | 3.7  | 13.9  | 47.7 | 28.6   | 36.3  | 13.4   | 70.9   | 25.6  | 85.7  | 27.5  | 40.0  |
| H3DNet          | 20.5 | 79.7 | 80.1  | 79.6  | 56.2 | 29.0 | 21.3  | 45.5  | 4.2  | 33.5  | 50.6 | 37.3   | 41.4   | 37.0   | 89.1   | 35.1  | 90.2  | 35.4  | 48.1  |

Table 2: 3D object detection accuracy on ScanNetV2 validation dataset. We show category-wise average precision (AP) results and mean average precision (mAP) with 3D IoU threshold 0.25.

| Method       | m-RMSE | m-nz-RMSE | m-rRMSE | m-nz-rRMSE |
|--------------|--------|-----------|---------|------------|
| H3DNet       | 1.1432 | 1.8661    | 0.5102  | 0.7971     |
| Context-H3D  | 1.0803 | 1.7690    | 0.5743  | 0.7635     |
| CMR3D        | 0.9647 | 1.6035    | 0.5097  | 0.6850     |

Table 3: 3D object detection accuracy on ScanNetV2 validation dataset. We show category-wise average precision (AP) results and mean average precision (mAP) with 3D IoU threshold 0.5.

Table 4: Performance comparison for counting objects using the m-RMSE error values with the baseline method, our method without multistage refinement, and our proposed method. Lower values means better performance.

| Method       | m-RMSE | m-nz-RMSE | m-rRMSE | m-nz-rRMSE |
|--------------|--------|-----------|---------|------------|
| H3DNet       | 1.1432 | 1.8661    | 0.5102  | 0.7971     |
| Context-H3D  | 1.0803 | 1.7690    | 0.5743  | 0.7635     |
| CMR3D        | 0.9647 | 1.6035    | 0.5097  | 0.6850     |

Table 5: Effectiveness of contextual modules are shown when compared with baseline, highlighted the baseline results.

| GCM | PCM | HCM | mAP@0.25 | mAP@0.50 |
|-----|-----|-----|----------|----------|
| ✓   |     | ✓   | 67.5     | 49.1     |
| ✓   | ✓   | ✓   | 67.3     | 49.4     |
| ✓   | ✓   | ✓   | 67.4     | 49.5     |
| ✓   | ✓   | ✓   | 67.7     | 49.8     |

Table 6: Ablation study results on the validation set of ScanNetV2 dataset. We report the max and (mean) of the five runs. Single backbone network shows a significant improvement in both our models compared with baseline.
We conduct ablation experiments to understand the effectiveness of each category individually. Table 4 shows the quantitative results using the mean RMSE values along with it variants. It can be observed that our model performs better compared to the baseline method. We believe that the multi-stage refinement is helping the model to better identify and localize objects. It can be observed that our Context-H3D also performs better than the baseline, which shows the importance of the contextual information in enhancing the model performance. Our sole intention to perform the counting on objects is to show that our model can better identify objects compared to the baseline method, and it can also be seen that contextual information is enhancing the model performance. Also, the introduction of multi-stage refinement module helps in improving the accuracy, as can be observed with the low RMSE values.

4.3 Ablation Study

We conduct ablation experiments to understand the effectiveness of the contextual modules. In Table 5, we can observe that introducing each module shows a slight increment in mAP.25 and almost 1% increase in mAP.50. This shows that the context modules are helping the model to learn better feature representations. Table 6 shows that our CMR3D performs better with single backbone, as it provides a significant amount of about 3% in mAP.25 and 5% in mAP.50. The average mAP of five runs follows the same trend indicating that contextualized multi-stage refinement is helping the model better understand the scenes. We provide few qualitative results in the supplementary material which shows that our CMR3D is better in localizing BB and predicting their classes.

5 CONCLUSION

We propose a contextualized multi-stage refinement for 3D object detection, where we capture the contextual information at different granularity levels. We utilize self-attention and multi-levels of refinement to capture the relationship between the objects and their surroundings and refine the location of object. We also show the benefits of our proposed method for object counting. For future work, variations of the self-attention modules can be employed to retrieve contextual information.

REFERENCES

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. arXiv preprint arXiv:1607.06450 (2016).

[2] Zhaowei Cai and Nuno Vasconcelos. 2018. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition. 6154–6162.

[3] Prithvijit Chattopadhyay, Ramakrishna Vedantam, Ramprasaath R Selvaraju, Dhruv Batra, and Devi Parikh. 2017. Counting everyday objects in everyday scenes. In Proceedings of the IEEE conference on computer vision and pattern recognition. 1135–1144.

[4] Jintai Chen, Biwen Lei, Qingyu Song, Haochao Ying, Danny Z Chen, and Yi-Feng Dai. 2018. Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition. 2022–2030.

[5] Bowen Cheng, Lu Sheng, Shaochun Shi, Ming Yang, and Zhiwei Fang. 2019. Voting feature learning on point sets in a metric space. Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems 31 (2018).

[6] Charles R Qi, Zhirong Huang, Hao Su, and Leonidas J Guibas. 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 977–986.

[7] Jun Liang, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Mu Li, and Yuhang Li. 2019. Dual attention network for scene segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 20–28.
In Proceedings of the IEEE/CVF International Conference on Computer Vision. 3712–3721.

[30] Qian Xie, Yu-Kun Lai, Jing Wu, Zhoutao Wang, Yiming Zhang, Kai Xu, and Jun Wang. 2020. Mlcvnet: Multi-level context votenet for 3d object detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 10447–10456.

[31] Saining Xie, Sainan Liu, Zeyu Chen, and Zhuowen Tu. 2018. Attentional shapecontextnet for point cloud recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 4606–4615.

[32] Bo Yang, Jianan Wang, Ronald Clark, Qingsong Hu, Sen Wang, Andrew Markham, and Niki Trigoni. 2019. Learning object bounding boxes for 3D instance segmentation on point clouds. Advances in neural information processing systems 32 (2019).

[33] Xiaoqing Ye, Jiamao Li, Hexiao Huang, Liang Du, and Xiaolin Zhang. 2018. 3D recurrent neural networks with context fusion for point cloud semantic segmentation. In Proceedings of the European conference on computer vision (ECCV). 403–417.

[34] Ruichi Yu, Xi Chen, Vlad I Morariu, and Larry S Davis. 2016. The role of context selection in object detection. arXiv preprint arXiv:1609.02948 (2016).

[35] Kaiyu Yue, Ming Sun, Yuchen Yuan, Feng Zhou, Errui Ding, and Fuxin Xu. 2018. Compact generalized non-local network. Advances in neural information processing systems 31 (2018).

[36] Cong Zhang, Hongsheng Li, Xiaogang Wang, and Xiaokang Yang. 2015. Cross-scene crowd counting via deep convolutional neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition. 833–841.

[37] Wenzhao Zhang and Chuanxu Xiao. 2019. PCAN: 3D attention map learning using contextual information for point cloud based retrieval. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 12436–12445.

[38] Yinda Zhang, Mingru Bai, Pushmeet Kohli, Shahram Izadi, and Jianxiong Xiao. 2017. Deepcontext: Context-encoding neural pathways for 3D holistic scene understanding. In Proceedings of the IEEE international conference on computer vision. 1192–1201.

[39] Zaiwei Zhang, Bo Sun, Haitao Yang, and Qixing Huang. 2020. H3dnet: 3D object detection using hybrid geometric primitives. In European Conference on Computer Vision. Springer, 311–329.

[40] Yin Zhou and Oncel Tuzel. 2018. Voxelnet: End-to-end learning for point cloud based 3D object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition. 4490–4499.