SCIENCE CHINA
Information Sciences

December 2019, Vol. 62 220104:1–220104:3
https://doi.org/10.1007/s11432-019-2636-x

• LETTER •
Special Focus on Deep Learning for Computer Vision

ARPNET: attention region proposal network for 3D object detection

Yangyang YE[1†], Chi ZHANG[2†] & Xiaoli HAO[1*]

[1]*School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing* 100044*, China;*
[2]*Institute of Automation, Chinese Academy of Sciences, Beijing* 100190*, China*

Received 16 May 2019/Revised 22 July 2019/Accepted 1 August 2019/Published online 5 November 2019

Citation Ye Y Y, Zhang C, Hao X L. ARPNET: attention region proposal network for 3D object detection. Sci China Inf Sci, 2019, 62(12): 220104, https://doi.org/10.1007/s11432-019-2636-x

Dear editor,

Three-dimensional (3D) object detection is a fundamental computer vision issue and demonstrates promising potential in intelligent surveillance, robot grasping, and autonomous driving. However, 3D object detection remains a challenging problem because it must predict many object details in 3D space, i.e., depth, shape, and orientation. Current 3D object detectors involve one-stage and two-stage frameworks.

The two-stage framework first uses a region proposal network (RPN) that produces non-uniform region proposals and converts their region-wise features into an identical form. Then, the identical features are sent to another network to filter and refine the initial proposals. The one-stage framework adopts a hypothetical region proposal scheme that imposes hypothetical non-uniform proposals to certain activations of the network and thus removes the region of interest (ROI) pooling block.

One-stage detectors ignore the statistic shape priors of the objects involved. Obviously, shape-specific proposals would aggregate more discriminative features and thus enhance detector performance, which is virtually the same as the top-down attention scheme [1] discovered in the human vision field.

We propose an attention RPN to provide a statistic shape prior to the one-stage detector. The proposed attention RPN is an embeddable, end-to-end and learnable network which is motivated by the success of attention modules in the natural language processing field [2]. To eliminate the non-uniform sampling of light detection and ranging (LiDAR) data as much as possible, we propose a distance-based sampling technique. The results are submitted to the KITTI benchmark [3] and demonstrate that the proposed attention region proposal network (ARPNET) outperforms many state-of-the-art techniques.

ARPNET. ARPNET takes a raw LiDAR point cloud as input, utilizes a distance-based voxel generator and a voxel feature extractor to generate voxel features, and then encodes with sparse convolutional layers and a feature pyramid network. Finally an attention RPN (Figure 1) provides information for 3D object detection.

We follow the procedure described in [4] to generate voxel representation from LiDAR point clouds. Assume the LiDAR point cloud includes a 3D space with range H, W, D, which represents vertical height, horizontal position, and distance, respectively. Each voxel has a size $\Delta_H = 0.4$, $\Delta_W = 0.2$, $\Delta_D = 0.2$. Then, the size of the whole voxel grid is H/Δ_H, W/Δ_W, D/Δ_D. The Velodyne HDL-64E LiDAR scanner has an angular resolution (azimuth) of 0.08° and a vertical resolution of 0.4° between every two rays. This leads to highly variable point density throughout the 3D space. The distribution of LiDAR point cloud is shown in Appendix A. We propose a distance-based sampling to handle the non-uniform sam-

* Corresponding author (email: xlhao@bjtu.edu.cn)
† Ye Y Y and Zhang C have the same contribution to this work.

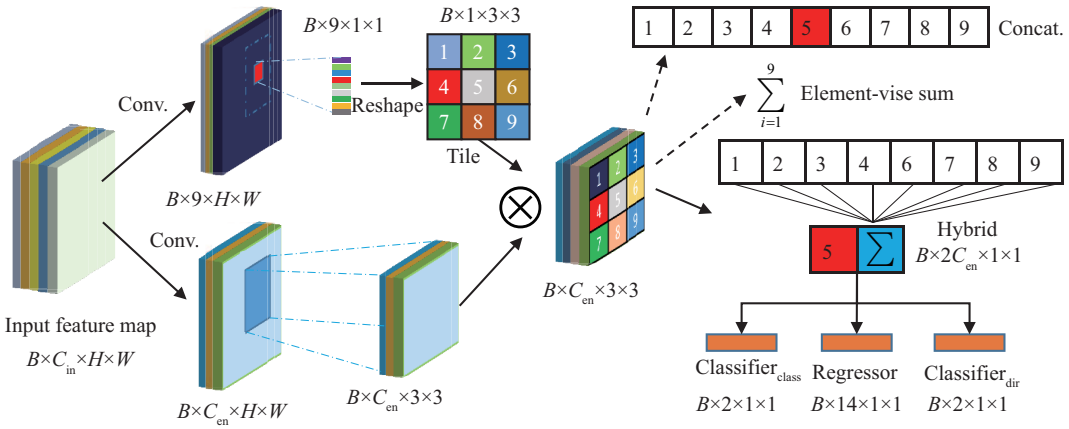

Figure 1 (Color online) Structure of attention RPN.

pling of LiDAR data. We define a sample point as a valid sample point if it satisfies all Dist $> M$. The Dist in 3D space can be formulated as a Euclidean distance. To further accelerate processing, we use an absolute value distance Eq. (1), where $*_{\text{val}}$ represents all existing valid sample points and x, y, z are points in the 3D space. Here, M is a numerical value, which indicates that we do not utilize two very close sample points in a single voxel. The representation of a voxel near the LiDAR sensor can be more stable and reasonable than the original random sampling.

$$\text{Dist} = |x - x_{\text{val}}| + |y - y_{\text{val}}| + |z - z_{\text{val}}|. \quad (1)$$

The feature extraction layers include a voxel feature extractor, sparse convolution layers, and a feature pyramid network (FPN). The voxel feature extractor includes voxel feature encoding layers and a fully connected network (FCN) [4] to generate voxel-wise features. The sparse convolutional layers include two sparse convolution phases to perform down-sampling along the z-axis. The first phase of sparse convolution involves a submanifold layer [5] and one sparse convolution layer. The second phase of sparse convolution includes two submanifold layers and a sparse convolution layer. The structure of the FPN is taken from [4,6]. The FPN includes down-sampling convolutional layers, convolutional layers (Conv2D), de-convolutional layers, and a concatenation layer. Each convolutional layer follows a BatchNorm layer and a ReLU layer. With the exception of the de-convolutional layer, all the convolutional layers use a 3×3 kernel. The kernel size of the de-convolutional layer depends on the expanding scale. When the scale is $\times$ 1, this is a Conv2D.

The attention RPN can generate shape-specific proposals that are ignored in existing one-stage detectors, and it can be embedded into nearly all one-stage detection networks to aggregate features with the learnable weights according to the statistic shape prior of objects. The naive attention mechanism of two-stage detectors is not effective when the RPN predicts a bounding box with plenty of background for an object or more than one object in a box. This will reduce performance due to both effective and ineffective features using the same weight in the ROI pooling layer. This problem can be improved by embedding an attention mechanism module that selects the important features by assigning different weights to each cell of a feature map according to the different significance.

Note that one-stage detectors suffer from some critical constraints, i.e., they cannot use the information of a predicted bounding box to obtain an ROI, and the output of one-stage detectors depends on the feature of the current anchor. We solve this problem as follows. Step 1: predefining a region. For example, we predefine a 3×3 kernel for each current anchor. The predicted information of a 3D object depends on the features of these regions. Here, we use $G = \{(-1,-1),(-1,0),\ldots,(0,1),(1,1)\}$ to represent the relative position of a current anchor. The relative location of the current anchor is $(0,0)$. Step 2: embedding attention mechanism. The proposed one-stage detector must predict nine attention weights to correspond to the predefined 3×3 region. The feature of the current anchor learns nine attention weights F_{att} using a convolutional layer and then reshapes F_{att} into 3×3. Simultaneously, the feature map of the local region utilizes an encoding convolutional layer to generate F_{en}. Then, for each anchor, we repeat the reshaped F_{att} and take an element-wise product of the repeated feature and F_{en}. Step 3: hybrid attention feature maps. The outputs of one-stage detectors only depend on only the feature of the current anchor (a size of 1×1); however, the size of obtained attention feature

maps for each anchor is 3×3. To handle this, an common choice is to concatenate the obtained feature maps for each anchor together (in a feature channel) or perform an element-wise summation. We address this issue by hybridizing concatenation and element-wise summation (Figure 1). The reasons for performing this hybridization process are summarized as follows. When there is a large predefined region (e.g., 5×5), the concatenation operation leads to a huge feature map and weakens the effect of the feature of the current anchor. Unlike two-stage detectors that provide a valid ROI, one-stage detectors must consider problems related to an invalid border. The element-wise summation operation can handle this problem to some extent, and we want to keep the current anchor's feature.

Taking a 3×3 predefined region as an example, the process of generating the local attention of each anchor can be formulated as

$$\begin{aligned} &F_{\text{att}} = \text{Conv}_{\text{att}}(F_{\text{in}}(0,0)),\\ &F_{\text{en}} = \text{Conv}_{\text{en}}(F_{\text{in}}),\\ &F_{\text{op}} = F_{\text{en}} \cdot \text{Repeat}(\text{Reshape}(F_{\text{att}})),\\ &F_{\text{hybrid}} = \left[F_{\text{op}}(0,0), \sum_{G(i)\neq(0,0)}^{N} F_{\text{op}}(G_{(i)}) \right]. \end{aligned} \tag{2}$$

Here, the F_{in} and F_{hybrid} mean the input and output of the local attention, respectively, Conv_{en} is the encoding convolutional layer with C_{en} channels, Conv_{att} is the attention convolutional layer with nine channels, G is the relative position of the current anchor, $\cdot$ is the element-wise product, Repeat is the operation to copy the feature map, $\sum$ is element-wise summation, and $[*]$ represents concatenation. As shown in Figure 1, $F_{\text{in}}(0,0)$ is the current anchor's feature, the size of F_{in} is $B \times C_{\text{in}} \times 3 \times 3$, and the size of F_{hybrid} is $B \times 2C_{\text{en}} \times 1 \times 1$.

We define the loss function as follows:

$$\text{Loss} = \alpha L_{\text{cls}} + \beta L_{\text{reg}} + \gamma L_{\text{dir}}. \tag{3}$$

Here, L_{cls} is the classification loss, L_{reg} is the regression loss, and L_{dir} is the direction classification loss. We set $\alpha = 1.0$, $\beta = 2.0$, and $\gamma = 0.2$ in our experiments. Each L_* is defined as follows.

Classification loss function. We apply focal loss to the proposed network's architecture.

Regression loss function. We set a 3D ground truth bounding box as x_g, y_g, z_g, l_g, w_g, h_g, θ_g, where x, y, x represent the center location, l, w, h represent the length, width, and height of the 3D bounding box, respectively, and θ is the yaw rotation around the z-axis. The positive anchor is parameterized as x_a, y_a, z_a, l_a, w_a, h_a, θ_a. Δx, Δy, Δz, Δl, Δw, Δh, and $\Delta\theta$ represent the corresponding residual. The residual is expressed as Eq. (4). Note that we use the SmoothL1 function to compute the regression loss.

$$\begin{aligned} &\Delta x = \frac{x_g - x_a}{d_a},\ \Delta y = \frac{y_g - y_a}{d_a},\ \Delta z = \frac{z_g - z_a}{h_a},\\ &\Delta l = \log\left(\frac{l_g}{l_a}\right),\ \Delta w = \log\left(\frac{w_g}{w_a}\right),\ \Delta h = \log\left(\frac{h_g}{h_a}\right),\\ &\Delta\theta = \theta_g - \theta_a. \end{aligned} \tag{4}$$

Here, $d_a = \sqrt{l_a^2 + w_a^2}$ is the diagonal of the base of the anchor.

Experiments. We evaluated the proposed ARPNET on the KITTI benchmark [3] for bird's eye view detection (BEV), 3D object detection (3D) and object orientation estimation (Ori.) which comprises 7481 training samples and 7518 testing samples, covering three classes for testing, i.e., Car, Pedestrian, and Cyclist. Compared to LiDAR-based methods [4,6], the proposed method achieves $\approx$ 8%–9% improvement in BEV for the Car and Cyclist classes, $\approx$ 5.3% improvement in 3D for the Cyclist class, and $\approx$ 7%–11% improvement in Ori. for the Car and Cyclist classes. The experimental details can be found in Appendixes B and C.

Acknowledgements This work was supported in part by National Key R&D Program of China (Grant No. 2018YFB1004600), Beijing Municipal Natural Science Foundation (Grant No. Z181100008918010), National Natural Science Foundation of China (Grant Nos. 61836014, 61761146004, 61602481, 61773375), Fundamental Research Funds of BJTU (Grant No. 2017JBZ002), and in part by Microsoft Collaborative Research Project.

Supporting information Appendixes A–C. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Baluch F, Itti L. Mechanisms of top-down attention. Trends Neurosci, 2011, 34: 210–224
2 Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: Proceedings of Advances in Neural Information Processing Systems, Long Beach, 2017. 5998–6008
3 Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The kitti vision benchmark suite. In: Proceedings of IEEE International Conference on Computer Vision, Providence, 2012. 3354–3361
4 Zhou Y, Tuzel O. Voxelnet: end-to-end learning for point cloud based 3D object detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 4490–4499
5 Graham B, van der Maaten L. Submanifold sparse convolutional networks. 2017. ArXiv:1706.01307
6 Yan Y, Mao Y X, Li B. Second: sparsely embedded convolutional detection. Sensors, 2018, 18: 3337