VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection

CVPR 2018

Summary

In this paper, VoxelNet is put forward as a novel end-to-end trainable architecture for 3D object detection. The network operates on a voxelized representation of the LiDAR point cloud for which voxel-wise features are computed using the proposed Voxel Feature Encoding layer (VFE). A variant of the popular Region Proposal Network (RPN) is used for oriented bounding box estimation. VoxelNet shows to outperform MV3D, the previous state-of-the-art, on all difficulty levels of the KITTI 3D and BEV Object Detection Benchmark by considerable margins.

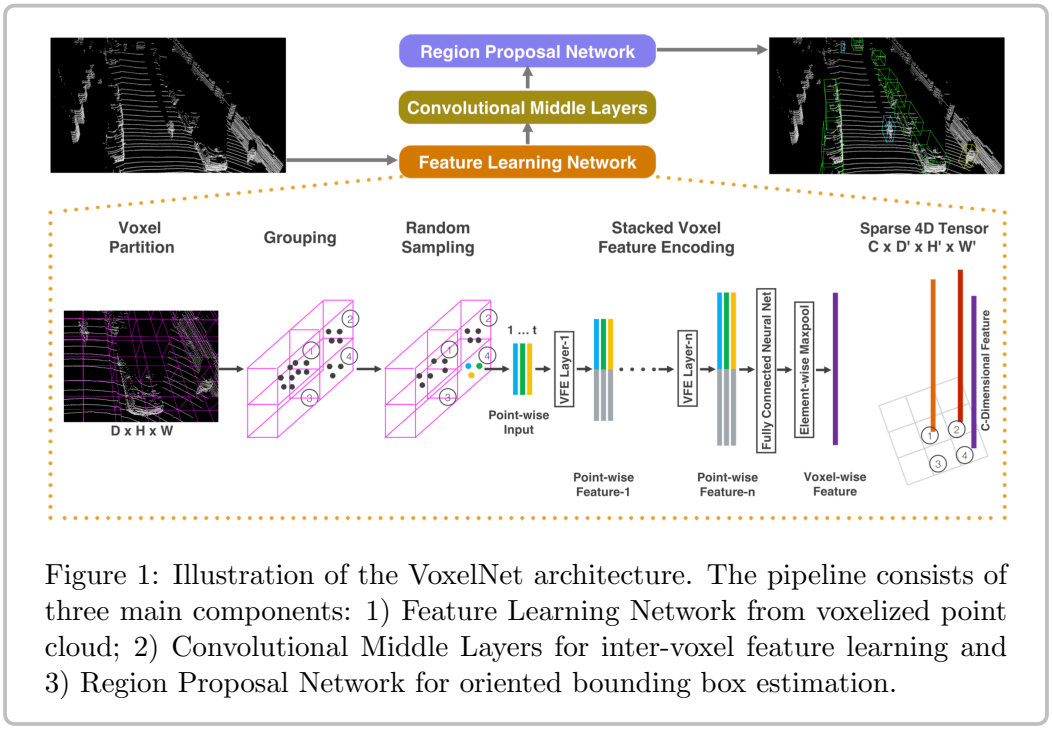

Figure 1: Illustration of the VoxelNet architecture. The pipeline consists of three main components: 1) Feature Learning Network from voxelized point cloud; 2) Convolutional Middle Layers for inter-voxel feature learning and 3) Region Proposal Network for oriented bounding box estimation.

Main Contributions

- **Novel end-to-end trainable 3D object detection architecture** VoxelNet is designed as an end-to-end trainable architecture that operates on the original point cloud directly without the need for any hand-crafted features or projections of the input data to e.g. BEV representation. This allows point clouds to be processed at full resolution, maintaining full information about 3D shapes as well as required algorithmic invariances.
- **Voxel Feature Encoding Layer (VFE)** The authors present the VFE layer as the main contribution of their work. The voxelization step results in a varying number of LiDAR points inside each voxel. The VFE layer computes point-wise features and aggregates them using max-pooling (see Figure 2). By stacking several FVE layers and passing the output to an MLP with final element-wise max-pooling, a voxel-wise feature representation is extracted that is used in the later stages of the pipeline.

Implementation Details

- **Convolutional Middle Layers** As the Feature Learning Network merely computes a per-voxel feature representation, inter-voxel dependencies are computed using a set of 3D convolutional layers. These convolutional operations down-sample the third dimension of the four dimensional input tensor such that the output of the convolutional middle layer is a 3D tensor of shape $C \times H \times W$ with C being the number of feature maps. Thus, the Region Proposal Network can resort to more computationally efficient 2D convolutions for the estimation of the oriented bounding boxes.

- **Region Proposal Network** The header network of the VoxelNet architecture can be seen as a modification of the original Region Proposal Network (RPN). Using a combination of convolutional and de-convolutional blocks, probability and regression output maps are computed in an encoder-decoder fashion at a down-sampling factor of two w.r.t to the size of the input tensor.

- **Efficient Implementation based on Sparse Point Structure** One of the main challenges of working with 3D point clouds is their inherent sparsity. The authors state that approximately 90% of the constructed voxel grid are typically empty. Consequenlty, voxel-wise feature extraction using VFE layers lends itself to an implementation based on sparse tensors. In order to save computational resources, a so-called *Voxel Input Feature Buffer* is constructed that assigns points to their corresponding voxel, disregarding empty voxels (see Figure 3). Based on this representation, a forward pass of the Feature Learning Network can be performed efficiently. Given the indices of the non-empty voxels, the sparse tensor can then be turned into a dense representation again, which is fed to the Convolutional Middle Layers for further processing.

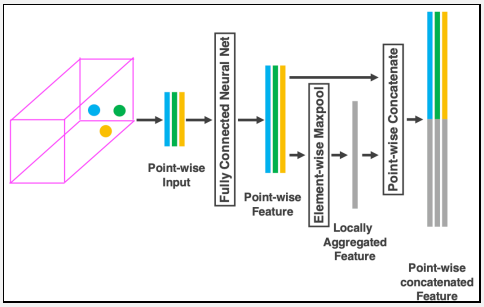

Figure 2: Illustration of the VFE layer for intra-voxel point-wise feature learning using MLP, max-pooling and concatenation.

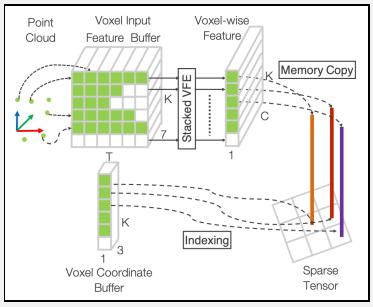

Figure 3: Illustration of the data representation based on sparse tensors for efficient feature learning.

Evaluation

- **KITTI 3D and BEV Object Detection** VoxelNet is compared to several previous arts with MV3D being the main reference. It shows to outperform both variants of MV3D, with and without image data, by large margins on all three difficulty levels of the 3D and BEV benchmark. Furthermore, promising results can be obtained for the considerably more difficult tasks of pedestrian and cyclist detection. However, no reference scores for other methods are reported.

References

This summary is solely based on my understanding of the original paper. All images used here are taken from the original paper as well. The paper can be found under the following link:
`https://arxiv.org/abs/1711.06396`