Point-Voxel CNN for Efficient 3D Deep Learning

Zhijian Liu*, Haotian Tang*, Yujun Lin, and Song Han

Project Page: http://pvcnn.mit.edu/
3D Deep Learning

3D Part Segmentation
(for Robotic Systems)

3D Semantic Segmentation
(for VR/AR Headsets)

3D Object Detection
(for Self-Driving Cars)

3D deep learning has been used in various applications on resource-constrained edge devices.
Efficient 3D Deep Learning

Off-chip DRAM access is much more expensive than arithmetic operation!

Random memory access is inefficient due to the potential bank conflicts!

Efficient 3D deep learning models should have small memory footprints and avoid random memory access.
Voxel-Based Models: Cubically-Growing Memory

Low resolutions lead to **significant information loss**.
High resolutions lead to **large memory consumption**.

3D ShapeNets [CVPR’15]
VoxNet [IROS’15]
3D U-Net [MICCAI’16]
Point-Based Models: Sparsity Overheads

Up to 80% of the time is wasted on structuring the sparse data, not on the actual feature extraction.
PVCNN combines the advantages of point-based models (small memory footprint) and voxel-based models (regularity).
Point-Voxel Convolution (PVConv)

Features from **Voxel-Based Branch:**

![Voxel-based features](image1)

Features from **Point-Based Branch:**

![Point-based features](image2)

Voxel-based branch captures **large, contiguous** parts. Point-based branch captures **isolated, discontinuous** details.
Results: 3D Part Segmentation (ShapeNet)

PVCNN outperforms PointCNN with 2.7x measured speedup and 1.5x memory reduction (on a GTX 1080Ti GPU).
Results: 3D Part Segmentation (ShapeNet)

0.25 PVCNN runs with **real-time performance** (20 FPS) on the lightweight edge device (Jetson Nano).
Results: 3D Semantic Segmentation (S3DIS)

PVCNN++ outperforms PointCNN with **6.9x** measured speedup and **5.7x** memory reduction (on a GTX 1080Ti GPU).
Results: 3D Semantic Segmentation (S3DIS)

0.25 PVCNN outperforms PointNet with 1.8x measured speedup and 1.4x memory reduction (on a GTX 1080Ti GPU).
## Results: 3D Object Detection (KITTI)

<table>
<thead>
<tr>
<th></th>
<th>Efficiency</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Cyclist</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Latency (GPU)</td>
<td>Memory (GPU)</td>
<td>Easy</td>
<td>Moderate</td>
</tr>
<tr>
<td><strong>F-PointNet++</strong></td>
<td>105.2 ms</td>
<td>2.0 GB</td>
<td>83.8</td>
<td>70.9</td>
</tr>
<tr>
<td><strong>PVCNN</strong> (efficient)</td>
<td>58.9 ms (1.8x)</td>
<td>1.4 GB (1.4x)</td>
<td>84.2 (+0.4)</td>
<td>71.1 (+0.2)</td>
</tr>
<tr>
<td><strong>PVCNN</strong> (complete)</td>
<td>69.6 ms (1.5x)</td>
<td>1.4 GB (1.4x)</td>
<td>84.0 (+0.2)</td>
<td>71.5 (+0.6)</td>
</tr>
</tbody>
</table>

PVCNN outperforms F-PointNet++ by 2.4% mAP with 1.5x measured speedup and 1.4x memory reduction.
Results: 3D Object Detection (KITTI)

PVCNN outperforms F-PointNet++ by 2.4% mAP with
1.5x measured speedup and 1.4x memory reduction.
Point-Voxel CNN for
Efficient 3D Deep Learning

Bottleneck Analysis

Hardware-Efficient Primitive