Anytime-Lidar: Deadline Aware 3D Object Detection

Ahmet Soyyigit\textsuperscript{1}, Shuochao Yao\textsuperscript{2}, Heechul Yun\textsuperscript{3}

\textsuperscript{1,3} University of Kansas, Lawrence, KS
\textsuperscript{2} George Mason University, Fairfax, VA
Perception in Autonomous Vehicles

• Object detection
  – Happens in 3D
    • Camera, Radar, Lidar, ...
  – Lidar-based deep neural networks
  – Timeliness
  – Time/accuracy requirements are environment dependent

Image credits(down): https://newsroom.intel.com/editorials/experience-counts-particularly-safety-critical-areas/#gs.8azpk6
Lidar-based Object Detection DNNs

• Point cloud to 3D bounding boxes (End-to-end)
  – Examples: Voxelnet, SECOND, PointPillars, CenterPoint

• Challenges: High computational cost, deadline-unaware
Execution Time Analysis of PointPillars

- Timing of PointPillars*:
  - High computational cost (>130 ms)
  - No flexibility in execution timing

(*) Executed on Jetson AGX Xavier
Architecture of PointPillars (multi-head)

Point Cloud Transform → Backbone (RPN) → Detection Head(s)*

Block 1
- Conv → Deconv

Block 2
- Conv → Deconv

Block 3
- Conv → Deconv → Concat 1,2,3

Detection Head (Car) → • → Detection Head (Traffic cone, Pedestrian) → NMS

Anytime Perception for Lidar-based Object Detection DNNs

• Enable dynamic time and accuracy tradeoff
• Prior work on anytime perception
  – Image-based, mostly object classification [1-6]
• Our key contribution
  – First work to enable anytime perception in the lidar domain
  – Novel scheduler framework: Accuracy + Timeliness

Outline

• Introduction
• Anytime-Lidar
• Evaluation
• Conclusion
Anytime-Lidar

- Enable anytime perception for lidar-based object detection DNNs
  1. Imprecise computation on the backbone
  2. Scheduling of detection heads
  3. Predicting past results of skipped heads
  4. Scheduling the above three
Imprecise Backbone

- Time and accuracy trade-off by skipping blocks
  - Added early exists to skip block 3 or blocks 2+3
  - Each block takes equal time
  - Take advantage of multi-block structure
Schedulable Detection Heads

• Allow skipping a subset of detection heads
  – Linearly save time from convolutions and NMS
• Address safety concerns
  – Proper det. head scheduling
  – Projection
Projection

• Project the past results of skipped det. heads to the current frame
• Projection/CPU - NN/GPU parallel execution
Scheduling

- Maximize detection accuracy while meeting the deadline with two-phase scheduler.
Scheduling

• **First scheduling phase**: Determine the **number of backbone blocks** and the **number of detection heads** to run

• Done using time/accuracy statistics collected offline

<table>
<thead>
<tr>
<th>RPN blocks</th>
<th>Detection heads</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>30.9</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

*Numbers are in milliseconds.*

WCET table

Normalized accuracy table
Scheduling

• **Second scheduling phase**: Decide which detection heads to execute
  – Provides safety while optimizing accuracy
  – Priority = Age x Confidence

\[
\begin{align*}
\text{Car} & : 1 \times 3.5 = 3.5 \\
\text{Truck, Constr. Vehicle} & : 2 \times 0.7 = 1.4 \\
\text{Bus, Trailer} & : 3 \times 0.6 = 1.8 \\
\text{Barrier} & : 3 \times 2.0 = 6.0 \\
\text{Motorcycle, Bicycle} & : 4 \times 1.2 = 4.8 \\
\text{Traffic cone, Pedestrian} & : 1 \times 4.5 = 4.5 \\
\end{align*}
\]
Outline

- Introduction
- Anytime-Lidar
- Evaluation
- Conclusion
Evaluation

• Implemented by modifying Multi-head PointPillars (OpenPCDet*, PyTorch)

• Evaluated on NVIDIA Jetson AGX Xavier
  – 512-core Volta iGPU
  – 8 core ARM v8.2 64-bit CPU
  – 16 GBs of RAM

• Evaluated using nuScenes dataset
  – Used ten scenes each being 20 seconds

Evaluation

• Divide the dataset of ten scenes into two equal sets
  – Calibration set
  – Testing set
• Collect time/accuracy statistics for all requiring methods (calibration)
• For each method being evaluated:
  – For each deadline in a list of deadlines from 140ms to 60ms:
    • Process all samples in the testing scenes one by one
    • Nullify detection results for samples where deadline is missed
    • Calculate NDS* (nuScenes Detection Score)

Evaluation

• Methods used for comparison:

<table>
<thead>
<tr>
<th>Method</th>
<th>Number of model parameters</th>
<th>Number of RPN blocks</th>
<th>RPN stage selection</th>
<th>Detection head scheduling</th>
</tr>
</thead>
<tbody>
<tr>
<td>PointPillars-3</td>
<td>6078K</td>
<td>3</td>
<td></td>
<td>Circulating</td>
</tr>
<tr>
<td>PointPillars-2</td>
<td>2626K</td>
<td>2</td>
<td></td>
<td>Class scores sum</td>
</tr>
<tr>
<td>PointPillars-1</td>
<td>1723K</td>
<td>1</td>
<td></td>
<td>Aging + Ground Truth</td>
</tr>
<tr>
<td>MultiStage</td>
<td></td>
<td></td>
<td></td>
<td>Aging + Aged confidences</td>
</tr>
<tr>
<td>RoundRobin</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ClsScrSum</td>
<td>9235K</td>
<td>3</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>NearOptimal</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Effect of Enabling Fine-grained Anytime Perception

- Meet tighter deadlines (60ms vs 100ms)
- Maintain superior accuracy all the time
Effect of Head Scheduling Method

- Disabled projection when testing
- Our method schedules the detection heads close to optimal

**Overhead**
- 4.75 ms
- 0.50 ms
- 1.50 ms
Effect of Projection

- Projection can work with any head selection scheme and increases accuracy by 10% on average.
Conclusion

• In this work, we presented:
  – A novel scheduling framework for lidar-based AI pipelines
    • Enables anytime perception through a combination of methods
      – Imprecise backbone, detection head scheduling, projection
  – We implemented our method on Multi-head PointPillars and evaluated its performance on Jetson AGX Xavier
  – Results show that our method significantly surpass baseline methods and enables anytime perception for lidar-based AI pipelines

• GitHub Link: https://github.com/CSL-KU/Anytime-Lidar
Thank You

Disclaimer:
This research is supported in part by NSF grants CNS1815959, CPS-2038923, and CPS-2038658

More details can be found in the following publication.