PAPER ANALYSIS

Presented by Yannis He

-

Paper: **LiDAR Semantic Segmentation: A Review**

Authors: Ehsan Taghavi, Ryan Razani, Sinclair Hudson

[*paper currently under review, no link to paper publicly available*]

INTRODUCTION

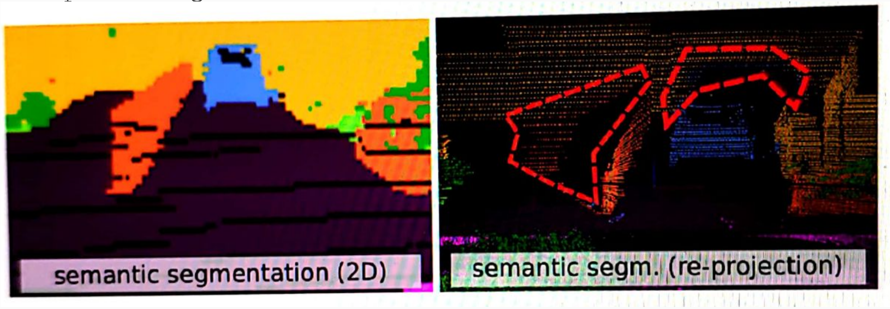
- Background:
  - LiDAR are important since its the only sensor that provides 3D geometric information of the robot (autonomous vehicle"'s surroundings with high accuracy and density
  - Semantic point cloud segmentation is an important in understanding an autonomous vehicle scene semantically.
- Problem Statement:
  - Given a set of points P, and n classes, assign each point to a class.
  - Formally: $Y = F(P; \theta)$
    - We seek a bijective function F that, given P points, produces a set of Y labels.
    - Each point $p \in P$ corresponds one-to-one with a $y \in Y$.
    - F is parametrized by θ, which is modified to minimize the difference between Y and the true label
- Dataset:

Dataset	Classes	Labelled FOV	Total Frame	Label Derivation
KITTI	9	90	41000	3D Bounding Boxes
SemanticKITTI	20	360	41000	Manually Labelled
A2D2	38	360	38481	Image Segmentation

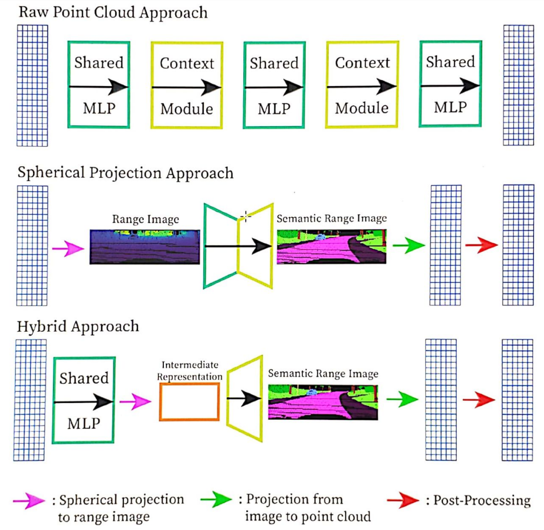
- **Semantic Segmentation Approach Summary**
  1. Unordered Point Cloud
  2. Spherical Projection of Data
  3. Hybrid of Available Techniques
  4. Bird's Eye View Representation
  5. Multi-View Fusion
- **1. Unordered Point Cloud: Y = MLP(CM(MLP(CM(...P))))**
  - Earliest approach, which uses raw point clouds
    - Pioneering implementations: PointNet, PointNet++
  - Input: a list of points, each with x, y, z, remission values
  - Output: a list of class labels
  - Techniques:
    - Multi-Layered Perceptrons
    - Kernel-based Feature extraction methods
    - Use of context modules (to increase the receptive field of the network)
  - Pros:
    - Proven success for small point clouds
  - Cons:
    - Computationally expensive and Slow implementations
    - Massive memory requirements for large point clouds
    - Hard to achieve the benchmark for real-time computing when using raw point cloud

Process a point cloud of 150,000 points in less than 100ms

- **2. Spherical Projection of Data: $Y = Post(J^{-1}(CNN(J(P))))$**
  - Projecting the LiDAR point cloud into a spherical coordinate system to create a 2D range image
    - Point cloud segmentation → image segmentation
    - $R^3 \rightarrow R^2$
  - Pros:
    - Fast calculation
    - CNNs, which were well studied, can be applied
  - Cons:
    - Cannot effectively represent occluded objects effectively
    - Projection from image back to point cloud is one-to-many
      - I.e. Shadow Problem: Semantic label of a single pixel gets mapped to multiple points, which could lead to erroneous predictions around the boundary
      - Post-processing needed to reduce those errors



- **3. Hybrid of Available Techniques: Y = Post(H(P))**
  - Use combination of techniques from raw point cloud approaches, BEV methods, and spherical projection approaches
- **4. Bird's Eye View (BEV) Representation: $Y = (G^{-1}(CNN(G(P))))$**
  - Process:
    - Point cloud is descretized into pixels based on their x and y coordinates using a function $G: R^3 \rightarrow R^2$
    - The resulting 2D images is then processed by a fully convolutional network that produces a class prediction for each pixel
    - Then, the predictions are mapped back into the point cloud using G^{-1}.
- **5. Multi-View Fusion:**



All methods take a LiDAR point cloud as input, and produce a LiDAR point cloud with per-point semantic predictions

RELATED PUBLICATIONS

- Overview:
  - PointNet & PointNet++
  - SequeezeSeg & SqueezeSegV2
  - Recurrent CRF Post-Processing
  - PointSeg
  - RangeNet++
  - DeepTemporalSeg
  - DBLidarNet
  - Naive Bayes Filter
  - RandLA-Net
  - LocSE module
  - Attentive Pooling module
  - Local Feature Aggregation module
  - Lattice Net
  - Distribute
  - DeformSlice
  - SalsaNet and SalsaNext
  - LU-Net
  - 3D-MiniNet
  - Projection Learning Module
  - Local Feature Extractor
  - Context Feature Extractor
  - Spatial Feature Extractor
  - Feature Fusion
  - MiniNet Backbone
  - KNN Post-Processing
  - KPConv
  - Deformable Kernels
  - Network Structure
  - PolarNet
  - Network Structure

METRICS & NOVEL APPROACHES AND TECHNIQUES

- Metrics: Mean Intersection over Union
  - Most popular metric for evaluating semantic point cloud segmentation:
    - Average of IoU for each class

$$IoU = \frac{TP}{(TP + FP + FN)}$$

- Novel Approaches and Techniques:
  - Point Cloud Semantic Segmentation Techniques
    - Late concatenation of range image
    - Augmenting the dataset
    - More per-point Features
    - Learned per-point Features
    - Normalization
    - Temporal filtering