

Semantic Segmentation using PointSeg Networks

- Train CNN to perform semantic segmentation of 3-D lidar point clouds.
- Each point is represented by label class, eg: cars, trucks etc.
- PointSeg is an end-to-end semantic segmentation CNN for road objects.
- It uses state-of-the-art methods like atrous spatial pyramid pooling (ASPP) & squeeze-and-extraction block to improve results.
- 2-D convolutions are faster than 3-D convolutions. Instead of 3-D lidar data as input network, the lidar data is first projected into 2-D image then given as input to NN.

Steps.

① Download & create five-channel images from lidar point clouds.

→ It prepares dataset for training by generating images & labels from lidar point cloud using function:

helpsGenerateTrainingData(lidarData, groundTruthData, imageFolder, labelFolder).

→ five-channels of images are:

x, y, z, intensity, range.

→ Ground truth data will have information about each point cloud belong to which class (class ID).

→ Images are saved as .mat files and labels are saved as .png files.

② Create a pixelLabelImageDatstore object to feed training data to the network.

→ pixelLabelImageDatstore object is used to combine (1) training data, label data & data augmentation for training purpose.

(2) Validation data & label data for validation purpose.

→ `ImageDataset` object extracts the five channels from the disk (r, g, b, intensity, range).

→ `pixelLabelDataset` object to read lidar pixel label image data.

→ In this step preparation of dataset for training, testing and validation is done.

③ Define network architecture

→ `createPointSeg` object creates the PointSeg network to train the data.

④ Specify training options

In trainingOptions function specify the

hyperparameters (eg: epochs, learning rate, batch size etc) to train the network.

→ for optimization used simprop algorithm as ~~an~~ argument to trainingOptions function.

(5) Train the network

→ `trainNetwork` function is used to train the data.

→ It takes training data and training options (step 4) as an input.

(6) Evaluate network performance.

~~`semanticsSeg` function~~

(a) Predict results on Test point cloud data.

→ `semanticsSeg` function is used to predict results on the test point cloud data.

→ This function takes trained model & test point cloud data.

→ This function returns the segmentation results on the test dataset.

(b) Validate network.

→ `evaluateSemanticSegmentation` function returns matrices 1) for entire dataset 2) individual classes 3) for each test image.

→ `metrics.classMetrics` → provides high-level overview of network performance.