

Homework 3

CSE 291 I00: Machine Learning for 3D Data (Winter 2018)

PLEASE READ:

Submit printouts of well commented codes on March 7. You are encouraged to discuss in groups (two people for each group), but please complete your programming and writeup individually. Indicate the name of your collaborator in your writeup.

This Programming Assignment is to help you kickstart your final project. You will be implementing a simplified version of PointNet (<https://arxiv.org/pdf/1612.00593.pdf>) for classification. You do not need to implement the Segmentation module. Please make sure to read the paper and understand the concepts before you begin the assignment. For both the parts, you must turn in a well commented code along with the results that you obtained.

1. PointNet (vanilla):

In this part of the assignment, you will implement the vanilla model of PointNet. The vanilla version of PointNet has no transformations (input and feature).

- Download ShapeNet40 from https://shapenet.cs.stanford.edu/media/modelnet40_ply_hdf5_2048.zip. This will have the required test and train files in h5 format. Use the provided data loader to load in the point clouds. Familiarize with the data by loading it in python. Each train h5 file consists of point clouds of 2048 points each. The 40 classes are given in the file *shape_names.txt*. Visualize the points for a few objects.
- Construct functions for data augmentation. The pointnet paper used two augmentation techniques: rotation and jittering. The rotation function should take as input, a point cloud and rotate it by a certain angle. For jittering, add random noise to the point cloud. (10 points)
- Using Tensorflow implement the architecture of PointNet as shown in the figure. For this part of the assignment, you do not need to implement the Feature and Input Transformations. For testing, use the test data and the provided evaluation functions and try to reproduce the results of the paper as much as possible. (40 points)

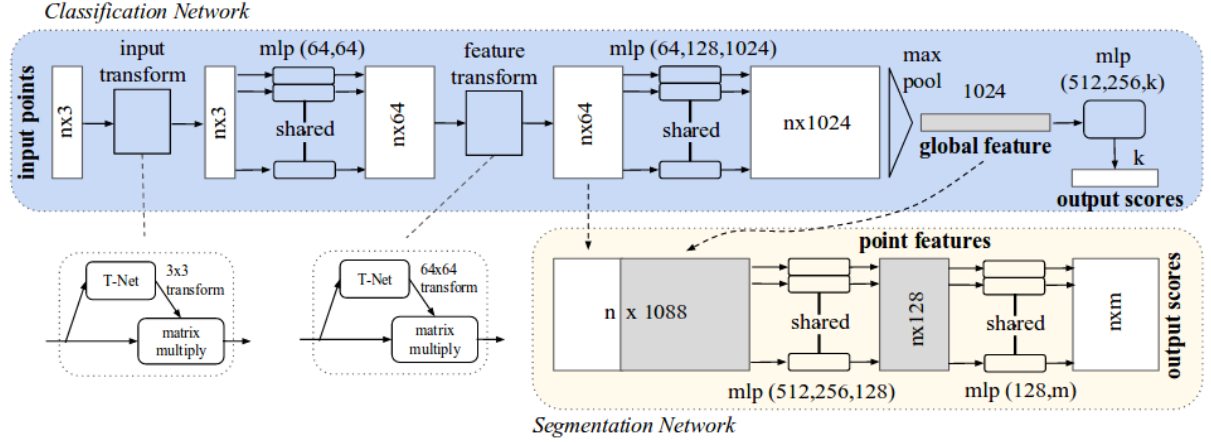


Figure 1: PointNet Architecture

2. PointNet:

- This part of the assignment is an extension of the previous part. In this part, implement the architecture of the PointNet with both input and feature transformations.
- Implement both the transformation modules. The input transformation module should take as input, a batch of point clouds and return a 3×3 transform matrix. The feature transform module should return a transformation matrix of dimension 64×64 . (30 points)
- Incorporate the modules implemented above with the PointNet Vanilla architecture to get the PointNet architecture as shown in Figure 1. Report the accuracy obtained on the test set. (20 points)