CIS 565 Final Project Proposal

Saket Karve, Kushagra Goel, Disha Jindal

November 4, 2019

1 Overview

The aim of our project is 3D point cloud classification using Graph convolutional neural networks on CUDA. GCNs are very effective because they efficiently exploit the local structure in point clouds. The plan is to follow the algorithm outlined in the paper [1], implement it on CPU as a benchmark and then on GPU using CUDA. The architecture involves nearest neighbor search to construct the graph from the point clouds. So we'll use efficient data structures (for instance: KD-Trees) to achieve that and do performance analysis. The graph layers take a graph as input and propagate the input preserving the graph structure across layers. Graph neural networks are an ongoing topic of research and have applications in various fields. Their architecture is slightly complicated and it might be challenging to implement back-propagation on the graph layers. We have planned some stretch goals (if time permits) to explore other different convolution layers and do a comparative study of all techniques used. One of those [2] is proposed to be replacement for traditional convolution layers.

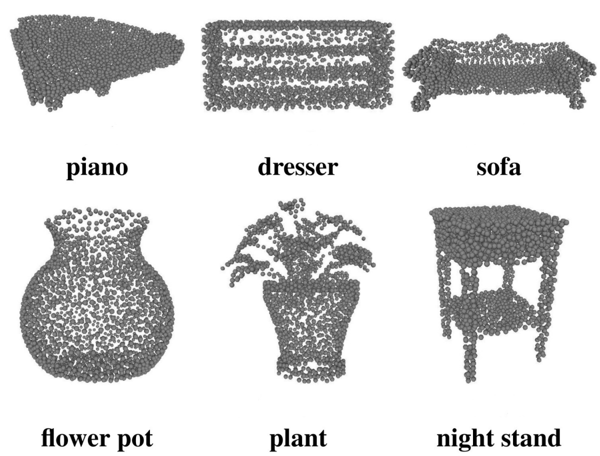

Figure 1: Sample images from Dataset

2 Milestones

- November 18 - CPU version of GCN
- November 25 - KD-Tree implementation for constructing graphs on GPU
- December 2 - Build and train GCN model using CUDA
- December 9 - Comparison and analysis of different architectures and implementations. Inference and visualizing the output.

References

[1] Yingxue Zhang and Michael Rabbat. A graph-cnn for 3d point cloud classification. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6279–6283. IEEE, 2018.

[2] Wolfgang Fuhl, Gjergji Kasneci, Wolfgang Rosenstiel, and Enkelejda Kasneci. Training decision trees as replacement for convolution layers. *arXiv preprint arXiv:1905.10073*, 2019.

[3] Lingxiao Ma, Zhi Yang, Youshan Miao, Jilong Xue, Ming Wu, Lidong Zhou, and Yafei Dai. Towards efficient large-scale graph neural network computing. *arXiv preprint arXiv:1810.08403*, 2018.