

Report of 3D Point Cloud Reconstruction using Deep Learning

Yipeng Liu, Yide Ma, Zhuocheng Guo, Mingwei Sui

Introduction:

In our project, we were given 4 groups of plants as datasets. We compared the performance of 6 types of point completion networks on plant-related point cloud datasets

Datasets:

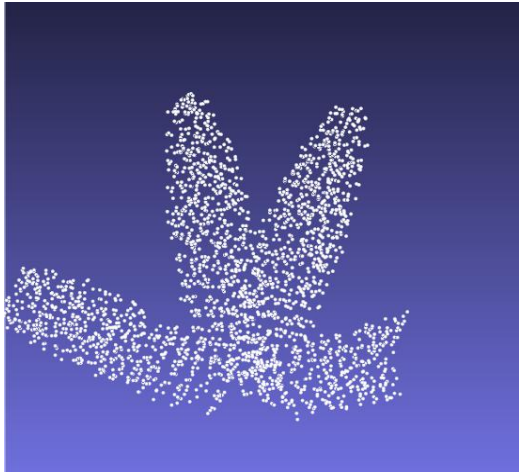


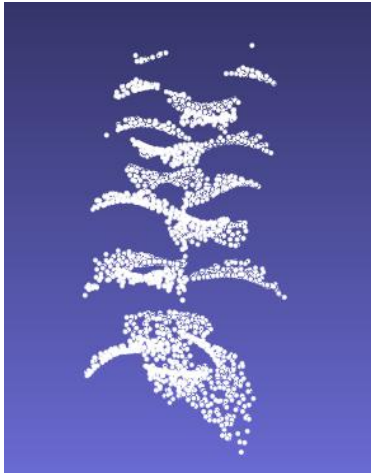
Using Virtual Lab creates .obj files. then using CloudCompare creates .pts files
Each plant includes 1000 samples. Each sample consists of about 3000 points. (Mimosa, Soybean, Corn, Bush)

Neural Networks:

PCN: Point Completion Networks	Point Cloud Completion by Learning Multi-step Point Moving Paths	FoldingNet: Point Cloud Auto-encoder via Deep Grid Deformation
PoinTr: Diverse Point Cloud Completion with Geometry-Aware Transformers	Snowflake Net	PFNet: Point Fractal Network for 3D Point Cloud Completion

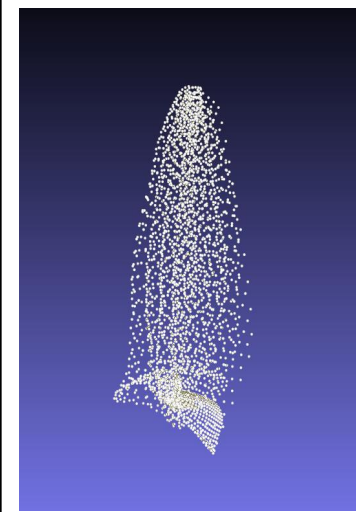
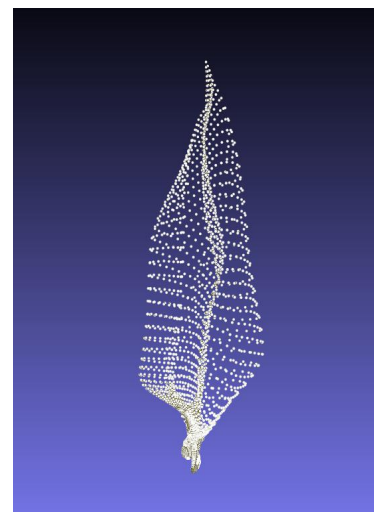
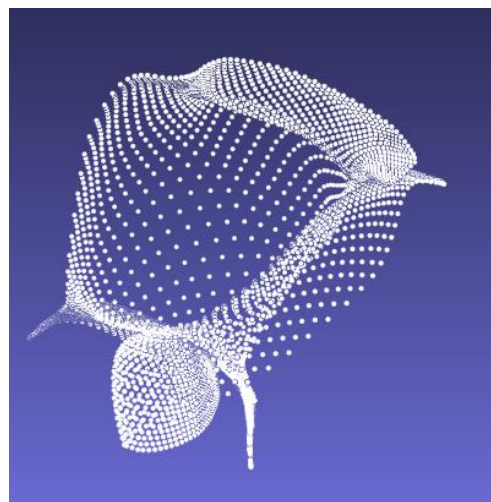
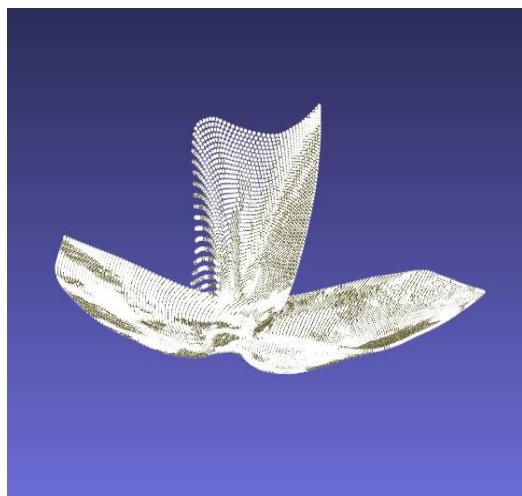
Results:

The following five nets output the whole reconstructed plant, so they may change some origin points of the input. We set approximately 4~5 hours for training one network for each plant. This is for making our comparison more reasonable. The first row “Partial” shows the four input point clouds, and the following rows will show the prediction of each network.

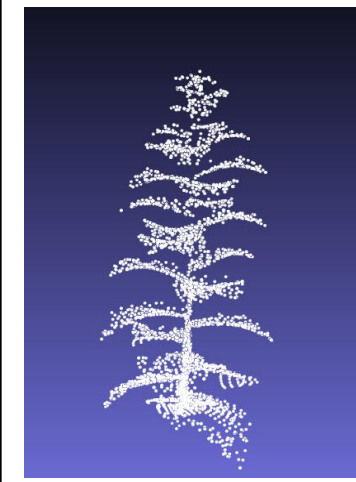
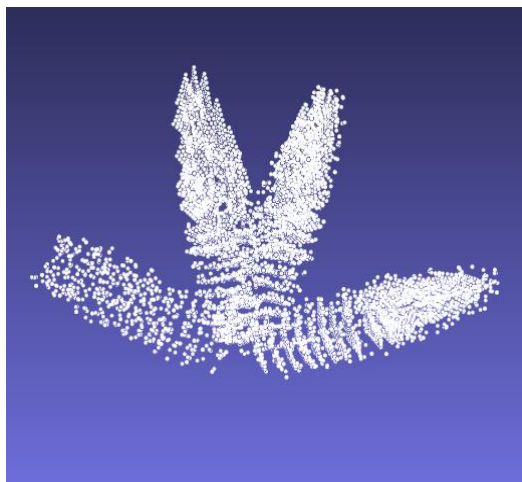
	Mimosa	Soybean	Corn	Bush
Partial				

PCN				
PMP				

Folding
Net

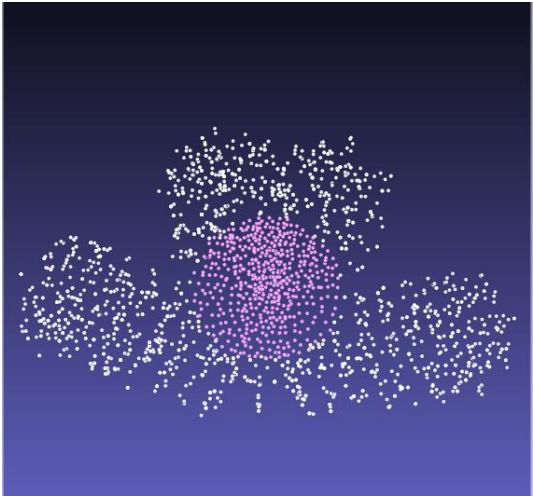

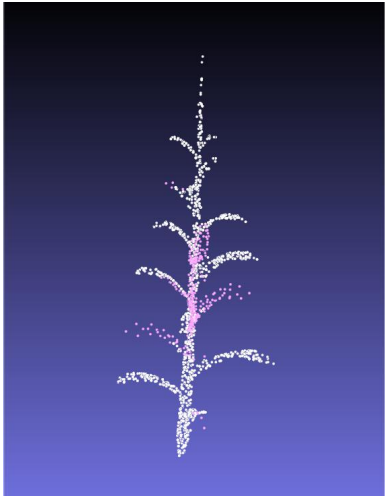
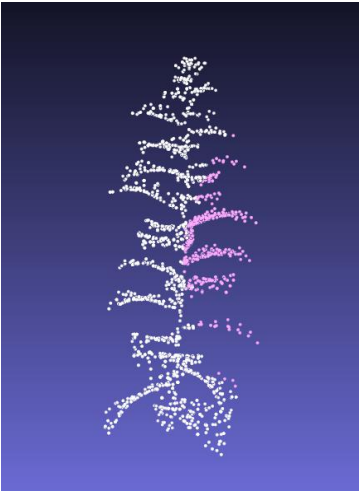


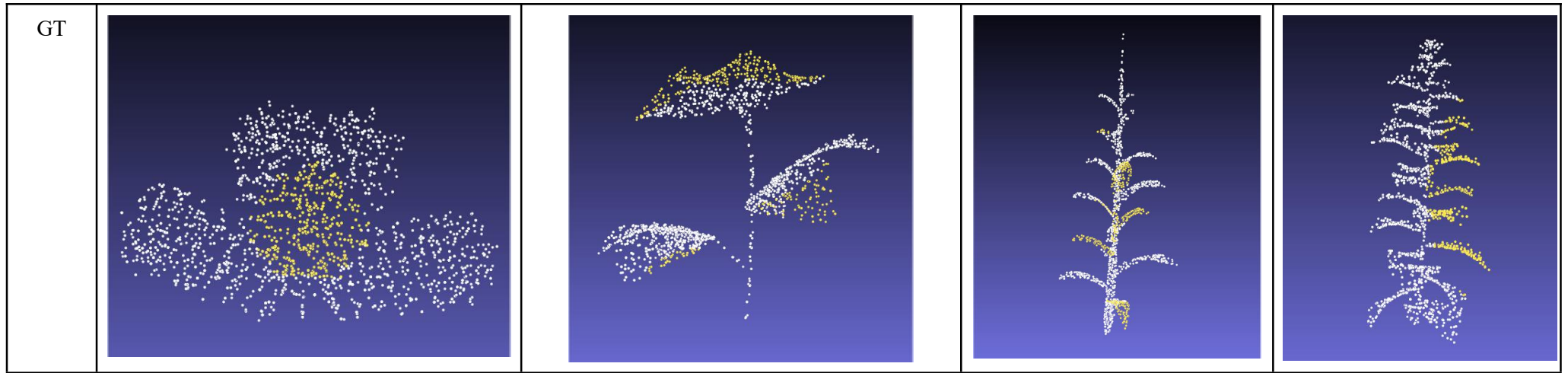
PoinTr
Net



<p>Snow Flake Net</p>				
<p>GT</p>				

PF-Net only outputs the missing part of the input plants, it retains all origin points of input. The white part is the incomplete point cloud, the pink part is the model prediction, and the yellow part is the ground truth.

	Mimosa	Soybean	Corn	Bush
PF-net				



Losses:

	mimosa	soybean	corn	bush
Folding Net	0.9069	2.8848	2.0156	5.272
PMP	0.4993	0.7724	1.6528	0.5981
PCN	0.0766	0.1015	0.1306	0.1470
SnowFlakeNet	0.6651	0.6443	0.6909	0.9219
PoinTr Net	0.0971	0.1201	0.1252	0.4141
PF Net	0.000633	0.001885	0.000804	0.001906

Overall, we find that the reconstruction of mimosa has the least loss among all four plants. This is because mimosa has the simplest shape, which is easiest to train. Bush has the most complicated 3 dimensional shape, which is the hardest to train. Compared to the losses, we can find the PF-Net is the best model in our project.

Difficulties:

In this project, the main difficulty is to debug and re-implement the neural nets. The dataset we test every neural nets are 1000 plants for each group and 3000 points for each point cloud. We control the training time for each group of plants using one net around 4~5 hours. In this way, the spend on training is almost the same even if we have different epochs for each net.

Analysis:

When we just started this project, we searched for these nets and had an impression in our mind. PCN uses MLP to understand the global feature and try to generate a shape based on the information it gains. So for PCN, it could have good performance when you input a larger amount of points in each point cloud data and more time for training. When we have equal training time, its performance might be bad at the detailed part. For PF-Net, we have trained it for several times and we found in PF-Net, one phenomenon always happening. Some of its points are sometimes “gathering” at places where the shape changes rapidly. However, there are also some advantages of PF-Net. Since it only outputs the missing part, its performance is good at the detailed part. In addition, there is one problem with PF-Net. When we are trying to give it more epoch to run, even 1000 epochs with 24 hour training time. The performance is not changing at all. We did some parameter tuning and the results remained the same. Another net we have re-implement is the folding net. For the folding net we have met the situation mentioned in the essay, it needs a very large epoch to train, even more than PCN. Even though right now the shape looks far from what we expect, but in a very large epoch, it should perform very well. Hence, when we train for around 4~5 hours, its performance is far from what we want. PMP is one net we have implemented. We think this net is not suitable for plant based point completion. PMP uses movement between points to understand the global and partial features. We think the reason for bad performance using PMP net is, it needs more density for each point cloud data. For snowflakes, when we use its original given learning rate, 0.001, the gradient explodes. Once we change the learning rate to 0.0001, it works perfectly. For snowflakes, when we are doing research we find it summarizes all good properties from recent point completion nets. In conclusion, PF-Net has the least loss under our setting of comparison, PoinTr net is the second net with good performance in our project.

References

- B. Li, Y. Zhang, B. Zhao and H. Shao, "3D-ReConstnet: A Single-View 3D-Object Point Cloud Reconstruction Network," in IEEE Access, vol. 8, pp. 83782-83790, 2020, doi:10.1109/ACCESS.2020.2992554.
- C. -H. Lin, C. Kong, and S. Lucey, "Learning Efficient Point Cloud Generation for Dense 3D Object Reconstruction", AAAI, vol. 32, no. 1, Apr. 2018.
- Huang, et al. "PF-Net: Point Fractal Network for 3D Point Cloud Completion." ArXiv.org, 1 Mar. 2020, arxiv.org/abs/2003.00410.
- Jie Mei, Liqiang Zhang, Shihao Wu, Zhen Wang & Liang Zhang (2016): 3D tree modeling from incomplete point clouds via optimization and L1-MST, International Journal of Geographical Information Science, DOI: 10.1080/13658816.2016.1264075.
- Liu, et al. " MSN-Net: Morphing and Sampling Network for Dense Point Cloud Completion. " AAAI2020, 2020, <http://cseweb.ucsd.edu/~mil070/projects/AAAI2020/paper.pdf>.
- M. Cheng, G. Li, Y. Chen, J. Chen, C. Wang and J. Li, "Dense Point Cloud Completion Based on Generative Adversarial Network," in IEEE Transactions on Geoscience and Remote Sensing, doi: 10.1109/TGRS.2021.3105551.
- Pan, L., Chen, X., Cai, Z., Zhang, J., Zhao, H., Yi, S. and Liu, Z., 2021. Variational Relational Point Completion Network. [online] Paul007pl.github.io. Available at: <<https://paul007pl.github.io/projects/VRCNet>> [Accessed 12 July 2021].
- P. Mandikal and V. B. Radhakrishnan, "Dense 3D Point Cloud Reconstruction Using a Deep Pyramid Network," 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), 2019, pp. 1052-1060, doi: 10.1109/WACV.2019.00117.
- W. Zhang, C. Long, Q. Yan, A. H. and C. Xiao, "Multi-Stage Point Completion Network with Critical Set Supervision", Graphvision.whu.edu.cn, 2021. [Online]. Available: <http://graphvision.whu.edu.cn/papers/zhangwenxiao2020GMP.pdf>. [Accessed: 30- Mar-2020].