CURVATURE INFORMED FURTHEST POINT SAMPLING

Anonymous authors
Paper under double-blind review

ABSTRACT

Point cloud representation is becoming increasingly popular due to its low memory footprint, ease of creation, collection and modification. As the size of the point cloud increases, we need to incorporate a down-sampling operation to meet the computational demands of the tasks. Classical approaches such as farthest point sampling perform exceedingly well over downstream tasks. The major drawback is that farthest point sampling is a mere heuristic and does not take geometric priors such as curvature into consideration. We propose a novel sampling procedure that conditions the output of farthest point sampling with curvature information. We create a joint rank by multiplying the soft furthest point rank with corresponding curvature scores obtained via a deep neural network and exchange a percentage of low-ranking points in the furthest set with the high-ranking points in the left-out set. Previous differentiable sampling approaches have failed to conform to the end-to-end learning paradigm due to instability while training. We demonstrate that our algorithm is compatible with end-to-end learning. Our sampling scheme consistently outperforms previous baselines on various downstream geometry processing tasks. Finally, we show detailed ablation studies regarding the qualitative and quantitative analysis of the role of different features used in the proposed algorithm.

1 INTRODUCTION

Applications making use of 3D data are becoming increasingly popular over the recent years. These applications usually accept data in the form of a point cloud. A point cloud representation represents the visual scene via colored points in 3D space. For reducing the computational burden that is presented by large scale point clouds we need to reduce the size of the point cloud. The reduction in the size of the point cloud is generally done via sampling. The sampling operation tends to preserve local shape details and overall shape structure of the point cloud.

The main question is, how to select the data points that preserve local shape details and overall shape structure of the point clouds? A simple and effective approach would be to sample k farthest points within the point cloud, (where k is the number of points required in the downsampled point cloud). This traditional approach is called farthest point sampling (FPS). FPS is used widely by the community due to its robustness and effectiveness over different types of downstream tasks.

A major drawback of the FPS approach is that the algorithm is task agnostic. Therefore, FPS algorithm would not be able to prioritize sampling points that are highly relevant to the downstream tasks. For example, points with high curvature and low density may be highly relevant for downstream tasks such as reconstruction. We can compare our subsampling task of points in 3D space to the 2D image space. In 2D space max pooling selects the most activated pixel feature from a local spatial grid. In 3D space we need an algorithm that performs the same task. The local spatial grid is defined by the K-nearest neighbor point features. We cannot select the farthest point features directly because the point features belong to a lifted higher-dimension space. Therefore, FPS cannot work directly in the feature space.

A simple formulation would model the subset creation problem of point clouds via MLP layers. The main issue with the simple formulation is that end-to-end learning becomes highly unstable for downstream tasks, since we may lose the entire structure of the object while creating a random subset of points. Therefore, recent approaches have followed a two step approach. The first step involves learning a model with farthest point sampling as an intermediate point cloud subset creation layer.

The second step involves freezing all the layers of the trained model and adding new MLP layers to replace the farthest point sampling layer. In this step, the aim is to train the sampling layers only. The two step procedure is useful in practice and outperforms the traditional farthest point sampling approach. The drawback of the 2 step approach is the overhead that comes along with re-training the model. The drawback hinders the mass adoption of such 2-step approaches.

In our proposed algorithm, we propose a layer called conditional farthest point sampling (CFPS) that is robust enough to allow for end-to-end learning over downstream tasks while outperforming all previous traditional and differentiable formulations of the point cloud subsampling problem. Our initial step begins with creating a soft rank out of furthest point feature based on when a point entered the furthest point set. We then create a joint rank J by multiplying the soft furthest point rank with corresponding curvature scores obtained via PCPNet Guerrero et al. (2018). Finally, we replace x% the points in the furthest point having the lowest joint rank with the points that could not make it to the furthest point set ,yet, still had high joint rank. Curvature along with the FPS feature ensure that the point subsets selected during the initial phases of training truly characterize the shape's local and global geometry. The clear benefit for the algorithm is that it can show out of the box better performance that Furthest Point Sampling by selecting the right x% value. We summarize our contributions as follows :

- We propose a subsampling layer that is compatible with the end-to-end learning paradigm.
- We extend the farthest point sampling algorithm with traditional and layer specific features for stable task-specific subset selection.
- W.

Note that works such as SampleNet Lang et al. (2020), Learning to Sample Dovrat et al. (2019) do not conform to our paradigm for comparison. Our method and the baselines are proposed for an end-to-end learning model wherein down-sampling will happen multiple times throughout the network. The approaches Lang et al. (2020)Dovrat et al. (2019) are proposed for a 2 step training approach where sampling only happens once. i.e before sending the input point cloud to the network.

2 Related work

2.1 Deep Learning on Point Clouds

Automated analysis of 3D data using deep learning has led to huge improvements on multiple tasks such as object detection Lang et al. (2019)Yang et al. (2018), object segmentationQi et al. (2017a) , registration Yang et al. (2020) etc. Directly using the raw point cloud offers numerous advantage. Namely,

- Ease of capture - Point cloud data of 3D world can easily captured via 3D scanners or photogrammetry based methods Liu et al. (2019).
- Low-memory footprint.
- High fidelity Representation - 3D shape reconstruction modules relying on point cloud representation tend to produce higher fidelity shapes over other popular representation such voxels Pan et al. (2021a).
- Diverse Applications - Point Cloud representation is used by multiple vision and robotics tasks such as robot navigation and perception, depth estimation Wang et al. (2021b).

A lot of research has been on learning different kinds of convolution operations Atrous ConvolutionPan et al. (2020) KP ConvThomas et al. (2019) GATVelickovic et al. (2017) for point clouds. These convolution operations generally work on point clouds of fixed sizes. To cope with ever increasing sizes of point clouds data volumes such as scenes, we need to down-sample point cloud features after each convolution. The down sampling operation's job is to subsample point cloud features that are relevant to the downstream task. The research on down-sampling tasks has seen little to no progress in the recent years. The de-facto algorithm for downsampling in most point libraries is - Furthest Point Sampling.

2.2 Furthest Point Sampling (FPS)

In the FPS Eldar et al. (1997) algorithm, we start with a point in the set, and then select the point to be added to the set which is furthest from all the points in the set. The whole process keeps repeating till we get the required number of points in the set. The aim of the algorithm is to achieve maximum coverage of the shape. FPS is a task-agnostic algorithm Qi et al. (2017b) Yu et al. (2018) Li et al. (2018). Nevertheless, the algorithm works very well in practice. The main drawbacks of the farthest point sampling algorithm are as follows -

- FPS is non-differentiable (task-adaptive)
- FPS is designed for low-dimensional euclidean space as opposed to high dimension feature space.
- FPS is sensitive to outliers.

2.3 Critical Points Layer

The Critical Points Layer (CPL) Nezhadarya et al. (2020) is a differentiable down-sampling modules that work directly with the high-dimension point level features. The critical points layer passes the most active features to the next layer.

The down-side of directly working in the feature space is that during initial few epochs of the training stage the down-sampling layer would take decision based solely on features that themselves have not yet generalized to the downstream task. This observation suggests that the network would have to take larger number of epochs to converge, as highlighted in the paper.

3 Proposed Solution

In this section, we propose a down-sampling algorithm that builds upon farthest point sampling. In the algorithm, we use farthest point sampling algorithm to create a binary feature which classifies a point as belong to the farthest point or not. Point Cloud Sampling methods such as Metzer et al. (2021) rely on an estimate of proxies for sparseness and sharpness of local regions to decide if a point is worthy of sampling or not. Intuitively, sparsely located points in a planar region can easily describe the planar shape whereas local regions with high density and curvature such as arm rest of a sofa need more points to retain the shape. The network learns to create the right trade-off between shape coverage, curvature based on the requirement of the down-stream task. We calculate robust point level features such as curvature and density as follows -

3.1 Robust Point Level Features

3.1.1 Curvature

Curvature is a simple and effective heuristic for sharp features Smith et al. (2019). Local curvature of each point is estimated, which is used as a proxy for sharpness. If normal information is not provided then we first estimate normal for each point via simple plane fitting technique Klasing et al. (2009). Then the difference in normal angles for each point and its k-nearest neighbors is calculated. The estimated curvature of a point is given by summing all of the angle differences with each of its neighbors.

4 Applications

4.1 Single-View Partial Point Cloud Completion

Scanned 3D point clouds are usually incomplete due to occlusions, noise and viewpoint of the 3D scanner which hampers the usability of the acquired 3D point cloud data. One of the drawbacks of using point cloud representation for Generative tasks such as shape completion is that, it is hard to represent rich topological shape details on the generated shape Wang et al. (2021a). We show in our experiments that the sampling procedure plays a fundamental role in improving the fidelity of the completed shape.

images/architecture_diagram.png

Figure 1: Architecture of Conditional Farthest Point Sampling Block

4.1.1 DATASET

We compare our results on the MVP datasetPan et al. (2021b) which was proposed in CVPR 2021.The MVP dataset is a large scale multi-view partial point cloud dataset having more than 100,000 high-quality scans. For each 3D CAD model, the dataset contains rendered partial 3D shapes from 26 uniformly distributed camera poses. The training set contains 62,400 partial-complete point cloud pairs and the Test set has 41,800 pairs. Each partial, complete point cloud has 2,048 points.

4.1.2 ARCHITECTURE

We use the VRCNet architecture baseline Pan et al. (2021a) for its proven high performance on the MVP dataset. The VRCNet model consists of 2 encoder-decoder sub-networks that aim to serve the “probabilistic modeling” (PMNet) and “relational enhancement” (RENet) tasks. The PMNet is responsible for producing a coarse down-sampled shape skeleton given an input partial point cloud. The shape skeleton gives cues about the global shape to the RENet module. The RENet module is responsible for enhancing the structural relations (local shape features) by learning multi-scale point features and reconstruct high quality point clouds from shape skeletons.

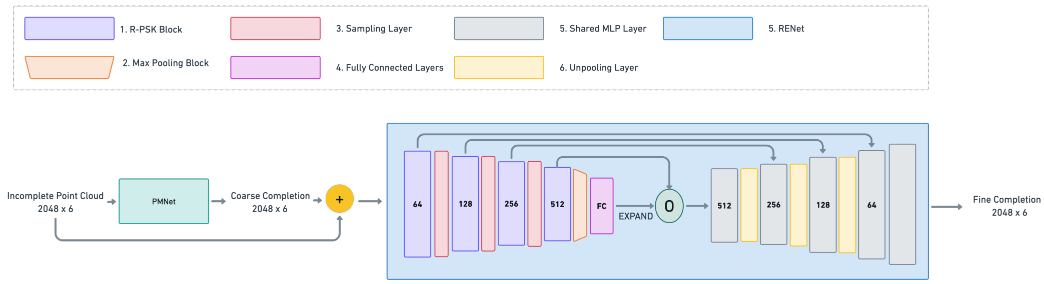

Figure 2: VRCNet Architecture for Shape Completion

Different relational structures can have different scales. Therefore, the R-PSK module learns to encode multi-scale feature information. For more information about the Unpooling and Feature Expansion modules refer, Pan et al. (2020) and Pan (2020) respectively. The feature expansion module is used to expand point features to produce high quality point cloud while keeping geometric consistency.

4.1.3 Training - Task Specific Loss

Given a partial shape, our model must be able to accurately reconstruct the ground truth shape by completing the missing point cloud. We use VRCNet baseline's loss function to improve the shape reconstruction capabilities of our task.

Our proposed algorithm CFPS, requires normal information. Refer Figure 1. Therefore, we also include a normal estimation loss as part of our task loss. The normal reconstruction loss is adapted from Sharma et al. (2021). Note that we do not include the Chamfer distance (CD) loss given in the original normal estimation loss in Sharma et al. (2021) as the Chamfer distance loss is already included in the VRCNet Shape reconstruction loss.

$$L_{task} = L_{Shape\ Reconstruction} + L_{Normal\ Estimation}$$

4.1.4 Evaluation - Loss

For encouraging accurate reconstruction bi-directional Chamfer distance (CD) Fan et al. (2017) metric is generally used for calculating the overall pairwise difference between the nearest points in the source and target shape. Previously, it has been shown that **CD** can be misleading due to outliers Tatarchenko et al. (2019). the benchmark suggests to evaluate using the F-score Knapitsch et al. (2017) to evaluate the distance between source and target surfaces. The F-score is the harmonic mean between precision and recall.

4.1.5 Implementation Details

We used the default training parameters as given in the official Pytorch Implementation of VRCNet in the MVP Benchmark repository. We directly used CFPS on the VRCNet model without any changes. We did minor ablation on the sampling ratio hyperparameter. Refer 3. Comparison against different methods on the dataset is presented in table 1

5 Conclusion

We show the improvements made by our proposed algorithm on the Shape Completion Task. The next steps would be to automatically detect the sampling ratio parameter from the input shape via few-shot learning.

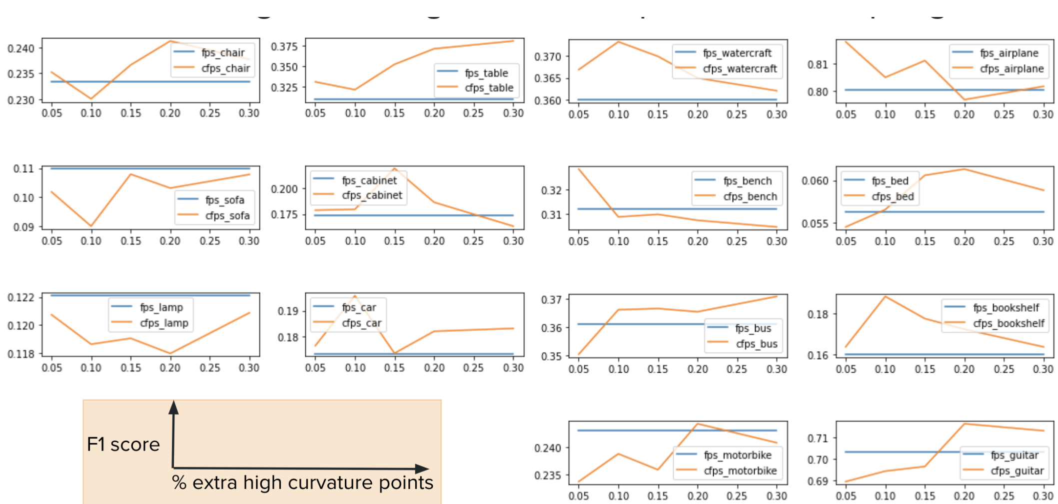

Figure 3: Result of changing sampling ratio on different classes in MVP dataset

Method	**CD**	**F1-Score@1%**
PCNYuan et al. (2018)	9.77	0.32
TopNetTchapmi et al. (2019)	10.11	0.308
CRNWang et al. (2020)	7.25	0.434
ECGPan (2020)	6.64	0.476
VRCNet + FPS	5.96	0.499
VRCNet-mod + FPS	5.96	0.48
VRCNet-mod + CFPS	5.8	0.50

Table 1: Comparisons of performance with existing methods on the MVP validation set. Both the input and output contain 2048 points. VRCNet-mod is VRCNet with Normal Estimation. CD loss multiplied by 10^4

REFERENCES

Oren Dovrat, Itai Lang, and Shai Avidan. Learning to sample. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2760–2769, 2019.

Yuval Eldar, Michael Lindenbaum, Moshe Porat, and Yehoshua Y Zeevi. The farthest point strategy for progressive image sampling. *IEEE Transactions on Image Processing*, 6(9):1305–1315, 1997.

Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 605–613, 2017.

Paul Guerrero, Yanir Kleiman, Maks Ovsjanikov, and Niloy J Mitra. Pcpnet learning local shape properties from raw point clouds. In *Computer Graphics Forum*, volume 37, pp. 75–85. Wiley Online Library, 2018.

Klaas Klasing, Daniel Althoff, Dirk Wollherr, and Martin Buss. Comparison of surface normal estimation methods for range sensing applications. In *2009 IEEE International Conference on Robotics and Automation*, pp. 3206–3211, 2009. doi: 10.1109/ROBOT.2009.5152493.

Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017.

Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12697–12705, 2019.

Itai Lang, Asaf Manor, and Shai Avidan. Samplenet: Differentiable point cloud sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7578–7588, 2020.

Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. *Advances in neural information processing systems*, 31, 2018.

Weiping Liu, Jia Sun, Wanyi Li, Ting Hu, and Peng Wang. Deep learning on point clouds and its application: A survey. *Sensors*, 19(19):4188, 2019.

Gal Metzer, Rana Hanocka, Raja Giryes, and Daniel Cohen-Or. Self-sampling for neural point cloud consolidation. *ACM Transactions on Graphics (TOG)*, 40(5):1–14, 2021.

Ehsan Nezhadarya, Ehsan Taghavi, Ryan Razani, Bingbing Liu, and Jun Luo. Adaptive hierarchical down-sampling for point cloud classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12956–12964, 2020.

Liang Pan. Ecg: Edge-aware point cloud completion with graph convolution. *IEEE Robotics and Automation Letters*, 5(3):4392–4398, 2020.

Liang Pan, Chee-Meng Chew, and Gim Hee Lee. Pointatrousgraph: Deep hierarchical encoder-decoder with point atrous convolution for unorganized 3d points. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1113–1120. IEEE, 2020.

Liang Pan, Xinyi Chen, Zhongang Cai, Junzhe Zhang, Haiyu Zhao, Shuai Yi, and Ziwei Liu. Variational relational point completion network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8524–8533, 2021a.

Liang Pan, Tong Wu, Zhongang Cai, Ziwei Liu, Xumin Yu, Yongming Rao, Jiwen Lu, Jie Zhou, Mingye Xu, Xiaoyuan Luo, et al. Multi-view partial (mvp) point cloud challenge 2021 on completion and registration: Methods and results. *arXiv preprint arXiv:2112.12053*, 2021b.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017a.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017b.

Rajat Sharma, Tobias Schwandt, Christian Kunert, Steffen Urban, and Wolfgang Broll. Point cloud upsampling and normal estimation using deep learning for robust surface reconstruction. *arXiv preprint arXiv:2102.13391*, 2021.

Edward J Smith, Scott Fujimoto, Adriana Romero, and David Meger. Geometrics: Exploiting geometric structure for graph-encoded objects. *arXiv preprint arXiv:1901.11461*, 2019.

Maxim Tatarchenko, Stephan R Richter, René Ranftl, Zhuwen Li, Vladlen Koltun, and Thomas Brox. What do single-view 3d reconstruction networks learn? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3405–3414, 2019.

Lyne P Tchapmi, Vineet Kosaraju, Hamid Rezatofighi, Ian Reid, and Silvio Savarese. Topnet: Structural point cloud decoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 383–392, 2019.

Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6411–6420, 2019.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *stat*, 1050:20, 2017.

Xiaogang Wang, Marcelo H Ang Jr, and Gim Hee Lee. Cascaded refinement network for point cloud completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 790–799, 2020.

Xiaogang Wang, Marcelo H Ang, and Gim Hee Lee. Voxel-based network for shape completion by leveraging edge generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13189–13198, 2021a.

Xu Wang, Yi Jin, Yigang Cen, Tao Wang, and Yidong Li. Attention models for point clouds in deep learning: a survey. *arXiv preprint arXiv:2102.10788*, 2021b.

Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 7652–7660, 2018.

Heng Yang, Jingnan Shi, and Luca Carlone. Teaser: Fast and certifiable point cloud registration. *IEEE Transactions on Robotics*, 37(2):314–333, 2020.

Lequan Yu, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. Pu-net: Point cloud upsampling network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2790–2799, 2018.

Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network. In *2018 International Conference on 3D Vision (3DV)*, pp. 728–737. IEEE, 2018.