Fast Point Cloud Sampling Network

Tianxin Huang[a], Yong Liu *[a,*], and Jie Liang[a,b]

[a] Laboratory of Advanced Perception on Robotics and Intelligent Learning, College of Control Science and Engineering, Zhejiang University, Hangzhou, China
[b] Beijing Institute of Mechanical and Electrical Engineering, Beijing, China

Abstract

The increasing number of points in 3D point clouds has brought great challenges for subsequent algorithm efficiencies. Down-sampling algorithms are adopted to simplify the data and accelerate the computation.

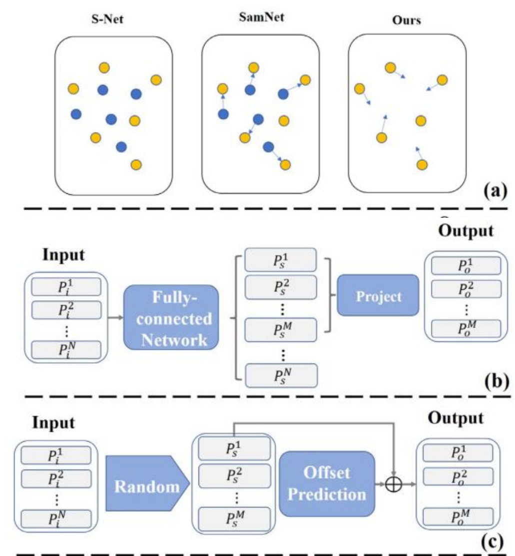

Figure 1: (a) shows the differences between learning-based sampling strategies, while (b) and (c) present the discrepancy between progress-net and our method in multiresolution sampling.

1 Introduction

Existing works [1]-[3] often use random sampling and the farthest point sampling(FPS) to down-sample the cloud points. The differences between our work and former learning-based works are presented in Fig:1 The discrepancy between progress-net and our method is presented in Fig:1-(b) and (c).

Our contributions can be summarized as:

- We propose a novel learning-based point cloud sampling framework named fast sampling network (FPN) by driving existing randomly sampled points to better positions;
- We introduce a hybrid training strategy to help FPN adapt to different sampling resolutions by randomly introducing selecting the resolution of initial points during training;

2 Methodology

2.1 Basic Pipeline

The basic pipeline of FPN is presented in Fig. 2. We aggregate global features from the input points with a set of multilevel perceptions (MLPs) and Max Pooling following PointNet [3].

*Corresponding Author

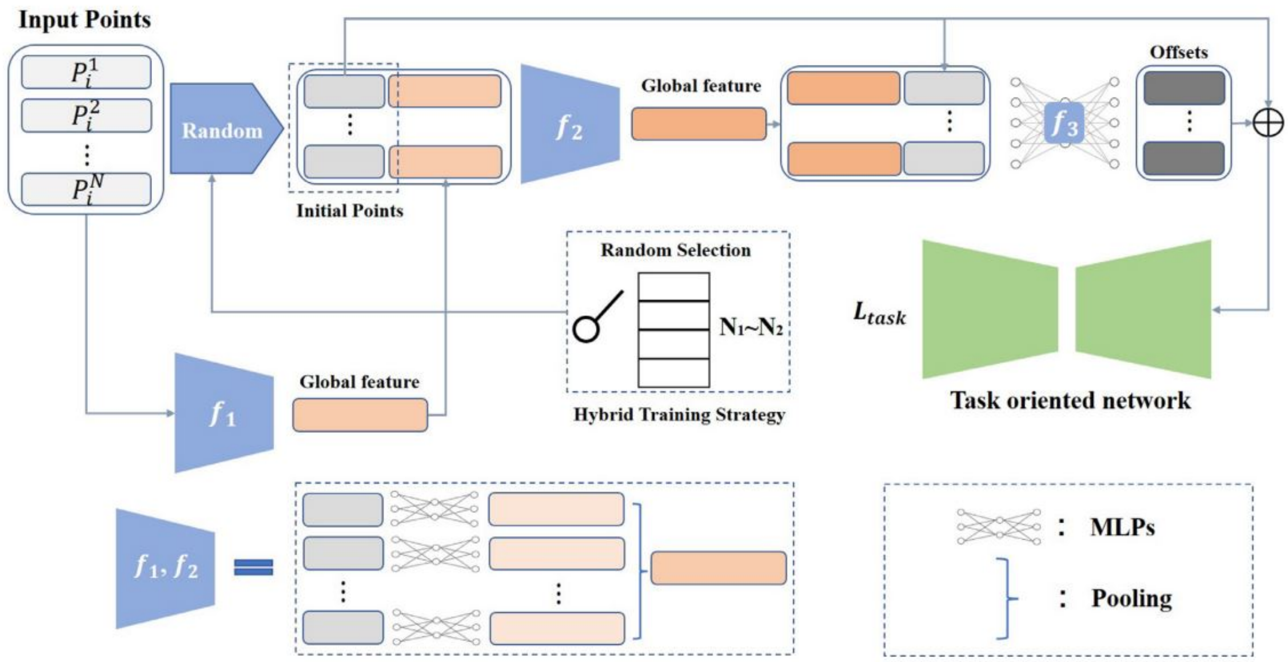

Figure 2: The whole pipeline of FPN. The + denotes element-wised addition. f1 and f2 aggregate features by MultiLayer Perceptrons(MLPs) and pooling

2.2 Hybrid Training Strategy

The achievement of HTS is presented as Algorithm 1.

2.3 Loss function

The range constraint can be presented as:

$$\mathcal{L}_{rc} = \frac{1}{N} \sum \| S_0 - S_i \|_2 \tag{1}$$

For reconstruction related tasks, it may be Chamfer Distance or Earth Mover Distance [1] defined as:

$$\mathcal{L}_{task} = \mathcal{L}_C D(S_1, S_2) = \frac{1}{2}(\frac{1}{\mid S_1 \mid} \sum_{x \in S_1} min_{y \in S_2} \| x - y \|_2 + \frac{1}{\mid S_2 \mid} \sum_{x \in S_2} min_{y \in S_1} \| x - y \|_2) \tag{2}$$

or

$$\mathcal{L}_{task} = \min_{\phi} : S_1 \longrightarrow S_2 \frac{1}{|S_1|} \sum_{x \in S_1} \| x - \phi(x) \|_2 \tag{3}$$

where S1 and S2 are input and output.ϕ is a bijection from S1 to S2.

3 Experiments

Table1

The number of neurons in networks. f_1, f_2, f_3 are modules in Fig. 2.

	f_1	f_2	f_3
MLPs	(128,256,256)	(128,256,256)	(128,128,3)

Table2

The comparison on optimal clustering.

Center	Iterations	1	10	100
16	FPS	2.43	2.00	1.98
	Ours	**2.16**	**1.98**	**1.96**
32	FPS	1.20	1.02	1.00
	Ours	**1.11**	**1.00**	**1.00**

VIDIA 2080ti GPU with a 2.9GHZ i5-9400 CPU based on Tensorflow. The hyper-parameter is tuned on the validation split of ShapeNet. Detailed network structures are shown in Table 3.

3.1 Discussion about clustering

Except down-stream tasks such as reconstruction or recognition, down-sampled points can also be adopted as the initial clustering centers.
The results are presented in Table 3.

Algorithm 1 Training with Hybrid Training Strategy

Input: data X, the number of iterations iter, the number of resolutions m;
for i=1 **to** iter **do**
Select the resolution r $prob_1, ...prob_m$
Train FPN by descending gradient: $\delta_{\theta_{VPN}} \mathcal{L}_{loss}(Y_{X,r})$
end for

4 The influence of range constraint

The sampling efficiency is important for real-world applications. In this section, we compare the time efficiency of different. sampling strategies under different resolutions between 64-1024. Though random sampling is a little faster than our method, it often gets the worst results as shown in Fig. 3 and FPN outperforms commonly used sampling strategies such as FPS and SNet on task performances, while [2] it is only slower than the random sampling.

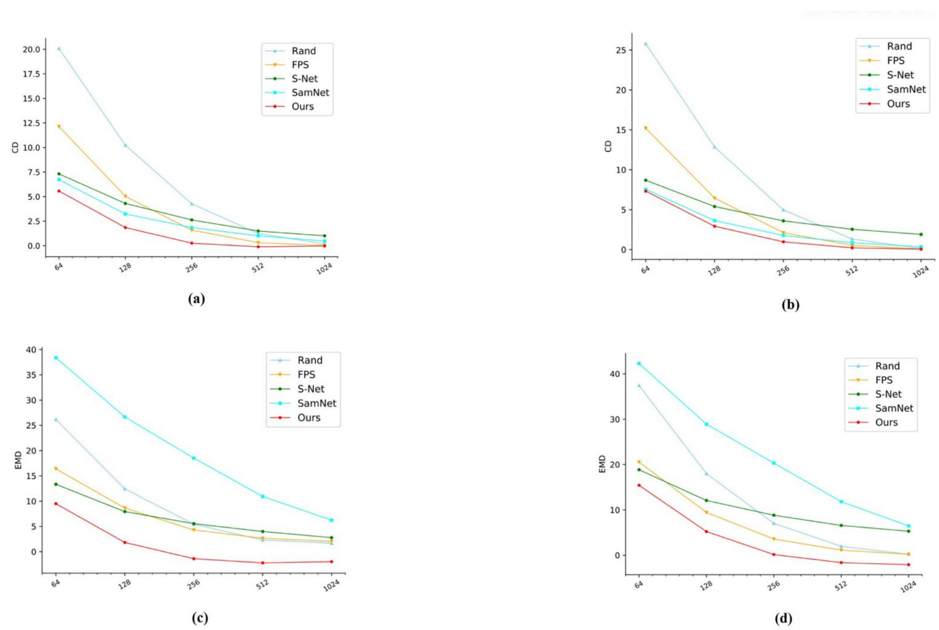

Figure 3: Reconstruction comparisons between sampling strategies. (a) and (b) denote errors evaluated on ModelNet10 and ModelNet40 for CD-based reconstruction networks, while (c) and (d) show performances of EMD-based reconstruction networks on ModelNet10 and ModelNet40.

Acknowledgement

We thank all reviewers and the editor for excellent contributions. This work is supported by the Key Research and Development Project of Zhejiang Province under Grant 2021C01035.

References

[1] Itai Lang, Asaf Manor, and Shai Avidan. "Samplenet: Differentiable point cloud sampling". In: (2020), pp. 7578–7588.

[2] Charles R Qi et al. "Deep hough voting for 3d object detection in point clouds". In: (2019), pp. 9277–9286.

[3] Charles R Qi et al. "Pointnet: Deep learning on point sets for 3d classification and segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 652–660.