Spherical Frustum Sparse Convolution Network for LiDAR Point Cloud Semantic Segmentation

Yu Zheng, Shanghai Jiao Tong University

Guangming Wang, University of Cambridge

Jiuming Liu, Shanghai Jiao Tong University

Marc Pollefeys, ETH Zürich

Hesheng Wang， Shanghai Jiao Tong University

1 Introduction

Related Work

Point-Based Semantic Segmentation

PointNet PointNet++

RandLA KPConv

- High Computational Complexity of Neighbor Querying
- Non-efficient

2D Projection-Based Semantic Segmentation

SqueezeSeg RangeNet++

SqueezeSegV3 FIDNet

CENet RangeViT

- Efficient
- Quantized Information Loss During Projection

Preserve the Efficiency using 2D Projection-Based Processing & Overcome the Quantized Information Loss

2 Idea & Intuition

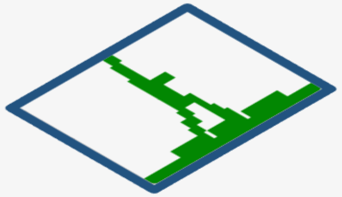
Spherical Frustum V.S. Spherical projection

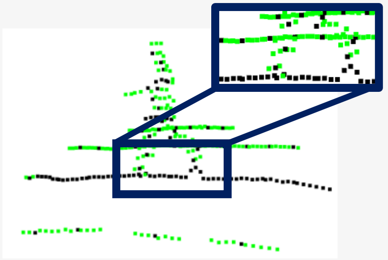
Challenge:

2D Projection-Based Point Cloud Semantic Segmentation

Multiple Points Projected onto the Same 2D Position

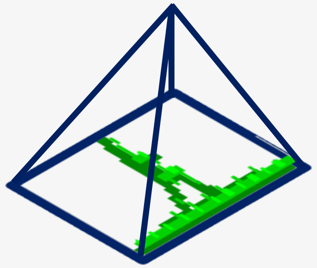
Spherical Projection → **Only Keep the Closest Point & Drop the others**

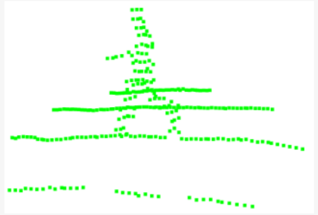




Dropped
Partial Information

Spherical Frustum (Ours) → **Keep All Points in Spherical Frustum**





All Preserved
Complete Information

✓ **Overcome Quantized Information Loss**

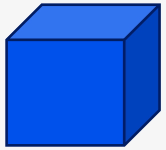
✓ **Preserve the Complete Geometric Structure of the Raw Point Cloud**

3

Method

Hash-Based Representation of Spherical Frustum

Every Points in Spherical Frustum can be represented as a 3-tuple coordinates (u_i, v_i, m_i) → **3-D Dense Grids**



Realize Efficient Neighbor Querying (Array Querying)

Drawback

- Pad the every spherical frustum to maximal point number
- Huge redundant memory usage.

Naive

Ours

Memory-efficient Representation of Spherical Frustum → Use Hash Table to represent neighbor relation & Save the point cloud in original irregular Point Set

Efficient Neighboring Query → Use Hash Table to efficiently map the 3-tuple coordinates to the index in original Point Set

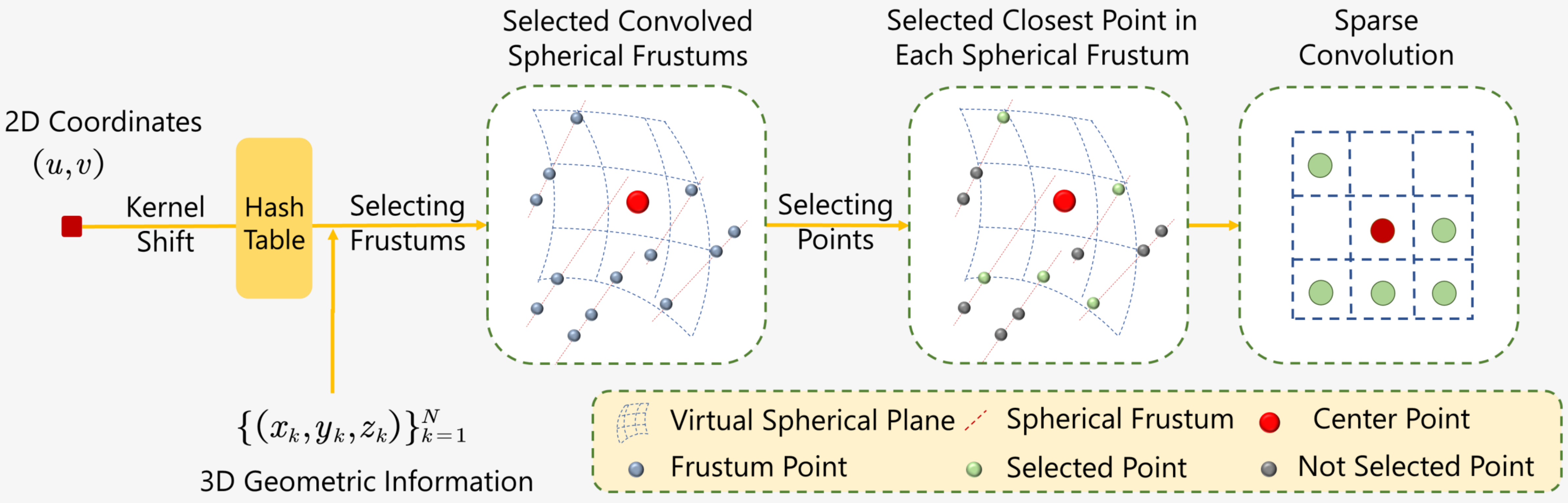
Query Index $\{(u_i, v_i, m_i)\}_{i=1}^{M}$ → Hash Table $(u_i, v_i, m_i) \rightarrow i$

Spherical Frustum sparse Convolution (SFC)

Target:

Implementing 2D Convolution for multiple points on one 2D Position.

- **Select the Closest Point towards the Center Point in Each Frustum.**
- **Project it onto the Visual Spherical Plane of the Center Point.**
- **Perform 2D Convolution on the Visual Spherical Plane.**



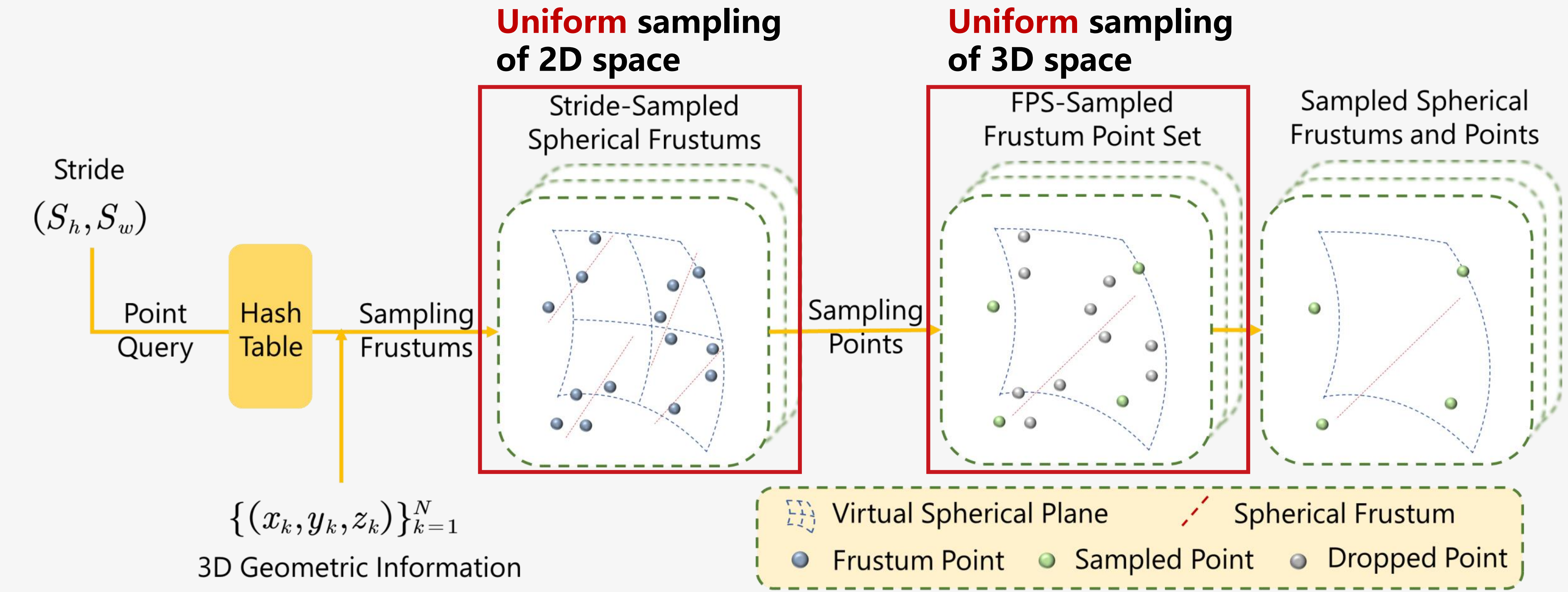
Frustum Farthest Point Sampling (F2PS)

Target:

Efficient & Uniform Sampling of Point Cloud with Spherical Frustum Representation.

- **Stride-based frustum sampling**
- **Farthest point sampling of the frustum points**

Point Limited and Efficient



4 Experiments

Experiment Results

Results on SemanticKITTI Dataset:

Approach	**mIoU (%)**	car	bicycle	motorcycle	truck	other-vehicle	person	bicyclist	motorcyclist	road	parking	sidewalk	other-ground	building	fence	vegetation	trunk	terrain	pole	traffic-sign
RangeNet++ [2]	52.2	91.4	25.7	34.4	25.7	23.0	38.3	38.8	4.8	91.8	65.0	75.2	27.8	87.4	58.6	80.5	55.1	64.6	47.9	55.9
PolarNet [3]	54.3	93.8	40.3	30.1	22.9	28.5	43.2	40.2	5.6	90.8	61.7	74.4	21.7	90.0	61.3	84.0	65.5	67.8	51.8	57.5
SqueezeSegV3 [4]	55.9	92.5	38.7	36.5	29.6	33.0	45.6	46.2	20.1	91.7	63.4	74.8	26.4	89.0	59.4	82.0	58.7	65.4	49.6	58.9
SalsaNext [5]	59.5	91.9	48.3	38.6	38.9	31.9	60.2	59.0	19.4	91.7	63.7	75.8	29.1	90.2	64.2	81.8	63.6	66.5	54.3	62.1
KPRNet [6]	63.1	**95.5**	54.1	47.9	23.6	42.6	65.9	65.0	16.5	**93.2**	**73.9**	**80.6**	30.2	91.7	68.4	**85.7**	69.8	**71.2**	58.7	64.1
Lite-HDSeg [7]	63.8	92.3	40.0	55.4	37.7	39.6	59.2	71.6	**54.1**	93.0	68.2	78.3	29.3	91.5	65.0	78.2	65.8	65.1	59.5	**67.7**
RangeViT [8]	64.0	95.4	55.8	43.5	29.8	42.1	63.9	58.2	38.1	93.1	70.2	80.0	**32.5**	**92.0**	**69.0**	85.3	70.6	**71.2**	60.8	64.7
CENet [9]	64.7	91.9	58.6	50.3	**40.6**	42.3	68.9	65.9	43.5	90.3	60.9	75.1	31.5	91.0	66.2	84.5	69.7	70.0	61.5	67.6
SFCNet (Ours)	**65.0**	95.1	**64.2**	**63.2**	23.5	**45.6**	**78.3**	**73.1**	26.4	87.9	65.6	71.9	29.1	91.1	64.5	83.7	**72.6**	69.6	**62.6**	67.2

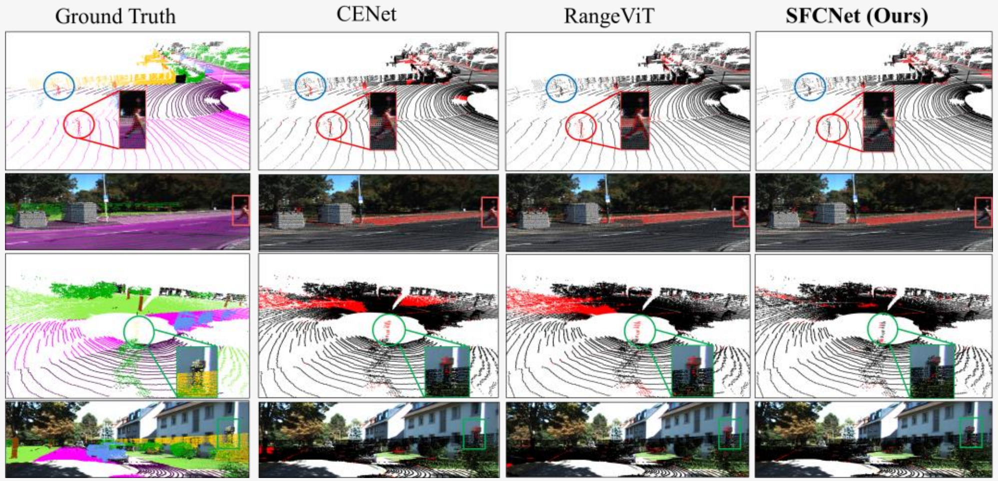
Exceed existing 2D projection methods, with a significant performance improvement on small objects.

Results on nuScenes Dataset :

Approach	**mIoU (%)**	barrier	bicycle	bus	car	construction	motorcycle	pedestrian	traffic-cone	trailer	truck	driveable	other flat	sidewalk	terrain	manmade	vegetation
RangeNet++ [2]	65.5	66.0	21.3	77.2	80.9	30.2	66.8	69.6	52.1	54.2	72.3	94.1	66.6	63.5	70.1	83.1	79.8
PolarNet [3]	71.0	74.7	28.2	85.3	90.9	35.1	77.5	71.3	58.8	57.4	76.1	96.5	71.1	**74.7**	74.0	87.3	85.7
SalsaNext [5]	72.2	74.8	34.1	85.9	88.4	42.2	72.4	72.2	63.1	61.3	76.5	96.0	70.8	71.2	71.5	86.7	84.4
RangeViT [8]	75.2	75.5	**40.7**	88.3	90.1	**49.3**	79.3	77.2	**66.3**	65.2	80.0	96.4	71.4	73.8	73.8	**89.9**	87.2
SFCNet (Ours)	**75.9**	**76.7**	40.4	**89.5**	**91.3**	46.7	**82.0**	**78.1**	65.8	**69.4**	**80.6**	**96.6**	**71.6**	74.5	**74.9**	89.0	**87.5**

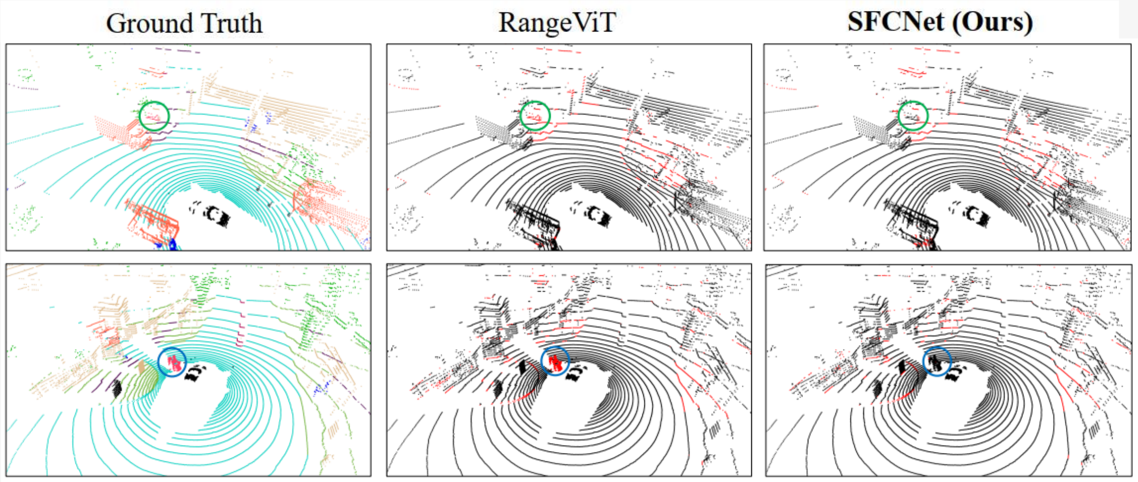
Surpass existing 2D projection methods on both two datasets.

Visualization on Small Object Segmentation on SemanticKITTI:



Our model performs better segmentation for small objects across various scenes, such as intersections and urban streets!

Visualization on Small Object Segmentation on nuScenes:



5 Conclusion & Future Work

Conclusion & Future Work

Summary

SFCNet overcomes the quantized information loss and **enhances the performance** of 2D projection-based Point Cloud Semantic Segmentation.

- **Spherical Frustum** Structure **overcoming** Quantized information loss
- **Memory-Efficient** Hash-Based Representation of Spherical Frustum
- **Efficient** Spherical Frustum sparse **Convolution** & Frustum Farthest Point **Sampling**
- **Code** will be released at https://github.com/IRMVLab/SFCNet.

Future Work

- Expanding Receptive Field by combining Spherical Frustum with Transformer and Mamba
- Application on multi-modal fusion-based Point Cloud Semantic Segmentation
- Application on Point Cloud Registration & Scene Flow Estimation

…