JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

Poster tag

THU-PM-115

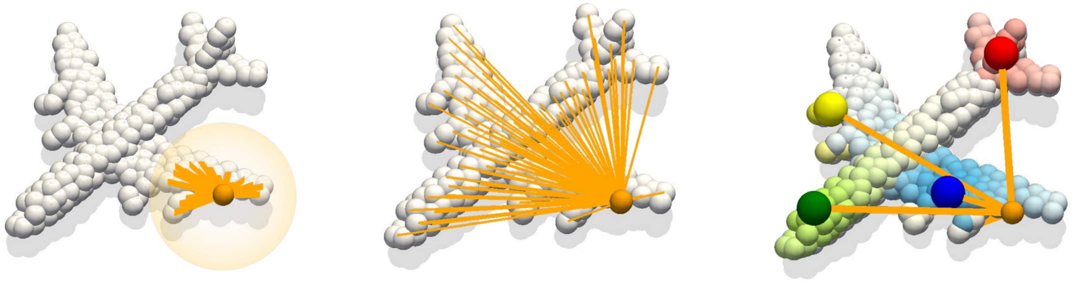
Self-positioning Point-based Transformer for Point Cloud Understanding

Jinyoung Park[1*], Sanghyeok Lee[1*], Sihyeon Kim[1], Yunyang Xiong[2], Hyunwoo J. Kim[1†]

[1]Korea University, [2]Meta Reality Labs

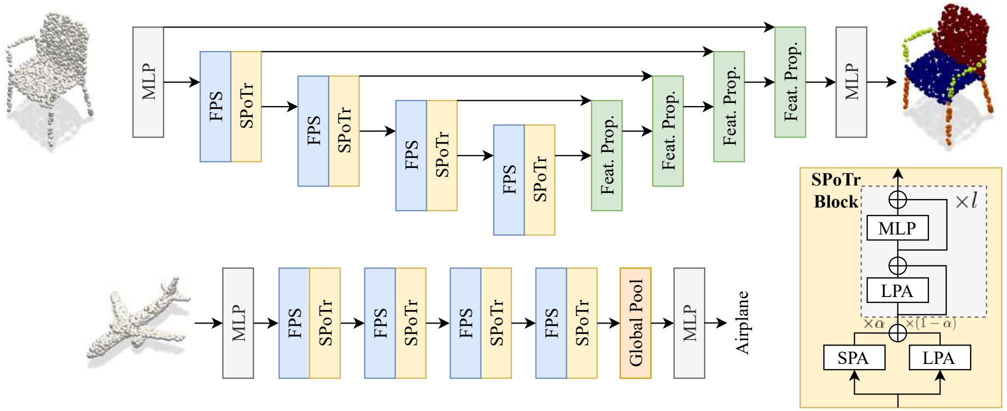
Preview

Motivation

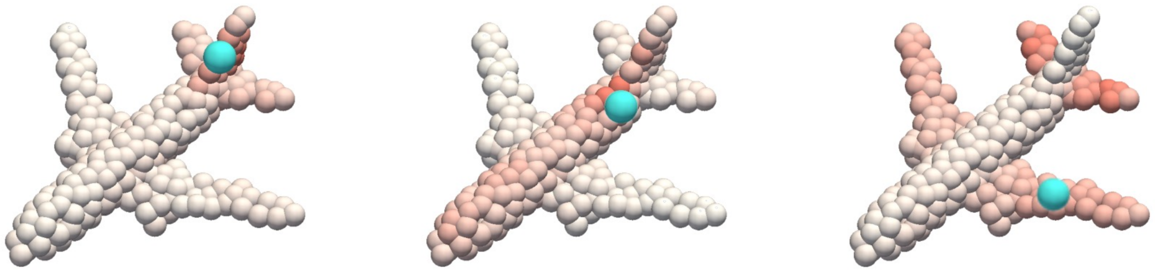


Local attention | Global attention | SP attention (Ours)

SPoTr



SP attention



SP attention according to SP point

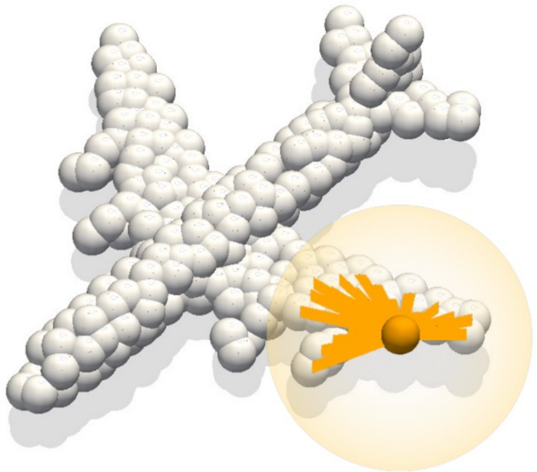
Experiments

Shape Classification: 88.6 OA

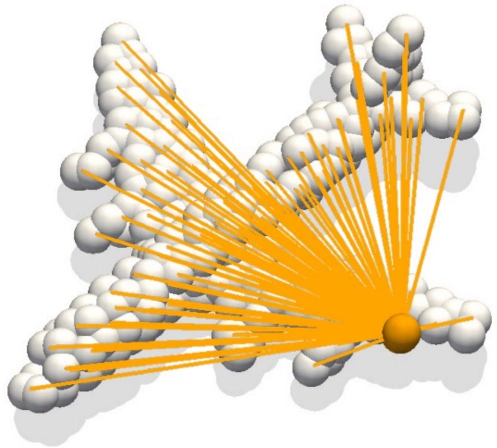
Part Segmentation: 87.2 Ins. mIOU

Scene Segmentation: 70.8 mIOU

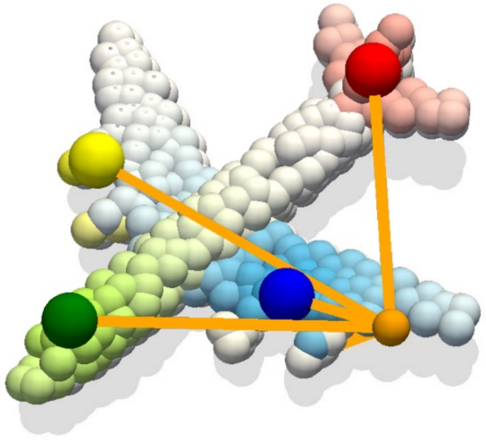
Attention-based methods for point cloud understanding



Local attention



Global attention



SP attention

Local attention cannot capture global shape context.

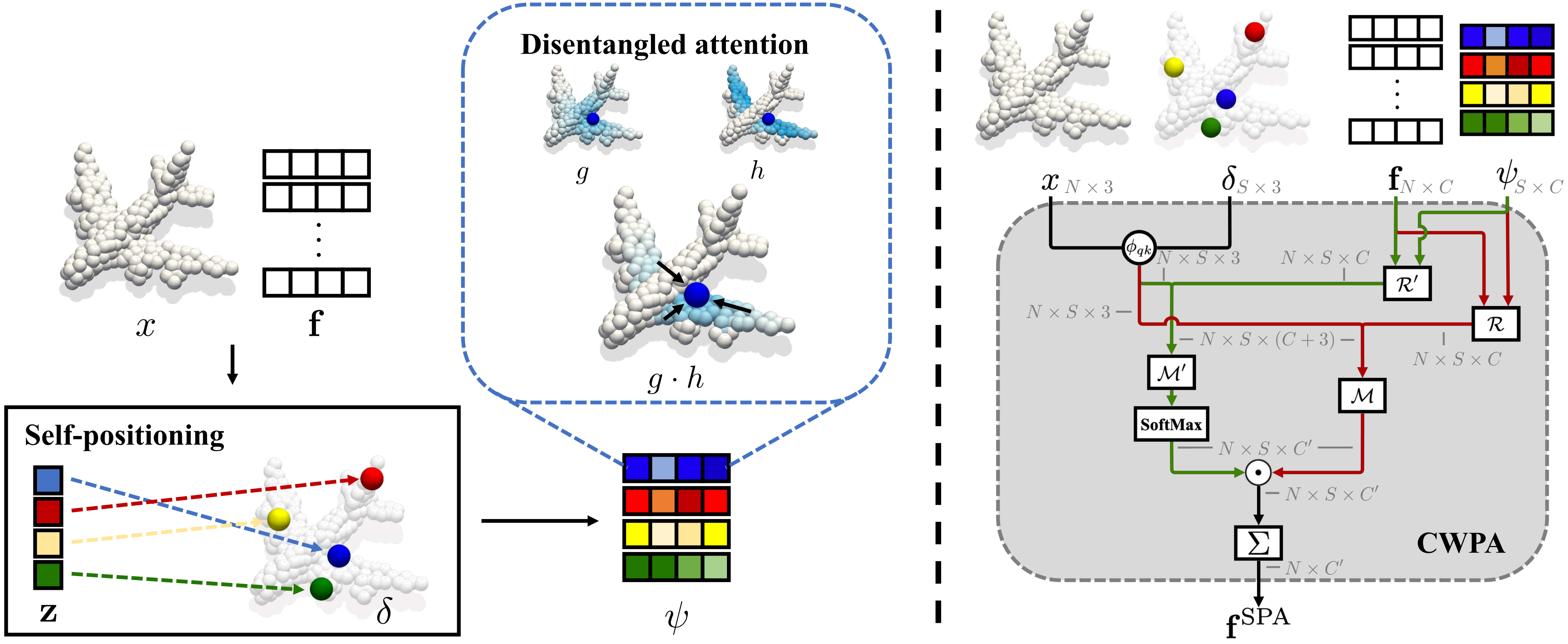
Global attention requires heavy computational cost.

SP attention capture both local and global shape contexts with reduced complexity.

[1] Zhao, Hengshuang, et al. "Point transformer." *Proceedings of the IEEE/CVF international conference on computer vision*. 2021.

[2] Yan, Xu, et al. "Pointasnl: Robust point clouds processing using nonlocal neural networks with adaptive sampling." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020.

Self-positioning point-based attention



Channel-wise point attention

CWPA

$$\text{CWPA}\left(x_q, \mathbf{f}_q, \{x_k\}_{k \in \Omega_{\text{key}}}, \{\mathbf{f}_k\}_{k \in \Omega_{\text{key}}}\right)$$

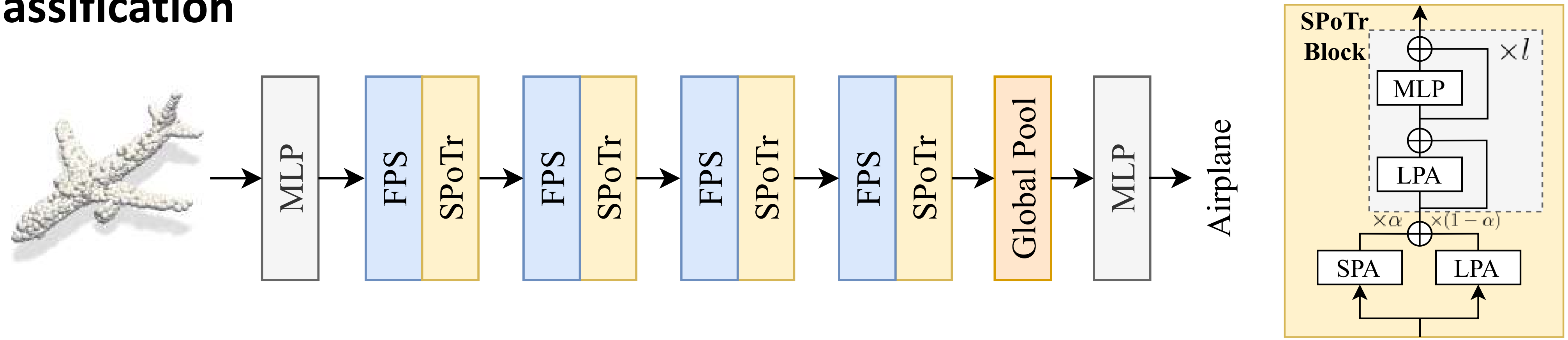
$$= \sum_{k \in \Omega_{\text{key}}} \mathbb{A}_{q,k,:} \odot \mathcal{M}\left([\mathcal{R}(\mathbf{f}_q, \mathbf{f}_k); \phi_{qk}]\right),$$

Attention

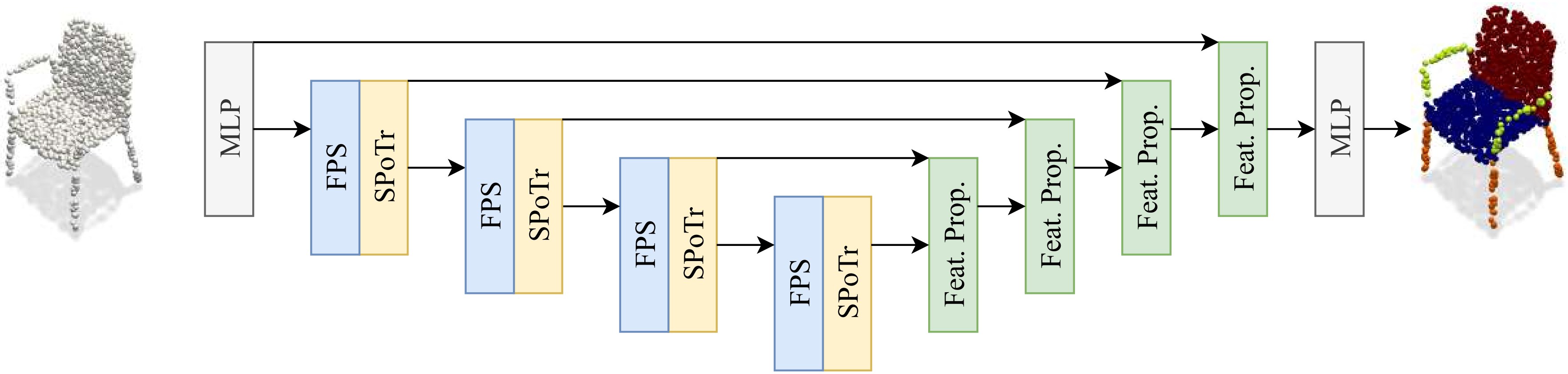
$$\mathbb{A}_{q,k,c} = \frac{\exp\left(\mathcal{M}'\left([\mathcal{R}'(\mathbf{f}_q, \mathbf{f}_k); \phi_{qk}]/\tau\right)_c\right)}{\sum_{k' \in \Omega_{key}} \exp\left(\mathcal{M}'\left([\mathcal{R}'(\mathbf{f}_q, \mathbf{f}_{k'}); \phi_{qk'}]/\tau\right)_c\right)},$$

Self-positioning point-based Transformer

Classification



Segmentation



Experiments

> Datasets

- Shape classification
  - ScanObjectNN (SONN)
- Part segmentation
  - SN-Part
- Scene segmentation
  - S3DIS

Experiments

> Experimental results

Methods	Year	mAcc	OA
PointNet [36]	2017	63.4	68.2
PointNet++ [27]	2017	75.4	77.9
SpiderCNN [9]	2018	69.8	73.7
PointCNN [7]	2018	75.1	78.5
DGCNN [53]	2019	73.6	78.1
DRNet [54]	2021	78.0	80.3
GBNet [55]	2021	77.8	80.5
SimpleView [33]	2021	-	80.5
PRA-Net [56]	2021	77.9	81.0
MVTN [34]	2021	-	82.8
CT [48]	2021	83.1	85.5
PointMLP [39]	2022	84.4	85.7
RepSurf-U [40]	2022	83.1	86.0
PointNeXt [43]	2022	85.8±0.6	87.7±0.4
SPoTr	2023	**86.8**	**88.6**

Table 1. **Shape classification results on PB_T50_RS in SONN.** mAcc is the mean of class accuracy and OA is the overall accuracy.

Methods	Year	cls. mIoU	ins. mIoU
PointNet [36]	2017	80.4	83.7
PointNet++ [27]	2017	81.9	85.1
PointCNN [7]	2018	84.6	86.1
DGCNN [53]	2019	82.3	85.1
RSCNN [5]	2019	84.0	86.2
KPConv [6]	2019	85.1	86.4
PointConv [10]	2019	82.8	85.7
PointASNL [26]	2020	-	86.1
PCT [46]	2021	-	86.4
PAConv [11]	2021	84.6	86.1
AdaptConv [38]	2021	83.4	86.4
PointTransformer [25]	2021	83.7	86.6
CurveNet [50]	2021	-	86.8
PointMLP [39]	2022	84.6	86.1
PointNeXt [43]	2022	85.2 ± 0.1	87.0 ± 0.1
SPoTr	2023	**85.4**	**87.2**

Table 2. **Part segmentation results on SN-Part.** ins. mIoU is the mean of instance IoU. cls. mIoU is the mean of the class IoU.

Methods	Year	OA	mAcc	mIoU
PointNet [36]	2017	-	-	41.1
PointCNN [7]	2018	85.9	63.9	57.3
PointWeb [59]	2019	87.0	66.6	60.3
KPConv [6]	2019	-	72.8	67.1
PCT [46]	2021	-	67.7	61.3
CT [48]	2021	-	-	67.9
PointTransformer [25]	2021	**90.8**	-	70.4
RepSurf-U [40]	2022	90.2	76.0	68.9
PointNeXt [43]	2022	90.6 ± 0.1	-	70.5 ± 0.3
SPoTr	2023	90.7	**76.4**	**70.8**

Table 3. **Semantic segmentation results on S3DIS.** OA is the overall accuracy, mAcc is the mean of class accuracy, and mIoU is the mean of instance IoU.

Quantitative analysis

> Ablation study

Method	g	h	SP	OA
w/o SPA (*baseline*)				87.9
w/o self-positioning	✓	✓		87.7
w/o disentangled attention	✓		✓	88.2
SPoTr (*ours*)	✓	✓	✓	**88.6**

Table 4. **Ablations on SONN_PB.** g: spatial kernel, h: semantic kernel, SP: self-positioning points. OA is the overall accuracy.

Quantitative analysis

> Efficiency

Method	Param ↓ (M)	FLOPs ↓ (G)	Memory ↓ (GB)	Throughput ↑ (shapes/s)
GSA	**1.7**	114.0	24.2	17.7
SPA (*ours*)	**1.7**	**10.8**	**2.5**	**281.5**
	(-)	(- 90.5%)	(- 89.7%)	(× 15.9)

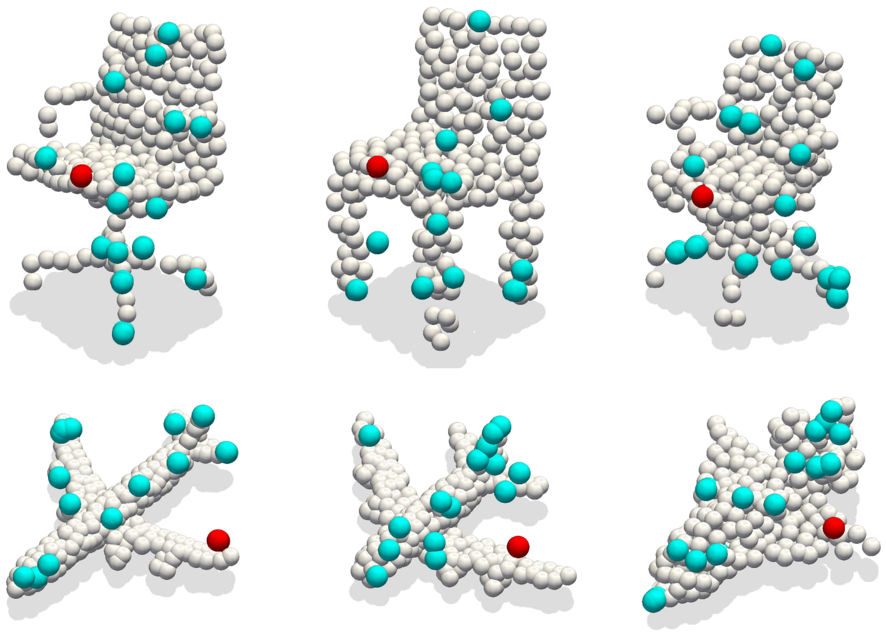
Table 6. **Complexity analysis on SN-Part.** SPA: self-positioning point-based attention, GSA: global self-attention.

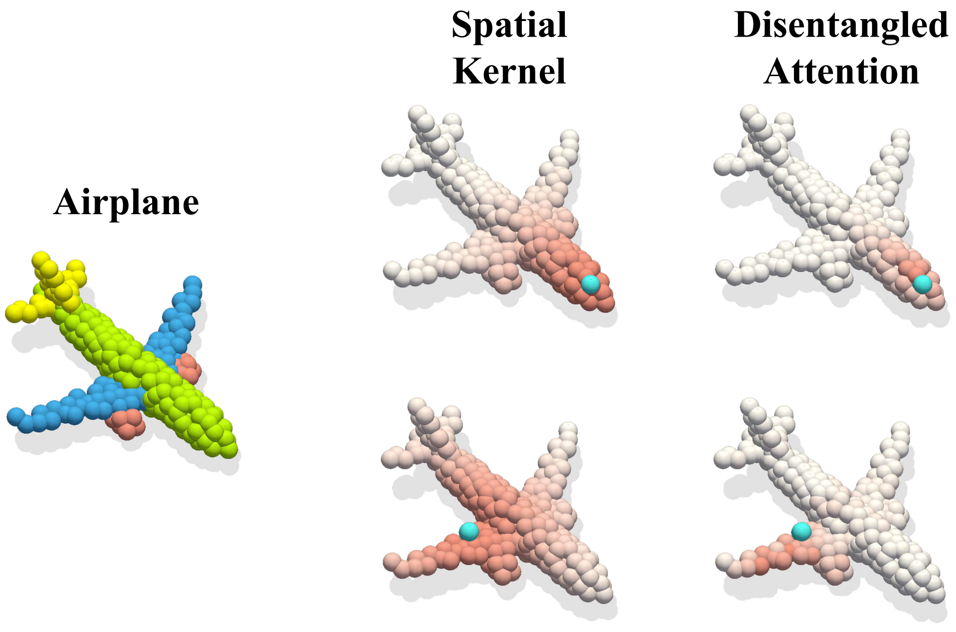
SONN	OA ↑	Param ↓ (M)	FLOPs ↓ (G)
PointMLP [20]	85.7	13.2	31.4
RepSurf [24]	86.0	6.8	4.9
SPoTr*	88.2	1.6	5.5
SPoTr	**88.6**	3.3	12.3

Table 7. **Efficiency comparison on SONN.** SPoTr* is a light version of SPoTr

Qualitative analysis

> Self-positioning points and disentangled attention





- SP points are adaptively located considering the input shape.

- SP attention aggregates feature considering both spatial proximity and semantic proximity via disentangled attention.

Conclusion

- We design a novel Transformer architecture (SPoTr) to tackle the long-range dependency issues and the scalability issue of Transformer for point clouds.
- We propose a global cross-attention mechanism with flexible self-positioning points (SPA). SPA aggregates information on a few self-positioning points via disentangled attention and non-locally distributes information to semantically related points.
- SPoTr achieves the best performance on three point cloud benchmark datasets (SONN, SN-Part, and S3DIS) against strong baselines.