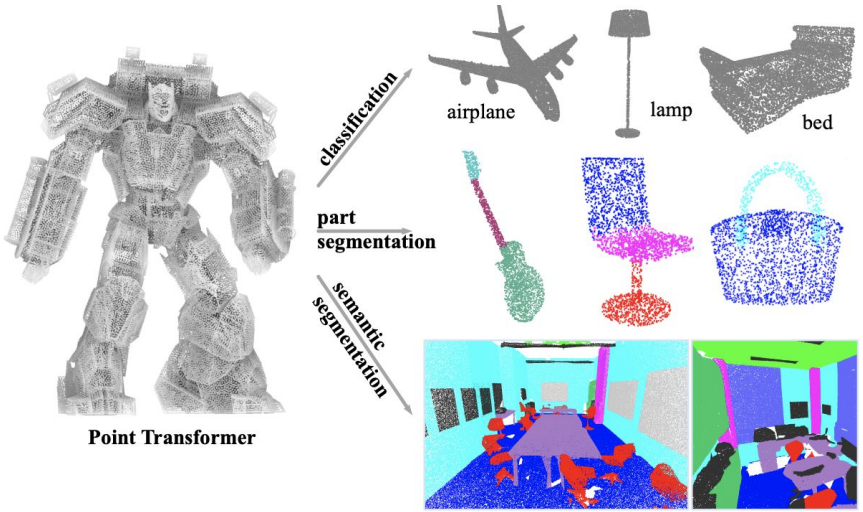
Point Transformer



Authors: Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, Vladlen Koltun

Presented by: Connor Vines, Daniel Zhang, Han Lin

3D Point Clouds

- Application areas:
  - Autonomous driving, augmented reality, and robotics

- Structural difference with images:
  - Unlike images, which are arranged on regular pixel grids, 3D point clouds are sets embedded in continuous space
  - This precludes immediate application of deep network designs that have become standard in computer vision, such as networks based on CNNs.

Previous Work

- Projection Based Networks
  - Project 3D point clouds into various image planes, then use 2D CNNs to extract features
  - Drawbacks:
    - The geometric information inside point clouds is collapsed due to projection
    - Underutilize the sparsity of point clouds when forming dense pixel grids on projection planes

- Voxel-Based Networks
  - Transform to regular representations is 3D voxelization, followed by convolutions in 3D
  - Drawbacks: Incur massive computation/memory costs

- Point-Based Networks
  - Deep network structures that ingest point clouds directly: PointNet, PointNet++
  - Connect the point set into a graph and conduct message passing on this graph: DGCNN, ECC

In This Work

- We design a Point Transformer layer for point cloud processing, which is invariant to permutation and cardinality.

- We construct Point Transformer networks for classification and dense prediction on point clouds, which can serve as general backbones for 3D scene understanding.

- We set the new SOTA on multiple highly competitive benchmarks, outperforming long lines of prior work.

Methodology

Self-Attention

Scalar Self Attention

$$\mathbf{y}_i = \sum_{\mathbf{x}_j \in \mathcal{X}} \rho\big(\varphi(\mathbf{x}_i)^\top \psi(\mathbf{x}_j) + \delta\big)\alpha(\mathbf{x}_j),$$

Vector Self Attention

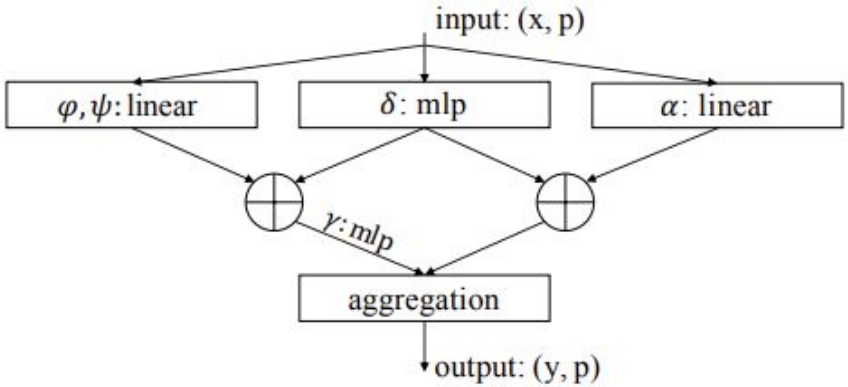
$$\mathbf{y}_i = \sum_{\mathbf{x}_j \in \mathcal{X}} \rho\big(\gamma(\beta(\varphi(\mathbf{x}_i), \psi(\mathbf{x}_j)) + \delta)\big) \odot \alpha(\mathbf{x}_j),$$

Point Transformer Layer

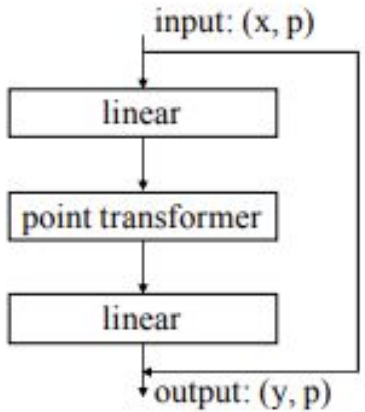
Point Transformer Layer

$$\mathbf{y}_i = \sum_{\mathbf{x}_j \in \mathcal{X}(i)} \rho\big(\gamma(\varphi(\mathbf{x}_i) - \psi(\mathbf{x}_j) + \delta)\big) \odot \big(\alpha(\mathbf{x}_j) + \delta\big)$$

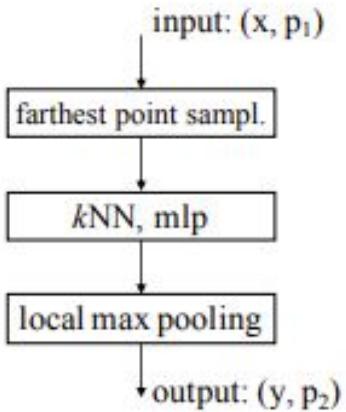
$$\delta = \theta(\mathbf{p}_i - \mathbf{p}_j).$$



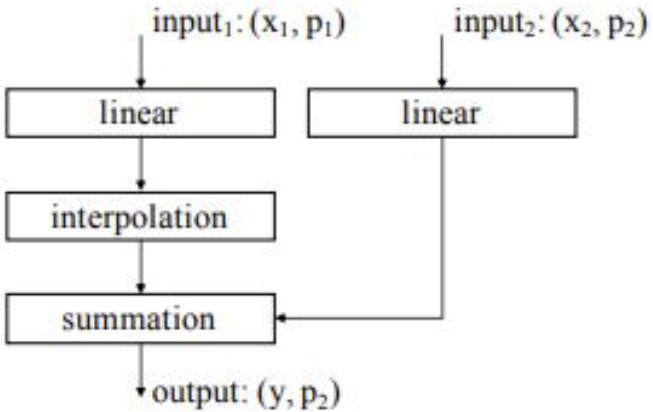
Module Design



(a) point transformer block



(b) transition down



(c) transition up

Architecture

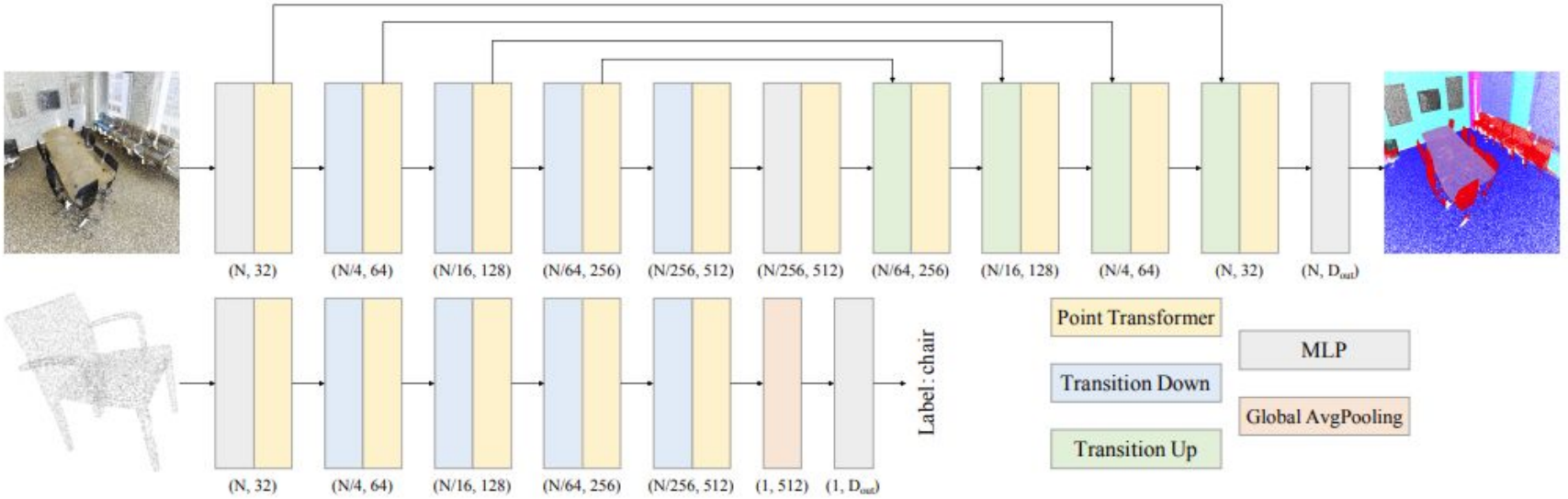


Figure 3. Point transformer networks for semantic segmentation (top) and classification (bottom).

Results

Semantic Segmentation

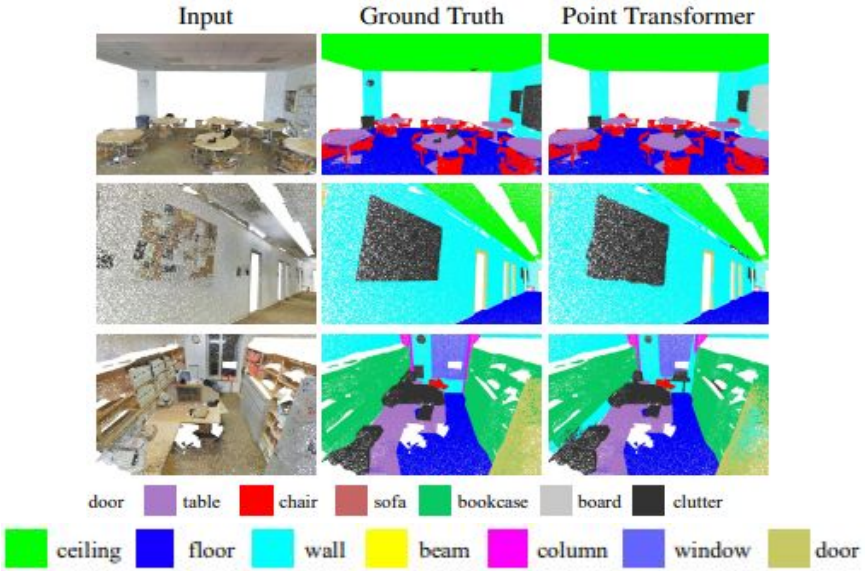
- S3DIS
- Outperforms all prior work across OA, mAcc, mIoU

Method	OA	mAcc	mIoU	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
PointNet [25]	–	49.0	41.1	88.8	97.3	69.8	0.1	3.9	46.3	10.8	59.0	52.6	5.9	40.3	26.4	33.2
SegCloud [36]	–	57.4	48.9	90.1	96.1	69.9	0.0	18.4	38.4	23.1	70.4	75.9	40.9	58.4	13.0	41.6
TangentConv [35]	–	62.2	52.6	90.5	97.7	74.0	0.0	20.7	39.0	31.3	77.5	69.4	57.3	38.5	48.8	39.8
PointCNN [20]	85.9	63.9	57.3	92.3	98.2	79.4	0.0	17.6	22.8	62.1	74.4	80.6	31.7	66.7	62.1	56.7
SPGraph [15]	86.4	66.5	58.0	89.4	96.9	78.1	0.0	42.8	48.9	61.6	84.7	75.4	69.8	52.6	2.1	52.2
PCCN [42]	–	67.0	58.3	92.3	96.2	75.9	0.3	6.0	69.5	63.5	66.9	65.6	47.3	68.9	59.1	46.2
PAT [50]	–	70.8	60.1	93.0	98.5	72.3	1.0	41.5	85.1	38.2	57.7	83.6	48.1	67.0	61.3	33.6
PointWeb [55]	87.0	66.6	60.3	92.0	98.5	79.4	0.0	21.1	59.7	34.8	76.3	88.3	46.9	69.3	64.9	52.5
HPEIN [13]	87.2	68.3	61.9	91.5	98.2	81.4	0.0	23.3	65.3	40.0	75.5	87.7	58.5	67.8	65.6	49.4
MinkowskiNet [37]	–	71.7	65.4	91.8	98.7	86.2	0.0	34.1	48.9	62.4	81.6	89.8	47.2	74.9	74.4	58.6
KPConv [37]	–	72.8	67.1	92.8	97.3	82.4	0.0	23.9	58.0	69.0	81.5	91.0	75.4	75.3	66.7	58.9
PointTransformer	**90.8**	**76.5**	**70.4**	94.0	98.5	86.3	0.0	38.0	63.4	74.3	89.1	82.4	74.3	80.2	76.0	59.3

Table 1. Semantic segmentation results on the S3DIS dataset, evaluated on Area 5.

Semantic Segmentation

- Outperforms across the board
- Several times fewer parameters



Method	OA	mAcc	mIoU
PointNet [25]	78.5	66.2	47.6
RSNet [12]	–	66.5	56.5
SPGraph [15]	85.5	73.0	62.1
PAT [50]	–	76.5	64.3
PointCNN [20]	88.1	75.6	65.4
PointWeb [55]	87.3	76.2	66.7
ShellNet [53]	87.1	–	66.8
RandLA-Net [37]	88.0	82.0	70.0
KPConv [37]	–	79.1	70.6
PointTransformer	**90.2**	**81.9**	**73.5**

Table 2. Semantic segmentation results on the S3DIS dataset, evaluated with 6-fold cross-validation.

Shape Classification

- ModelNet40
- Outperforms all prior work in both mAcc and OA

Method	input	mAcc	OA
3DShapeNets [47]	voxel	77.3	84.7
VoxNet [23]	voxel	83.0	85.9
Subvolume [26]	voxel	86.0	89.2
MVCNN [34]	image	–	90.1
PointNet [25]	point	86.2	89.2
A-SCN [48]	point	87.6	90.0
Set Transformer [17]	point	–	90.4
PAT [50]	point	–	91.7
PointNet++ [27]	point	–	91.9
SpecGCN [40]	point	–	92.1
PointCNN [20]	point	88.1	92.2
DGCNN [44]	point	90.2	92.2
PointWeb [55]	point	89.4	92.3
SpiderCNN [49]	point	–	92.4
PointConv [46]	point	–	92.5
Point2Sequence [21]	point	90.4	92.6
KPConv [37]	point	–	92.9
InterpCNN [22]	point	–	93.0
PointTransformer	point	**90.6**	**93.7**

Table 3. Shape classification results on the ModelNet40 dataset.

Shape Classification

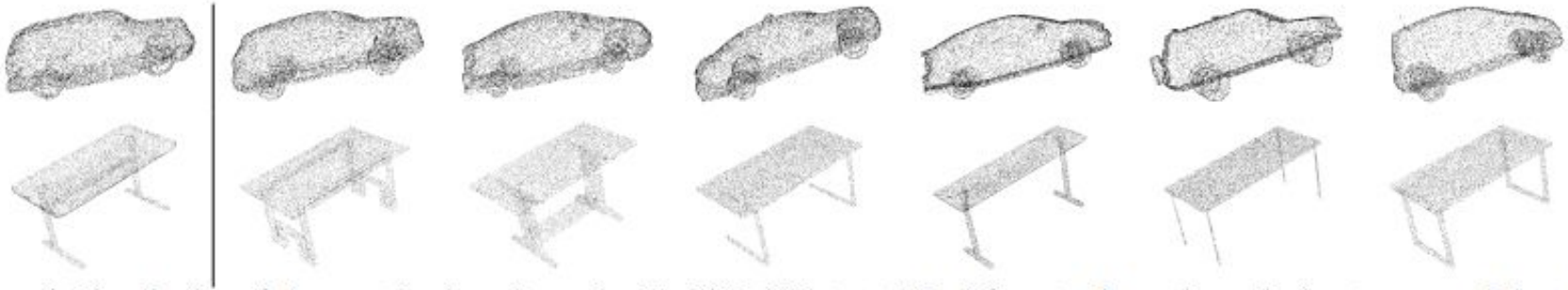


Figure 6. Visualization of shape retrieval results on the ModelNet40 dataset. The leftmost column shows the input query and the other columns show the retrieved models.

Object Part Segmentation

- ShapeNetPart
- SOTA instance mIoU
- Competitive category mIoU

Method	cat. mIoU	ins. mIoU
PointNet [25]	80.4	83.7
A-SCN [48]	–	84.6
PCNN [42]	81.8	85.1
PointNet++ [27]	81.9	85.1
DGCNN [44]	82.3	85.1
Point2Sequence [21]	–	85.2
SpiderCNN [49]	81.7	85.3
SPLATNet [33]	83.7	85.4
PointConv [46]	82.8	85.7
SGPN [43]	82.8	85.8
PointCNN [20]	84.6	86.1
InterpCNN [22]	84.0	86.3
KPConv [37]	**85.1**	86.4
PointTransformer	83.7	**86.6**

Table 4. Object part segmentation results on the ShapeNetPart dataset.

Object Part Segmentation

Figure 7. Visualization of object part segmentation results on the ShapeNetPart dataset. The ground truth is in the top row, Point Transformer predictions on the bottom.

Ablations

Pos. encoding	mIoU	mAcc	OA
none	64.6	71.9	88.2
absolute	66.5	73.2	88.9
relative	**70.4**	**76.5**	**90.8**
relative for attention	67.0	73.0	89.3
relative for feature	68.7	74.4	90.4

k	mIoU	mAcc	OA
4	59.6	66.0	86.0
8	67.7	73.8	89.9
16	**70.4**	**76.5**	**90.8**
32	68.3	75.0	89.8
64	67.7	74.1	89.9

Operator	mIoU	mAcc	OA
MLP	61.7	68.6	87.1
MLP+pooling	63.7	71.0	87.8
scalar attention	64.6	71.9	88.4
vector attention	**70.4**	**76.5**	**90.8**