CSC2457 3D & Geometric Deep Learning

Dynamic Graph CNN for Learning on Point Clouds

Yue Wang and Yongbin Sun and Ziwei Liu and Sanjay E. Sarma and Michael M. Bronstein and Justin M. Solomon

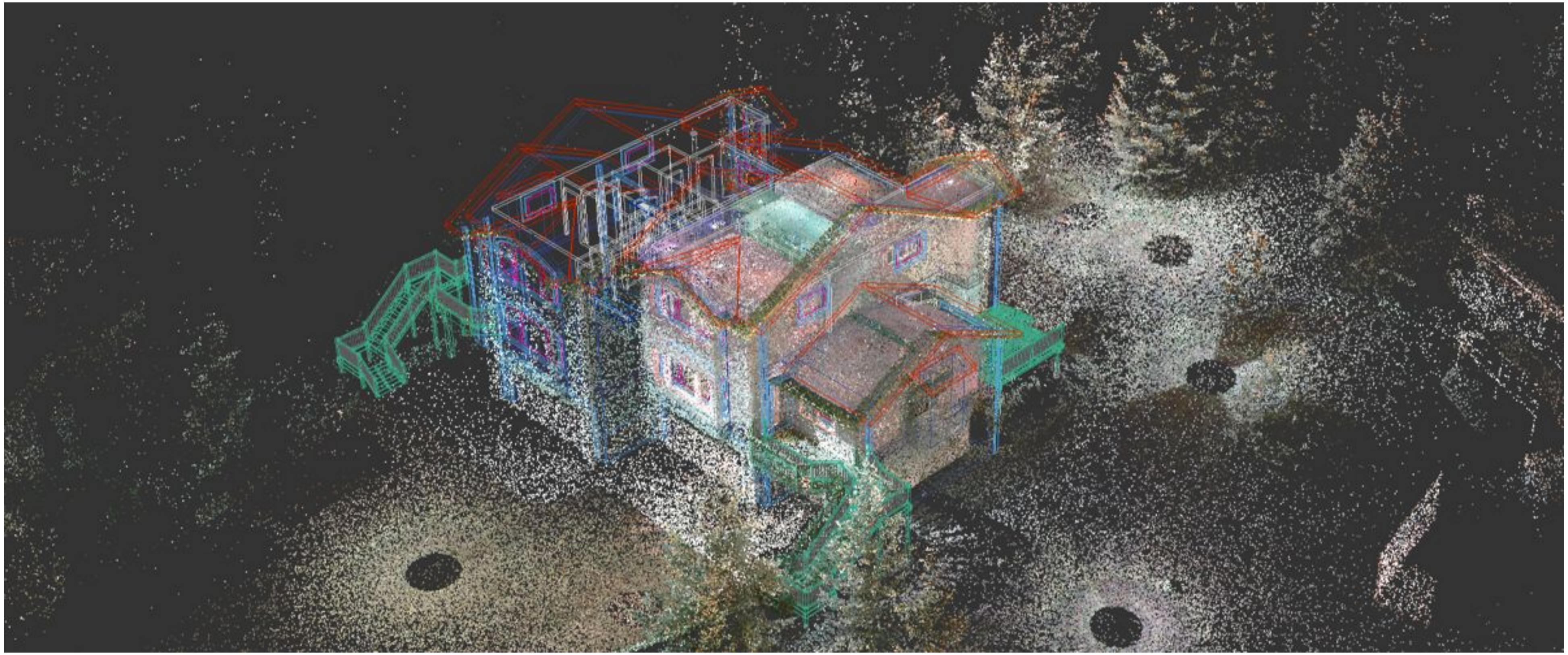
Date: January 26, 2021

Presenter: Mustafa Haiderbhai

Instructor: Animesh Garg

UNIVERSITY OF TORONTO

Motivation and Main Problem

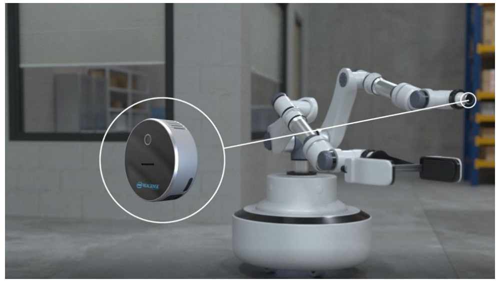


Motivation and Main Problem

Point cloud acquisition has become:

- Inexpensive
- Dense
- Accurate

Processing directly on raw point clouds has benefits in **speed**



Motivation and Main Problem

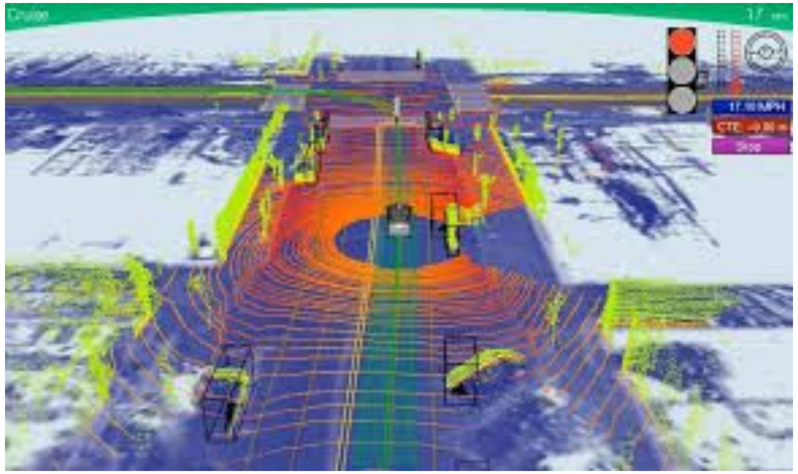
We want to extract high-level information about the scene

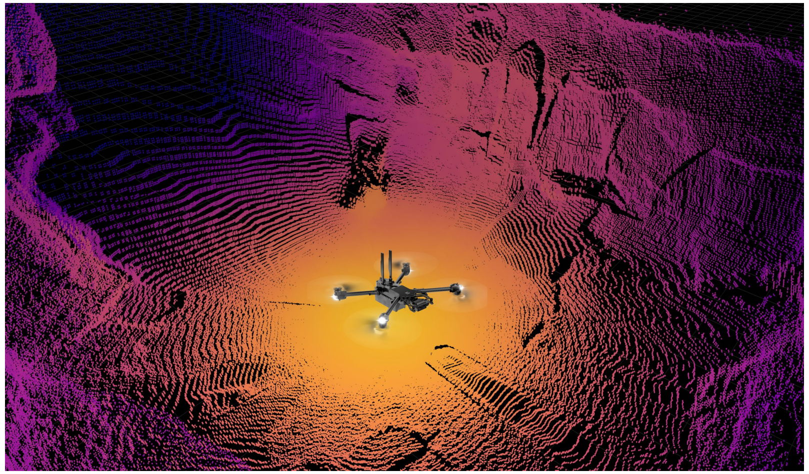
BUT 2D != 3D

SO

How can we take the successes of CNN on 2D, and apply them to 3D?

Motivation and Main Problem



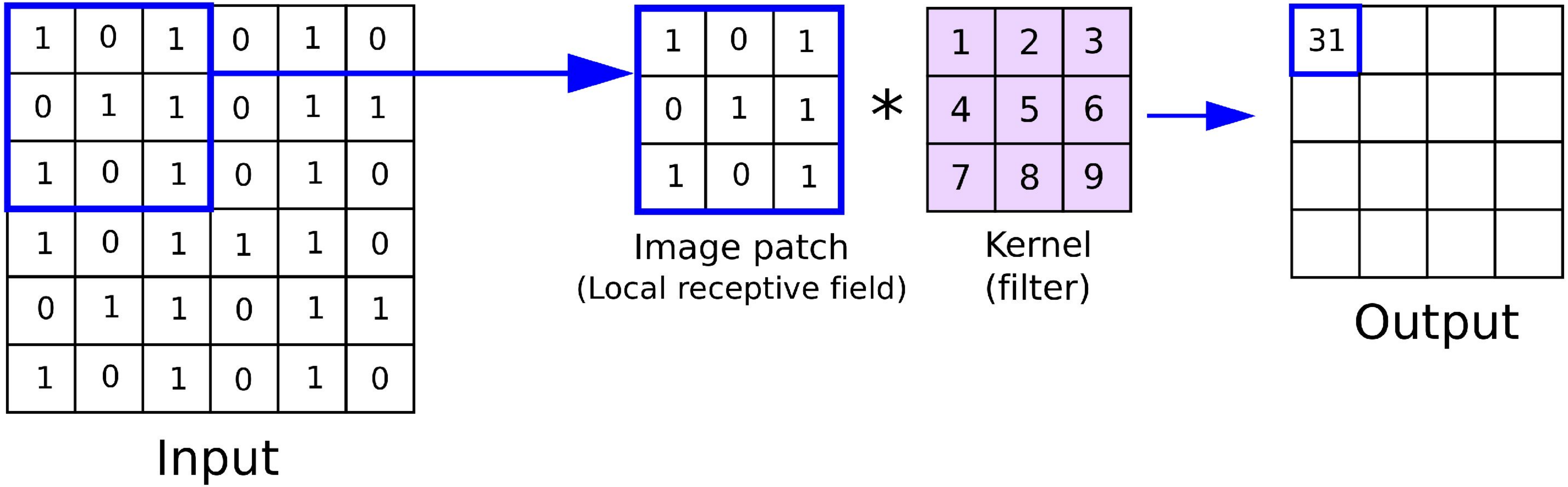


Previous research does not incorporate local information

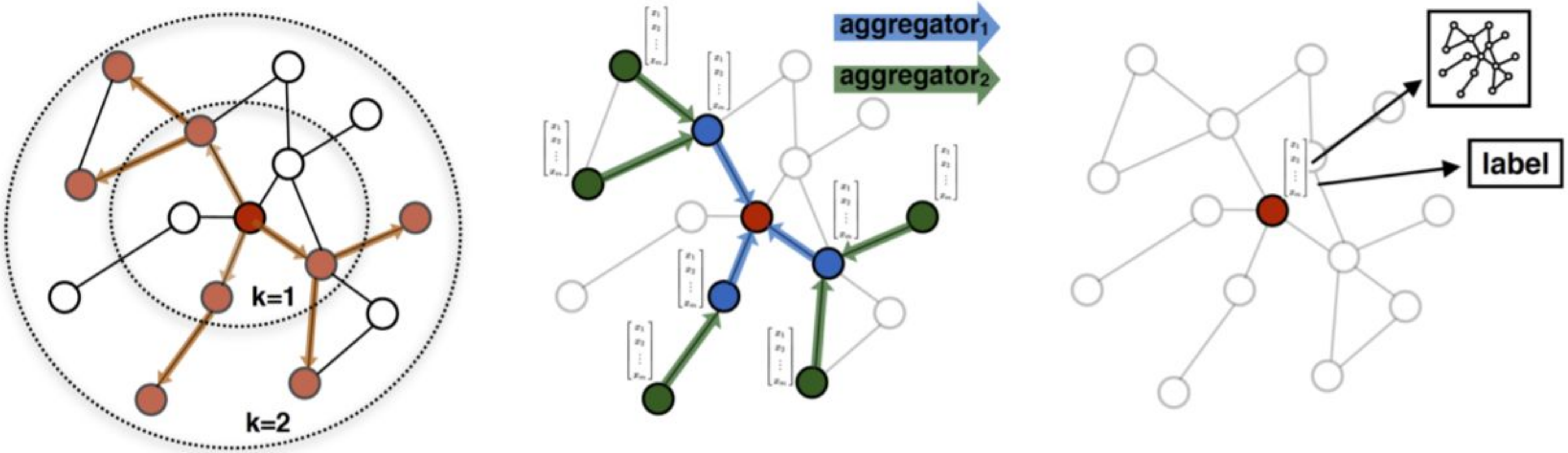
Contributions

- Extract high level information directly from point clouds
- Points must remain “permutation invariant”
- Prior work treats each point independantly, thereby ignoring local geometric information
- An operation can be performed to to transform local points into a graph, and applying convolution to the edges
- This operation can be repeatedly stacked to learn semantic relationships between groups of points
- Achieves state-of-the-art performance, and shows semantic labelling across large distances and sepeations

General Background



General Background

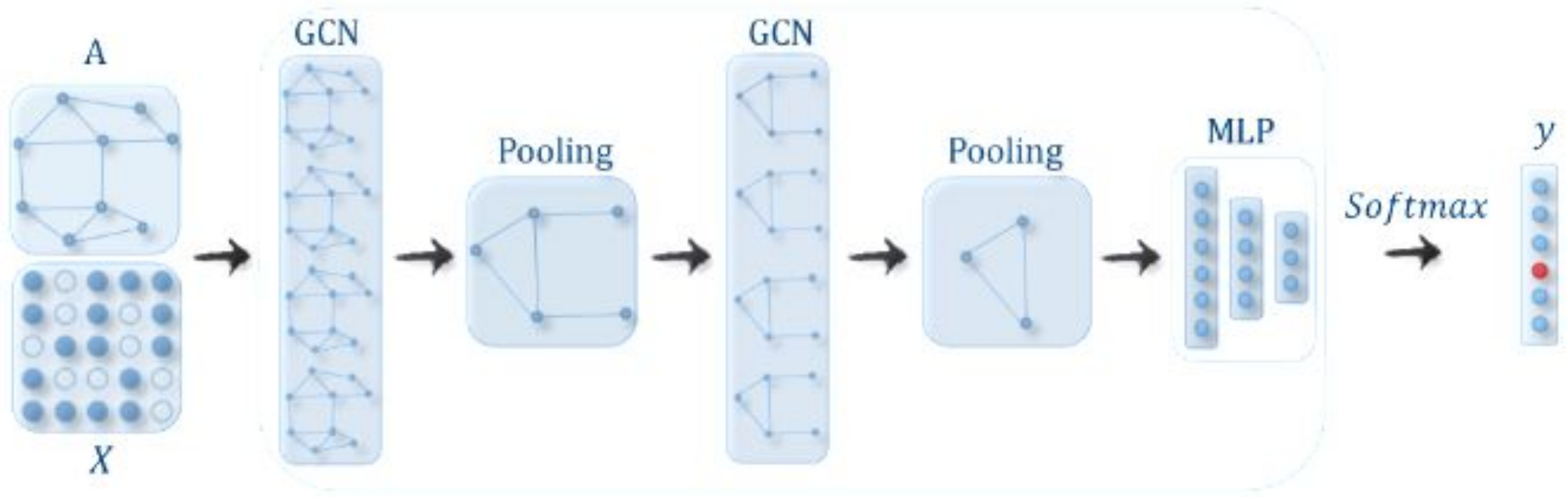


1. Sample neighborhood

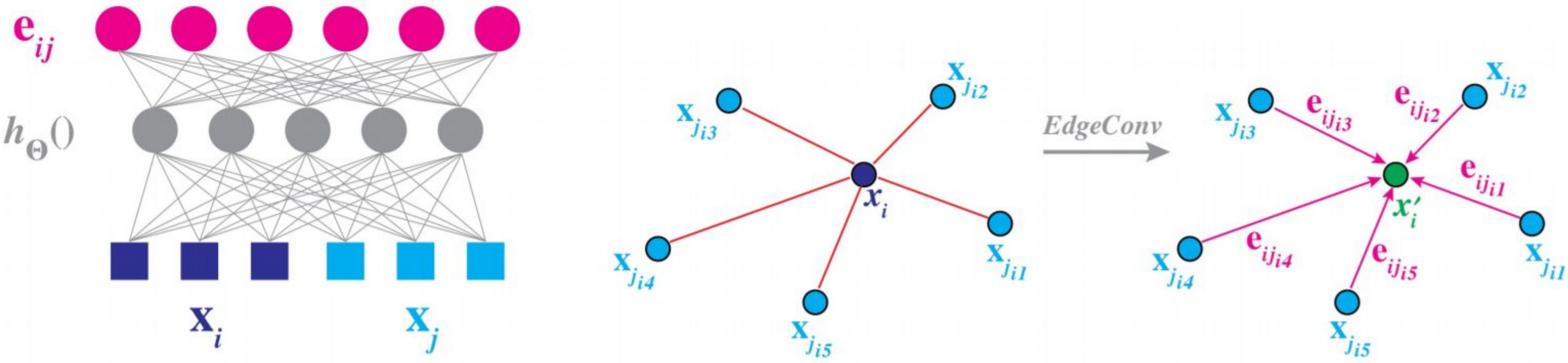
2. Aggregate feature information from neighbors

3. Predict graph context and label using aggregated information

General Background

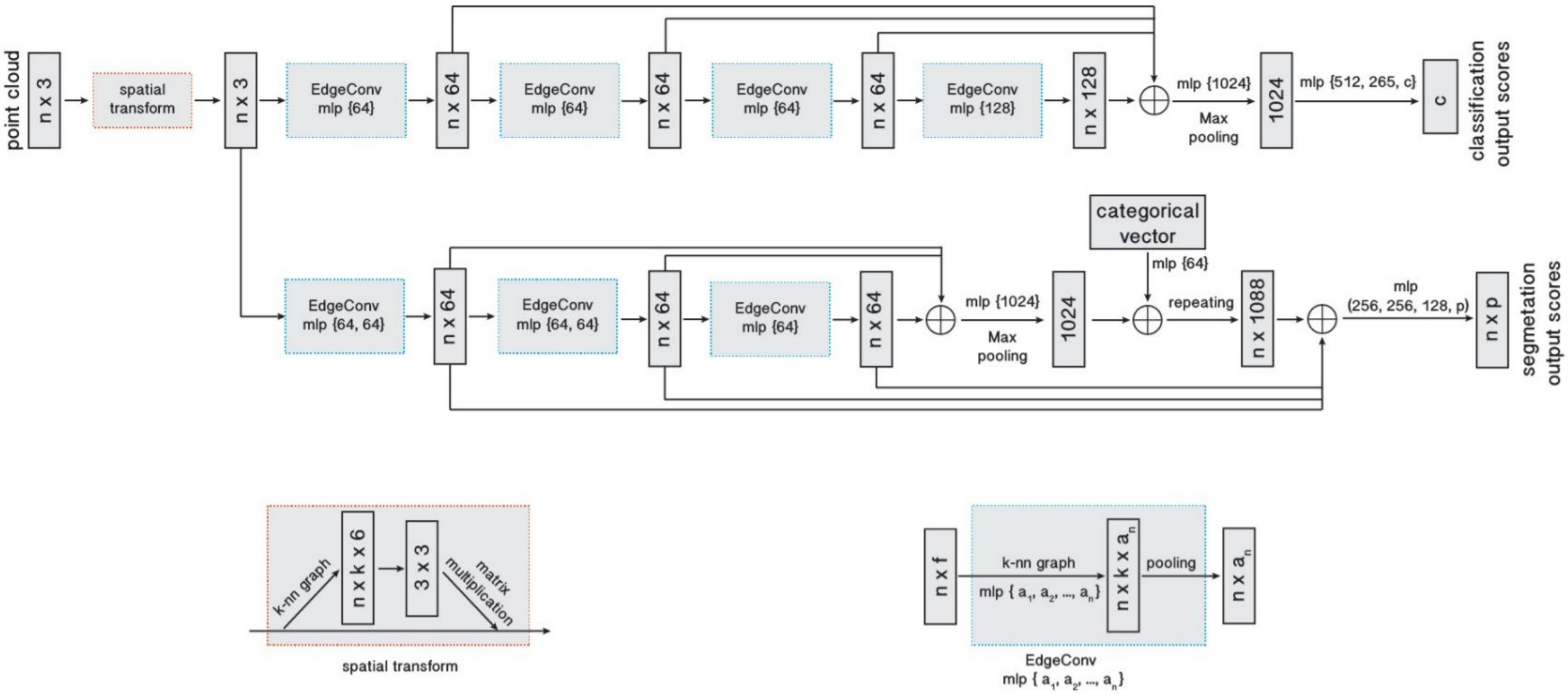


Problem Setting



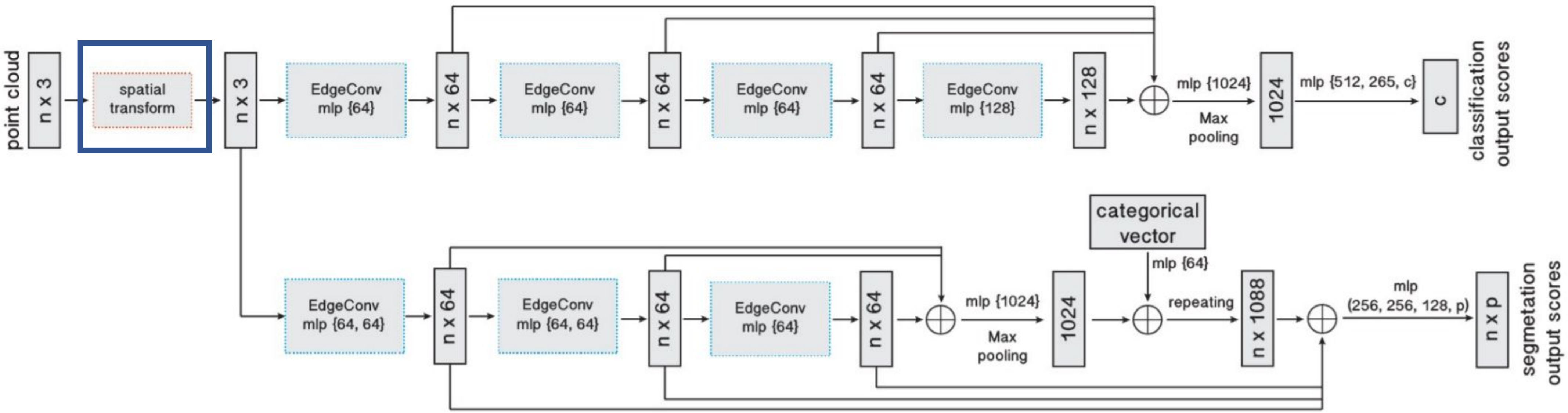
$$h_\Theta(\mathbf{x}_i, \mathbf{x}_j) = \bar{h}_\Theta(\mathbf{x}_i, \mathbf{x}_j - \mathbf{x}_i). \qquad (7)$$

Methods

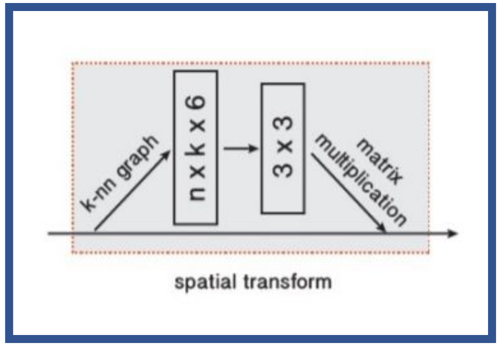


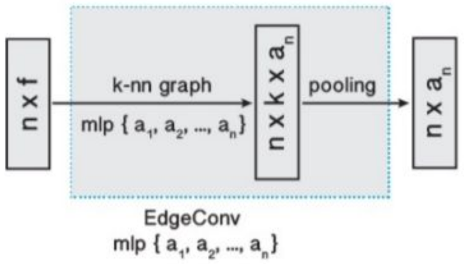
point cloud
n x 3
spatial transform
n x 3
EdgeConv mlp {64}
n x 64
EdgeConv mlp {64}
n x 64
EdgeConv mlp {64}
n x 64
EdgeConv mlp {128}
n x 128
mlp {1024}
Max pooling
1024
mlp {512, 265, c}
c
classification output scores
categorical vector
mlp {64}
EdgeConv mlp {64, 64}
n x 64
EdgeConv mlp {64, 64}
n x 64
EdgeConv mlp {64}
n x 64
mlp {1024}
Max pooling
1024
repeating
n x 1088
mlp (256, 256, 128, p)
n x p
segmetation output scores
k-nn graph
n x k x 6
3 x 3
matrix multiplication
spatial transform
n x f
k-nn graph
mlp { a1, a2, ..., an}
n x k x an
pooling
n x an
EdgeConv
mlp { a1, a2, ..., an}

Methods

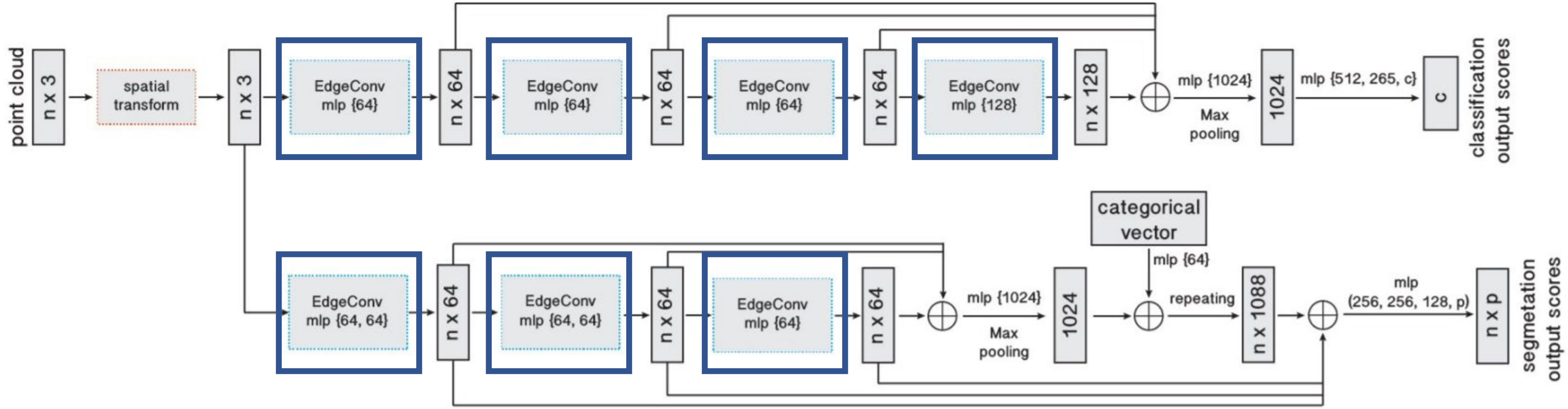


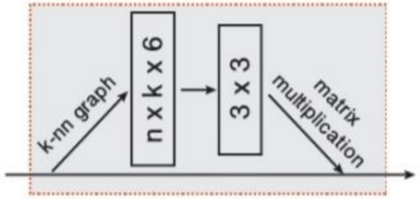
Converts point cloud into **canonical space** using co-ordinate differences of point with k-neighbours



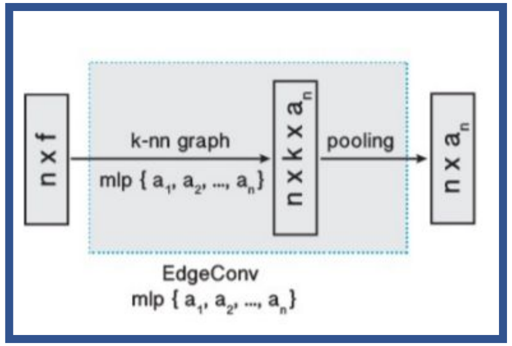


Methods



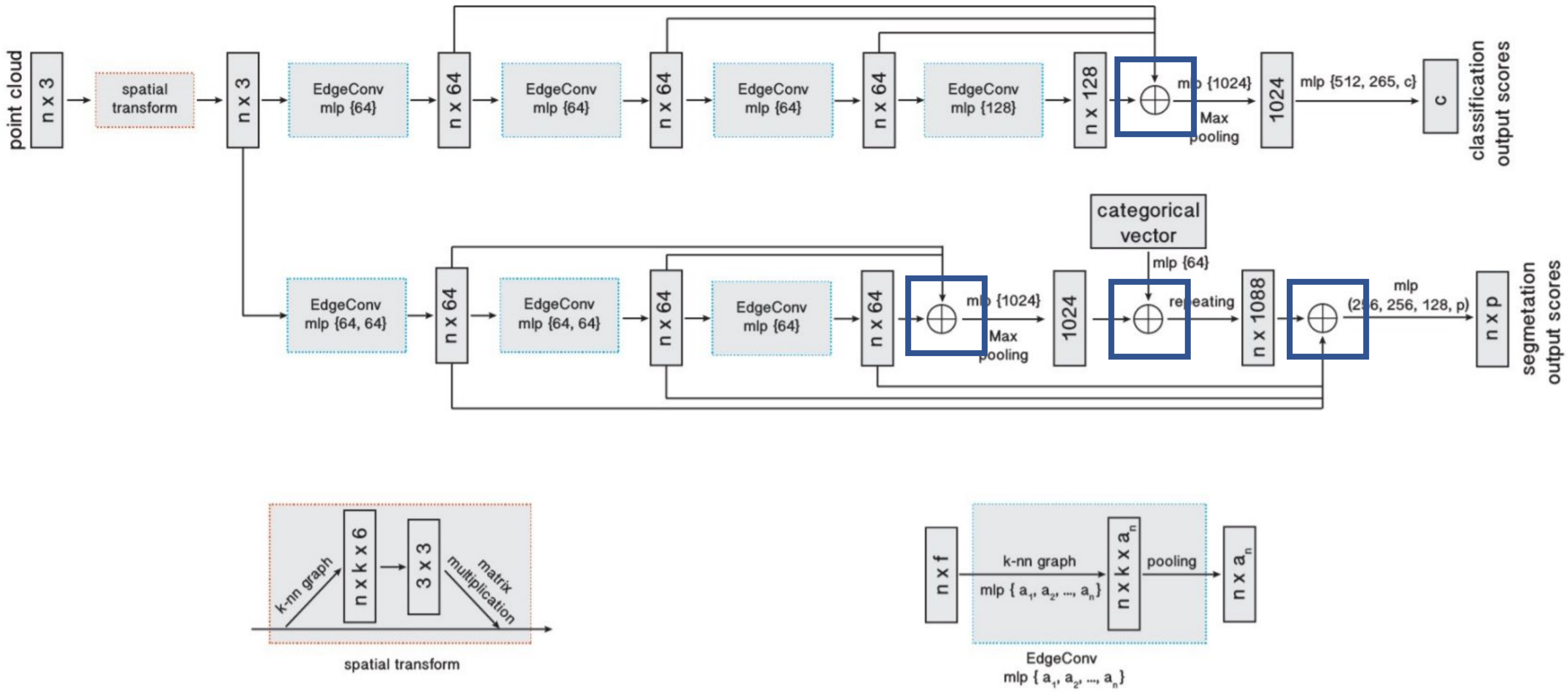


spatial transform



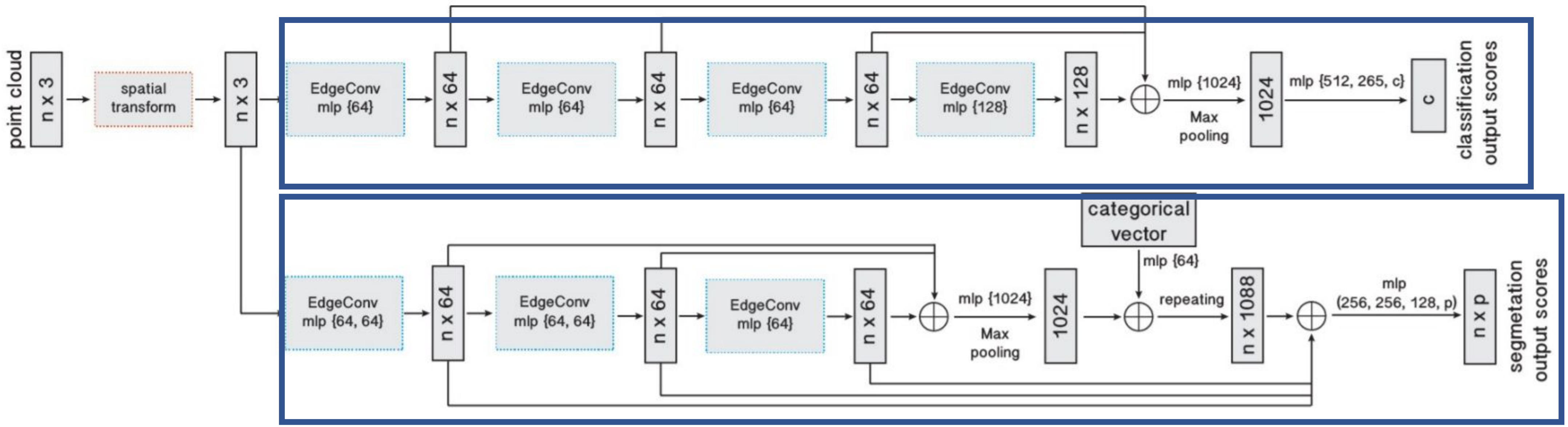
Converts point cloud into **canonical space** using co-ordinate differences of point with k-neighbours

Methods

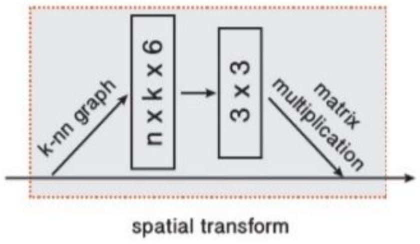


point cloud
n x 3
spatial transform
n x 3
EdgeConv mlp {64}
n x 64
EdgeConv mlp {64}
n x 64
EdgeConv mlp {64}
n x 64
EdgeConv mlp {128}
n x 128
mlp {1024}
Max pooling
1024
mlp {512, 265, c}
c
classification output scores
categorical vector
mlp {64}
EdgeConv mlp {64, 64}
n x 64
EdgeConv mlp {64, 64}
n x 64
EdgeConv mlp {64}
n x 64
mlp {1024}
Max pooling
1024
repeating
n x 1088
mlp
(256, 256, 128, p)
n x p
segmetation output scores
k-nn graph
n x k x 6
3 x 3
matrix multiplication
spatial transform
n x f
k-nn graph
mlp { a1, a2, ..., an}
n x k x an
pooling
n x an
EdgeConv
mlp { a1, a2, ..., an}

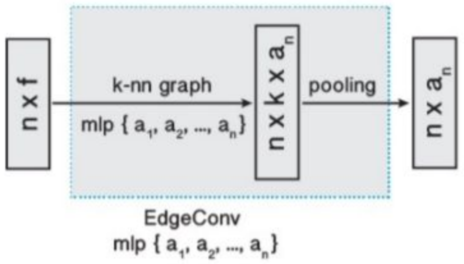
Methods



point cloud
n x 3
spatial transform
n x 3
EdgeConv mlp {64}
n x 64
EdgeConv mlp {64}
n x 64
EdgeConv mlp {64}
n x 64
EdgeConv mlp {128}
n x 128
mlp {1024}
Max pooling
1024
mlp {512, 265, c}
c
classification output scores
categorical vector
mlp {64}
EdgeConv mlp {64, 64}
n x 64
EdgeConv mlp {64, 64}
n x 64
EdgeConv mlp {64}
n x 64
mlp {1024}
Max pooling
1024
repeating
n x 1088
mlp (256, 256, 128, p)
n x p
segmetation output scores



k-nn graph
n x k x 6
3 x 3
matrix multiplication
spatial transform



n x f
k-nn graph
mlp { a1, a2, ..., an }
n x k x an
pooling
n x an
EdgeConv
mlp { a1, a2, ..., an }

Experimental Results

	Mean Class Accuracy	Overall Accuracy
3DShapeNets [Wu et al. 2015]	77.3	84.7
VoxNet [Maturana and Scherer 2015]	83.0	85.9
Subvolume [Qi et al. 2016]	86.0	89.2
VRN (single view) [Brock et al. 2016]	88.98	-
VRN (multiple views) [Brock et al. 2016]	91.33	-
ECC [Simonovsky and Komodakis 2017]	83.2	87.4
PointNet [Qi et al. 2017b]	86.0	89.2
PointNet++ [Qi et al. 2017c]	-	90.7
Kd-net [Klokov and Lempitsky 2017]	-	90.6
PointCNN [Li et al. 2018a]	88.1	92.2
PCNN [Atzmon et al. 2018]	-	92.3
Ours (baseline)	88.9	91.7
Ours	**90.2**	**92.9**
Ours (2048 points)	**90.7**	**93.5**

Table 2. Classification results on ModelNet40.

Experimental Results

State of the art performance while maintaining high efficiency

	Mean Class Accuracy	Overall Accuracy
3DShapeNets [Wu et al. 2015]	77.3	84.7
VoxNet [Maturana and Scherer 2015]	83.0	85.9
Subvolume [Qi et al. 2016]	86.0	89.2
VRN (single view) [Brock et al. 2016]	88.98	-
VRN (multiple views) [Brock et al. 2016]	91.33	-
ECC [Simonovsky and Komodakis 2017]	83.2	87.4
PointNet [Qi et al. 2017b]	86.0	89.2
PointNet++ [Qi et al. 2017c]	-	90.7
Kd-net [Klokov and Lempitsky 2017]	-	90.6
PointCNN [Li et al. 2018a]	88.1	92.2
PCNN [Atzmon et al. 2018]	-	92.3
Ours (baseline)	88.9	91.7
Ours	**90.2**	**92.9**
Ours (2048 points)	**90.7**	**93.5**

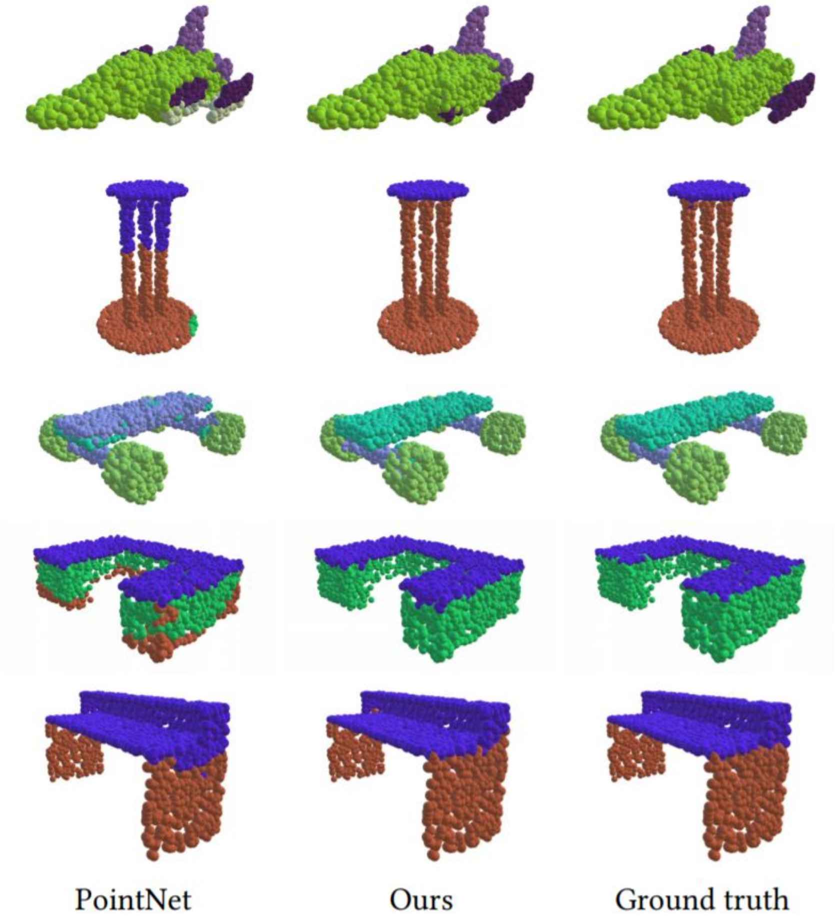
Table 2. Classification results on ModelNet40.

	Model size(MB)	Time(ms)	Accuracy(%)
PointNet (Baseline) [Qi et al. 2017b]	9.4	6.8	87.1
PointNet [Qi et al. 2017b]	40	16.6	89.2
PointNet++ [Qi et al. 2017c]	12	163.2	90.7
PCNN [Atzmon et al. 2018]	94	117.0	92.3
Ours (Baseline)	11	19.7	91.7
Ours	21	27.2	92.9

Table 3. Complexity, forward time, and accuracy of different models

Experimental Results

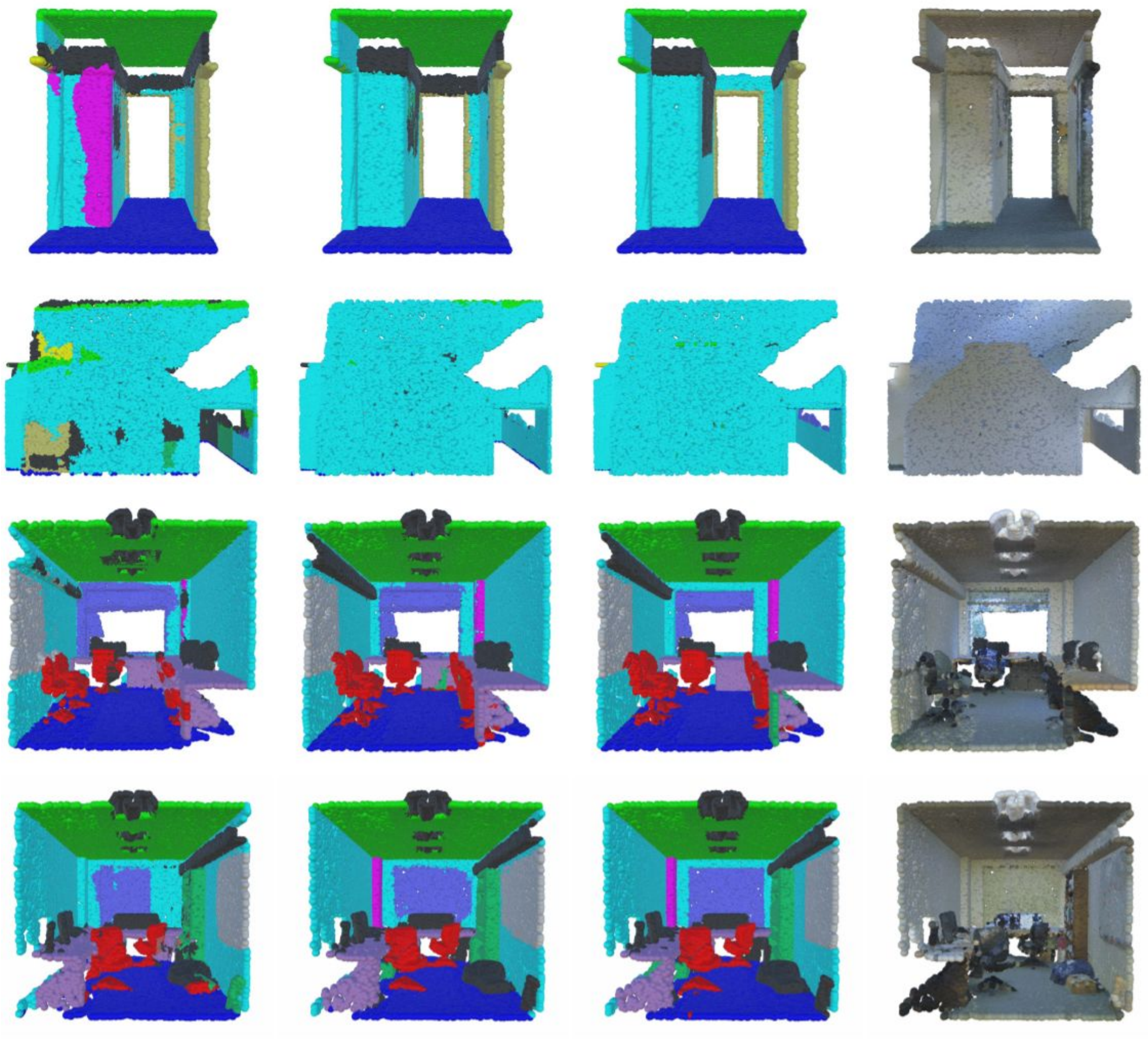
State of the art performance while maintaining high efficiency



Experimental Results

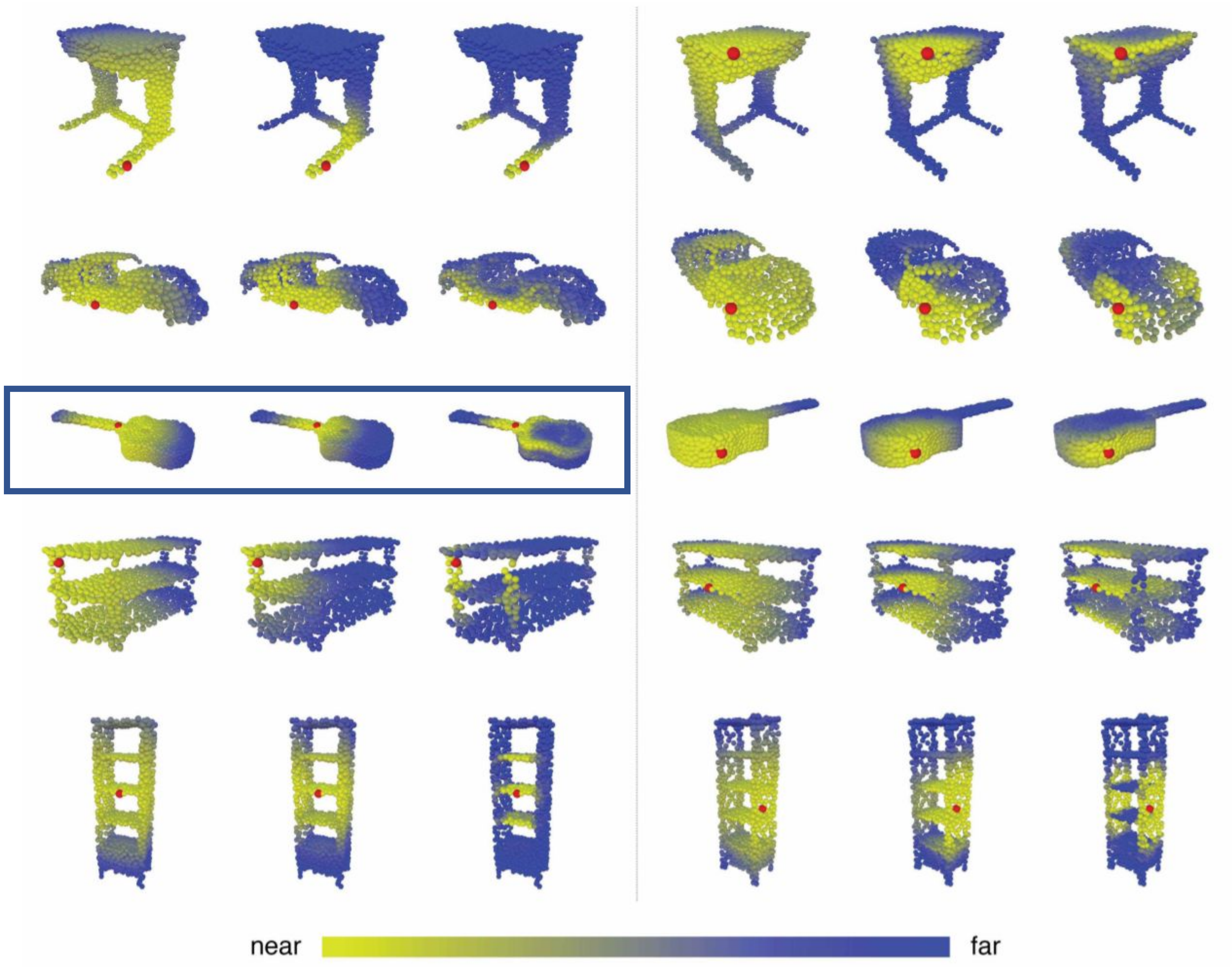
	MEAN	AREO	BAG	CAP	CAR	CHAIR	EAR PHONE	GUITAR	KNIFE	LAMP	LAPTOP	MOTOR	MUG	PISTOL	ROCKET	SKATE BOARD	TABLE
# SHAPES		2690	76	55	898	3758	69	787	392	1547	451	202	184	283	66	152	5271
PointNet	83.7	83.4	78.7	82.5	74.9	89.6	73.0	91.5	85.9	80.8	95.3	65.2	93.0	81.2	57.9	72.8	80.6
PointNet++	85.1	82.4	79.0	**87.7**	77.3	**90.8**	71.8	91.0	85.9	83.7	95.3	71.6	94.1	81.3	58.7	76.4	82.6
Kd-Net	82.3	80.1	74.6	74.3	70.3	88.6	73.5	90.2	87.2	81.0	94.9	57.4	86.7	78.1	51.8	69.9	80.3
LocalFeatureNet	84.3	**86.1**	73.0	54.9	77.4	88.8	55.0	90.6	86.5	75.2	**96.1**	57.3	91.7	83.1	53.9	72.5	**83.8**
PCNN	85.1	82.4	80.1	85.5	79.5	**90.8**	73.2	91.3	86.0	85.0	95.7	73.2	94.8	83.3	51.0	75.0	81.8
PointCNN	**86.1**	84.1	**86.45**	86.0	**80.8**	90.6	**79.7**	**92.3**	**88.4**	**85.3**	**96.1**	**77.2**	**95.3**	**84.2**	**64.2**	**80.0**	83.0
Ours	85.2	84.0	83.4	86.7	77.8	90.6	74.7	91.2	87.5	82.8	95.7	66.3	94.9	81.1	63.5	74.5	82.6

Table 6. Part segmentation results on ShapeNet part dataset. Metric is mIoU(%) on points.

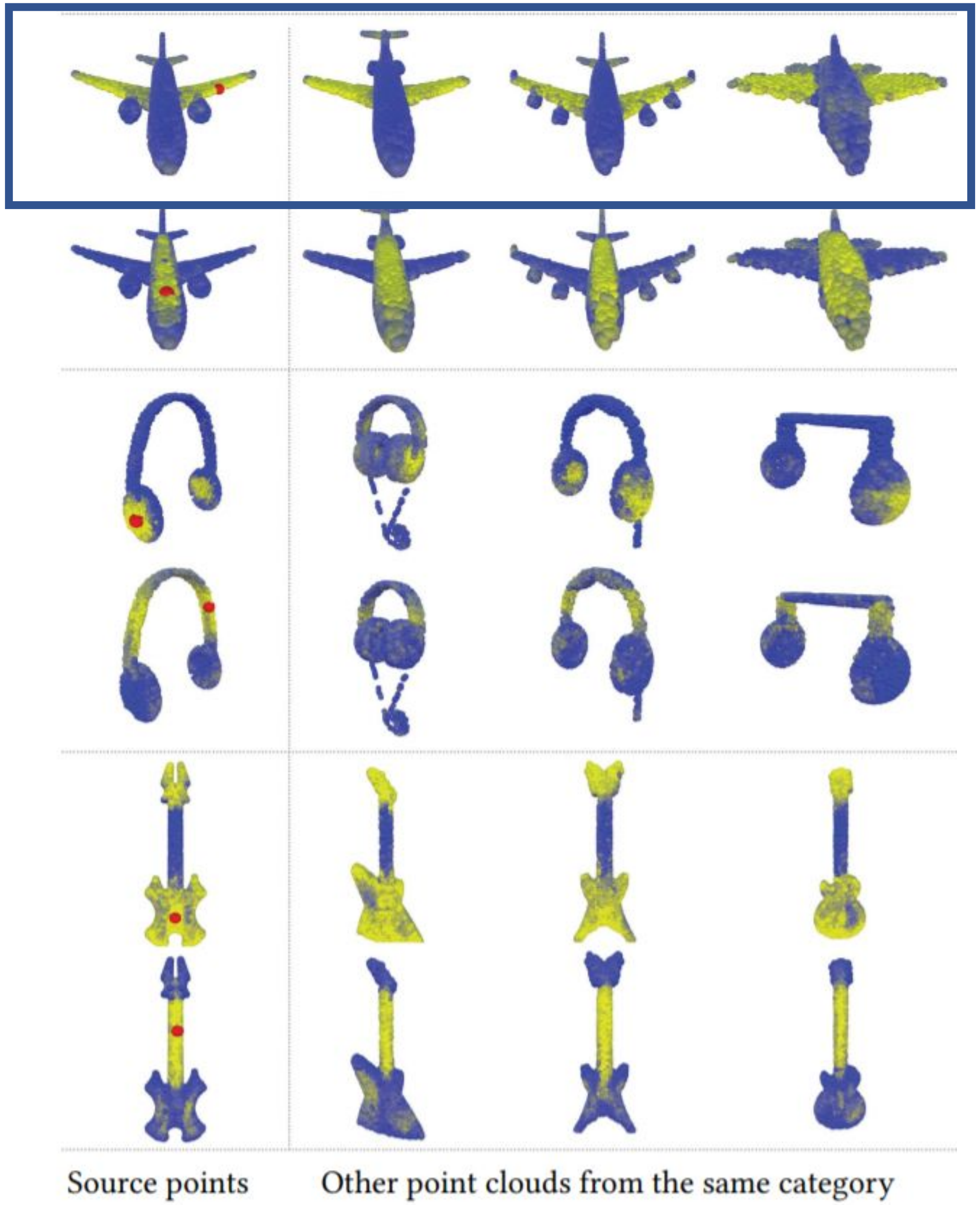


PointNet Ours Ground truth Real color

Distance in feature space in early, middle, and late stages of the newtork



Distance in feature space from source point (red) between separate point clouds



Discussion of results

- Principle of convolutions can be applied to point clouds
- Able to capture local information

- Interestingly able to capture semantic groupings across large distances
- Also able to transfer same distance in feature space to other point clouds

Critique / Limitations / Open Issues

- Why not state-of-the-art in part segmentation?

- By focusing on edges, this method ignores the relative positioning of points
  - Features are aggregated in patches, and thus deformation of the patches will not be seen
- Are relationships in high-level feature space robust?

Contributions (Recap)

- This paper learns **high-level information** from points clouds
- Point clouds are not the same as images, new techniques need to be applied
- Prior work did not incorporate **local information/geometry**
- By transforming the **point cloud into a graph**, convolutions can be applied
- **Dynamically** reconstructing graph produces stronger results
- State-of-the art performance on classification, and strong performance in part segmentation