Graph Neural Networks For Point Cloud Analysis

Juan Leon-Alcazar

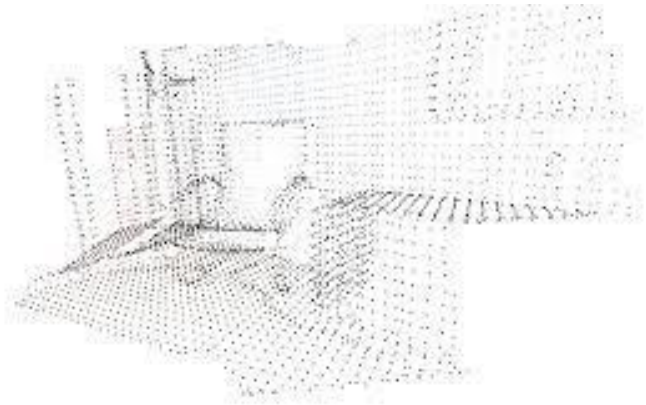
Point Clouds

A point cloud is a set of points in space, each point has an associated feature vector.

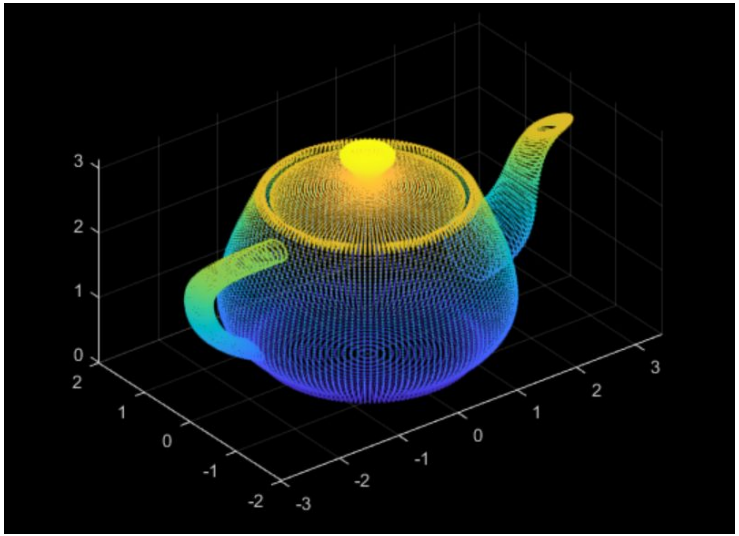
- Is a widely used format for 3D sensors such as LIDAR.

Since it is an **unordered** set of points it is invariant to permutations of its members.

Some Samples



http://www.open3d.org/docs/0.13.0/tutorial/geometry/pointcloud.html



https://www.revopoint3d.com/point-cloud-and-3d-image/

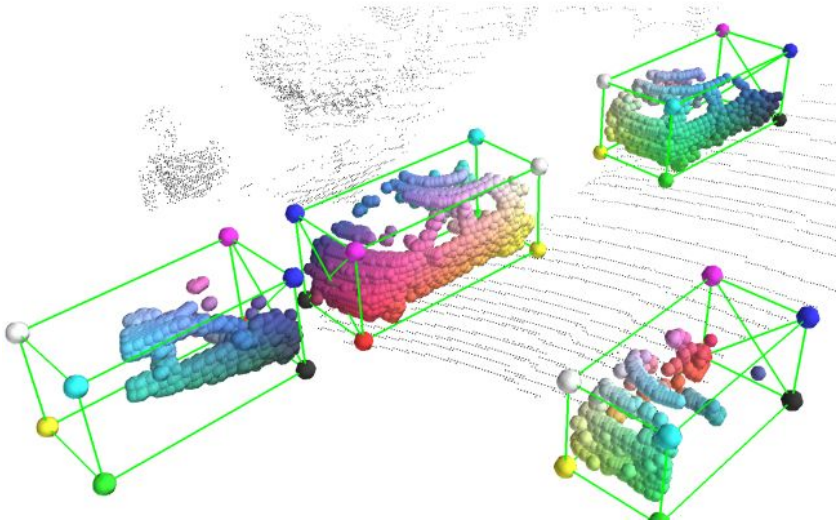
Tasks Over Point Clouds

Classification: Generate a global prediction over the pointcloud, typically. What type of shape is this?.

Detection: Generate a prediction over a subset of the cloud. What element inside the shape is this?

Segmentation: Generate an individual prediction for each point in the cloud.

Point Cloud Detection



https://arxiv.org/pdf/1907.03670.pdf

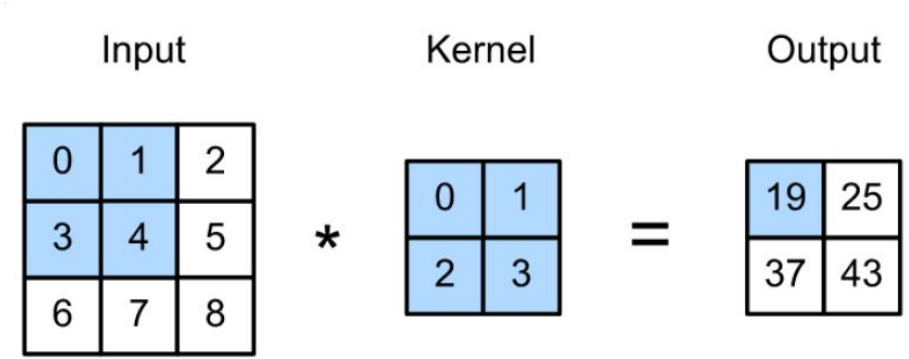
Point Cloud Semantic Segmentation

http://stanford.edu/~rqi/pointnet/

Working on Point Cloud Data

Convolutional architectures require highly regular input data formats, suitable for image grids or 3D voxels.

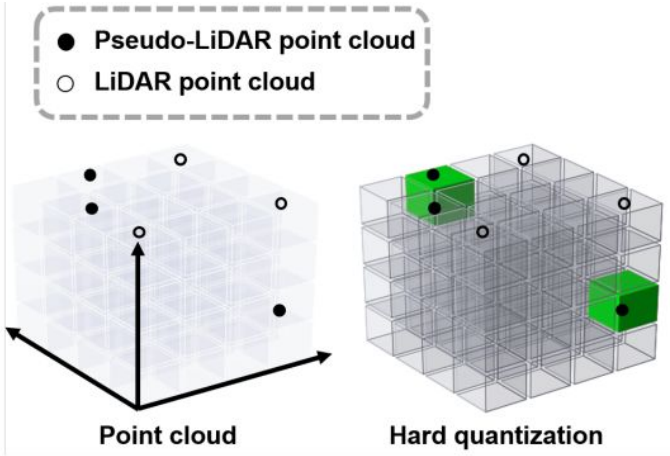
Due to to the irregular format, on point clouds some approaches transform such data to regular 3D D voxel grids or collections of images.



https://courses.cs.washington.edu/courses/cse446/22wi/sections/08/convolutional_networks.html

Point Cloud Quantization

Quantizations work…. but results in voluminous data, while also introducing quantization artifacts.

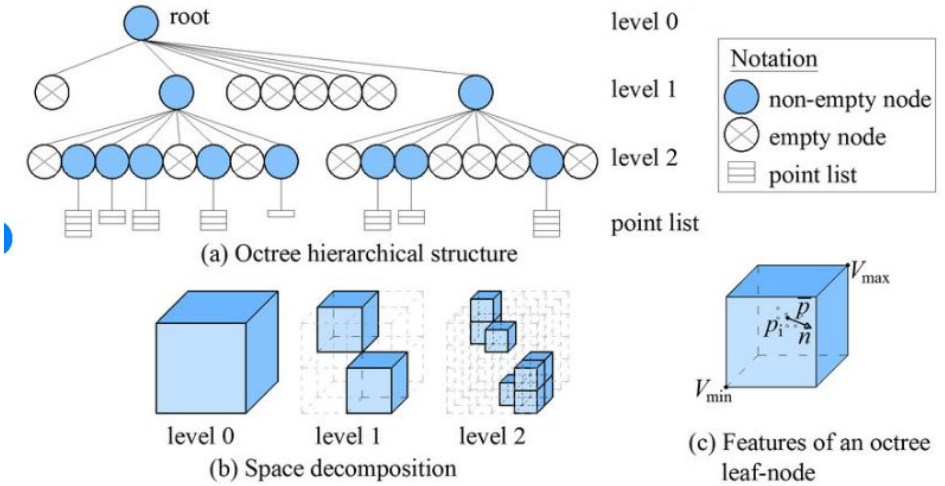


https://arxiv.org/pdf/2004.03080.pdf

You can Play Arround With Data Structures

An **octree** is a tree data structure. Octrees are used for spatial partitioning of 3D point clouds.

Non-empty leaf nodes of an octree contain one or more points that fall within the same spatial subdivision.

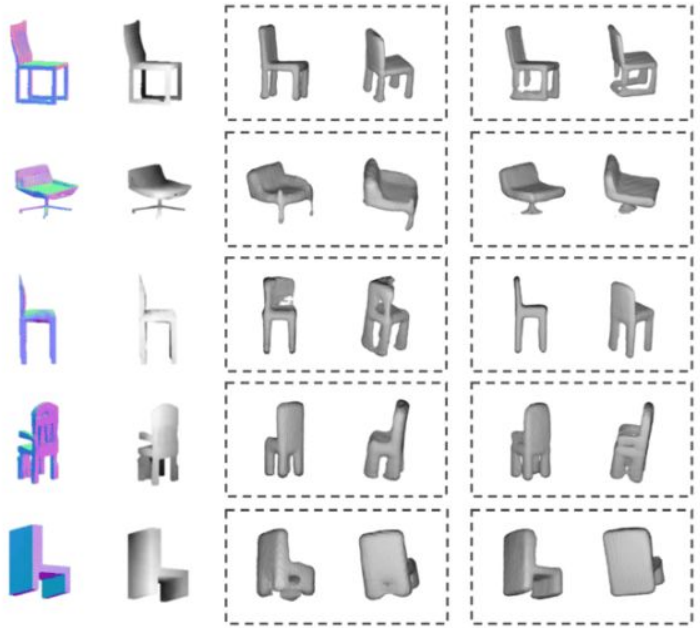


https://www.researchgate.net/figure/Example-of-octree-decomposition_fig4_274645446

Working Without Quantization

The density and other attributes of points **may not be uniform.**

The order of the analysis should not matter (so forget RNNs are hard to adapt).

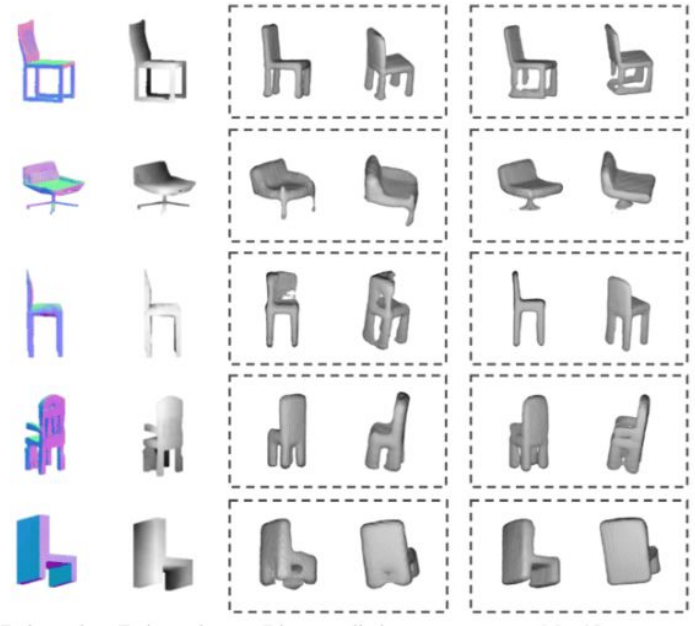


https://www.researchgate.net/figure/Results-on-rendered-images-of-ShapeNet-objects-Chang-et-al-2015-From-left-to-right_fig1_320976023

Working Without Quantization

Point clouds are **invariant to global transformation.**

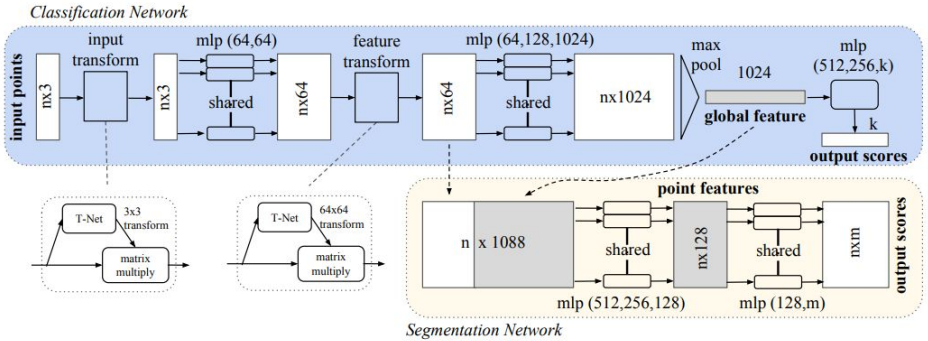
But very sensible to local transformations (which is not the case of images).



https://www.researchgate.net/figure/Results-on-rendered-images-of-ShapeNet-objects-Chang-et-al-2015-From-left-to-right_fig1_320976023

A First Step: PointNet

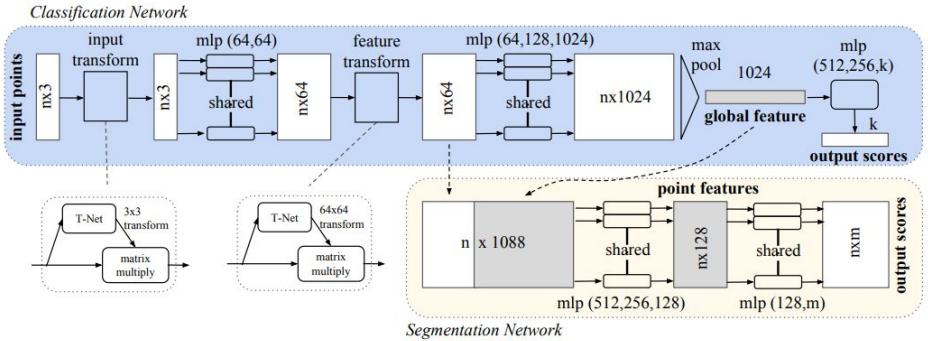
PointNet is a deep net architecture that **consumes point clouds** for applications ranging from object classification, part segmentation, to scene semantic parsing.



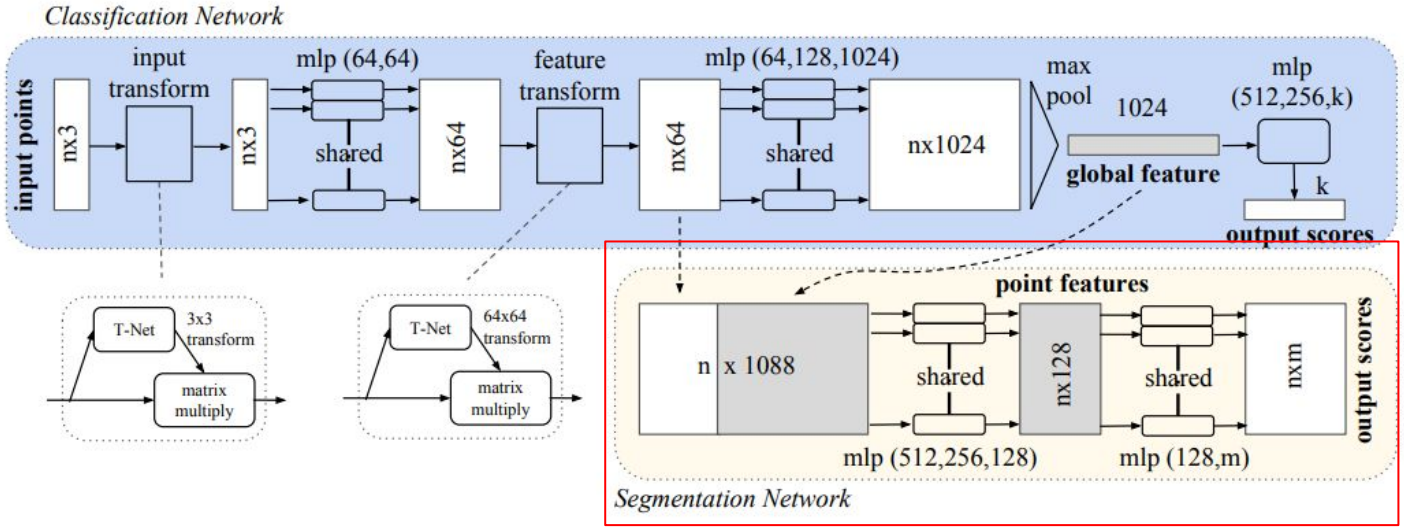
A First Step: PointNet

Approximate a general function defined on a point set by applying a symmetric function on transformed elements in the set:

$$f(\{x_1, \ldots, x_n\}) \approx g(h(x_1), \ldots, h(x_n)),$$



A First Step: PointNet



Connection To Graph Neural Networks

We are looking for permutation invariant g functions

- Just like the graph diffusion operators.

Function f is a local feature transformation over individual nodes

- Shared along all the available nodes

$$f(\{x1, \ldots, xn\}) \approx g(h(x1), \ldots, h(xn)),$$

Connection To Graph Neural Networks

We should be able to map

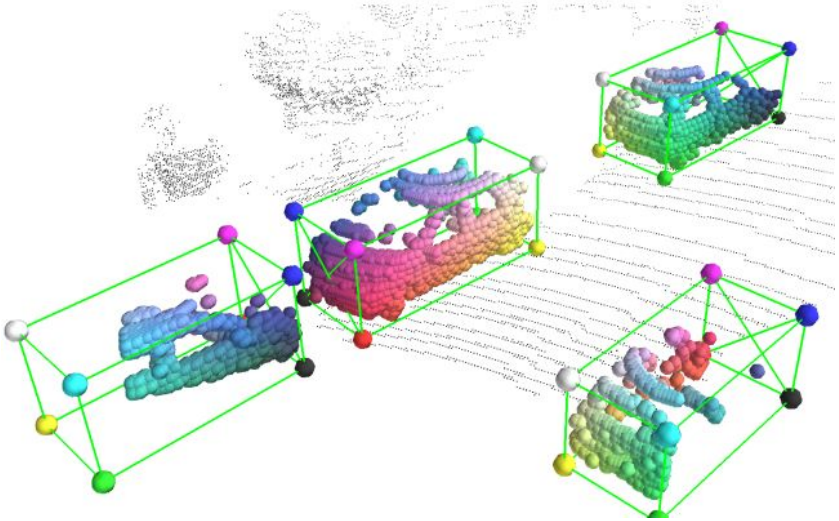
$$f(\{x1, \ldots, xn\}) \approx g(h(x1), \ldots, h(xn))$$

Into a **graph convolution.**

In fact the GCNConv layer is a special case of f where g is an average and h is learned weight.

Connection To Graph Neural Networks

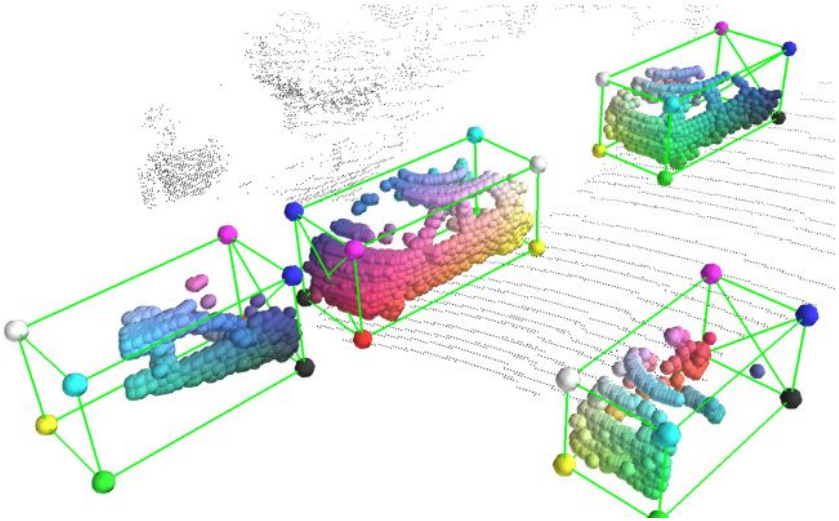
We are missing something right?



Connection To Graph Neural Networks

We are missing something right?

How do we make edges in a point cloud?



PointNet++

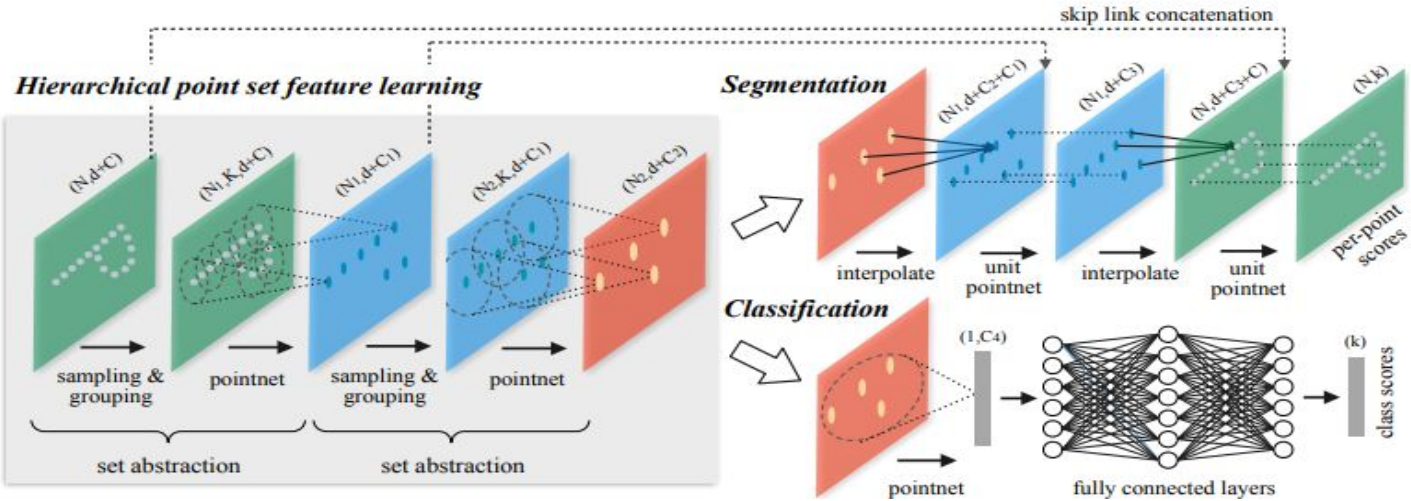
Partition the set of points into **overlapping local regions** by the distance metric of the underlying space.

group the local features are into larger units and keep processing to produce higher level features.

PointNet++ applies PointNet recursively on a nested partitioning of the input set

- This is the very same inductive bias found on CNNs

Feature Pooling



PointNet builds a hierarchical grouping of points and progressively abstract larger and larger local regions along the hierarchy.

PointNet++ Formal Definition

Given an unordered point set {x1, x2, ..., xn} we define a set function f : X → R

where γ and h are usually multi-layer perceptron (MLP) networks.

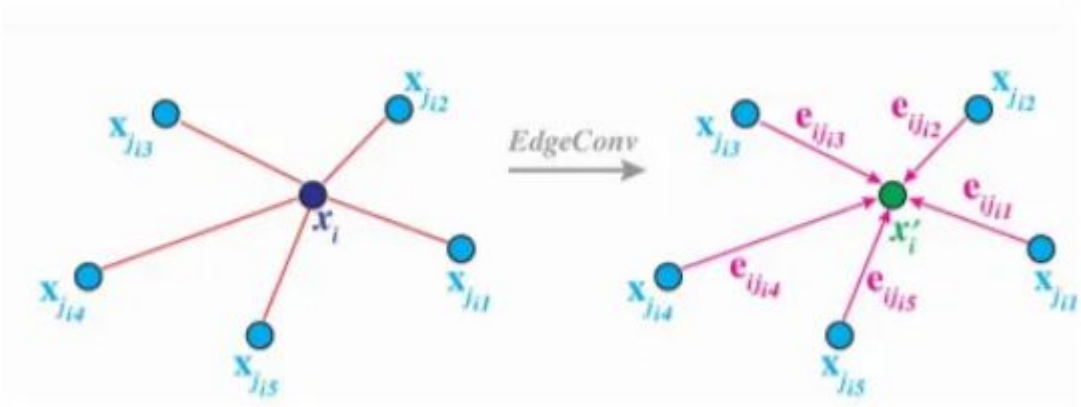
$$f(x_1, x_2, ..., x_n) = \gamma \left(\underset{i=1,...,n}{\text{MAX}} \{h(x_i)\} \right)$$

Edge Convolution

- Is **invariant** to input point permutations
- Can arbitrarily approximate any continuous set function.

A set abstraction level takes an N × (d + C) matrix as input that is from N points with d-dim coordinates and C-dim point feature. It outputs an N0 × (d + C 0) matrix of N0 subsampled points with d-dim coordinates and new C 0 -dim feature vectors summarizing local context

Edge Convolution



$$\mathbf{x}_i' = \sum_{j \in \mathcal{N}(i)} h_{\mathbf{\Theta}}(\mathbf{x}_i \,\|\, \mathbf{x}_j - \mathbf{x}_i),$$

$H(Xi,Xj) = \Theta Xj$	Classical Euclidean convolution
$H(Xi,Xj) = H(Xi)$	Encoding global information of the local feature structure. (Pointnet)
$H(Xi,Xj) = H(Xj - Xi)$	Encodes the local information, considering the shape as collection of small patches and losing the global information.
$H(Xi,Xj) = H(Xi, Xj - Xi)$	Edge function which combines the global shape structure(Xi) and local shape features(Xj-Xi)