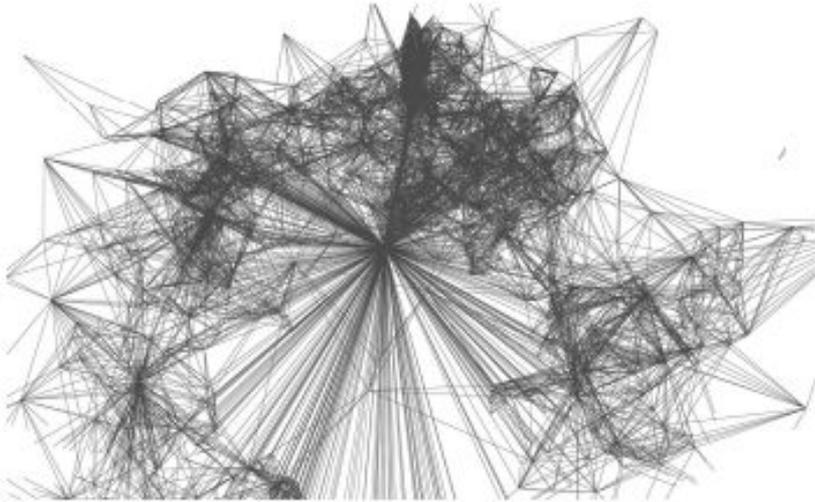


Nuages de Points et Modélisation 3D

5 - Machine learning II

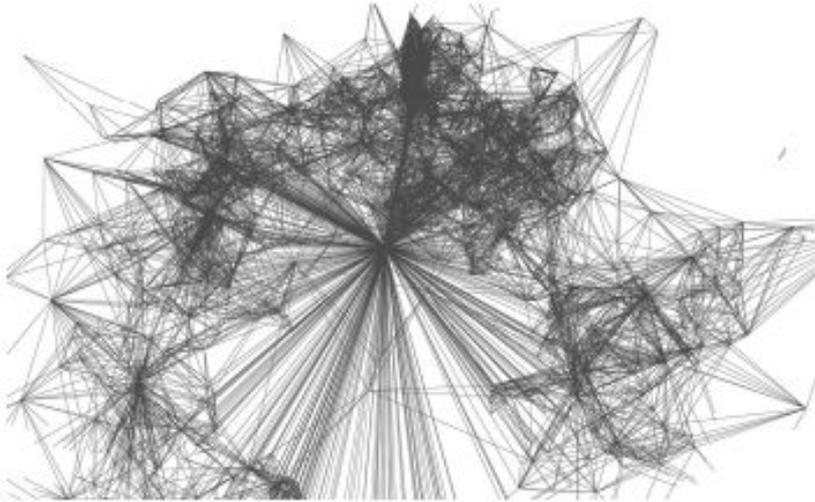
Overview

- I. Image-based approaches
 - A. Principles
 - B. Classification
 - C. Segmentation
- II. Geometric deep learning
 - A. Graph convolution



Overview

- I. Image-based approaches
 - A. Principles
 - B. Classification
 - C. Segmentation
- II. Geometric deep learning
 - A. Graph convolution



Idea

Image-based approaches



Image
processing



Image processing is a well studied problem

Idea

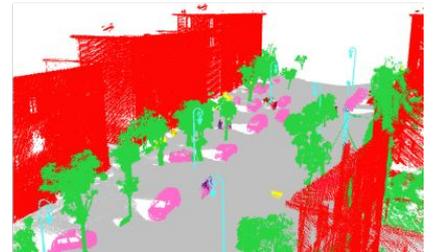
Image-based approaches



Image
processing

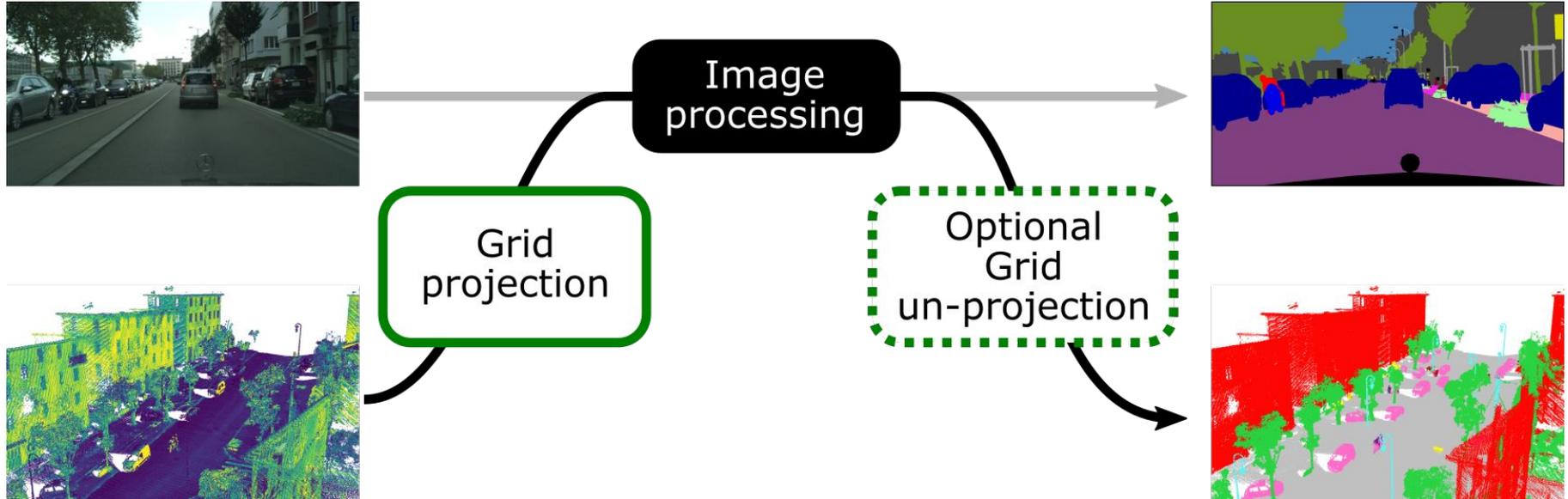


?



Idea

Image-based approaches



Regular grid projections

Image-based approaches

Images are pixels arrays

Implicit neighborhoods

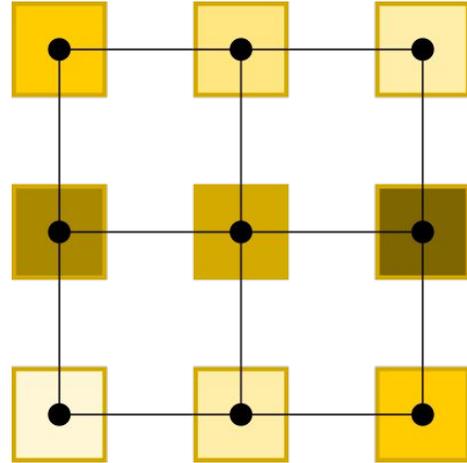
Information is in the color and relative position of the pixels

Thanks to this grid structure

Optimized network architectures

Fast (hardware optimization)

Relatively low memory cost



2D projections

Image-based approaches

2D convolution for an image patch centered on pixel n :

$$\mathbf{h}[n] = \sum_{f \in \{1, \dots, C\}} \sum_{m \in \{-M/2, \dots, M/2\}^2} \mathbf{K}_f[m] \mathbf{f}_f[n + m]$$

With \mathbf{f} : input features and \mathbf{K} : convolution kernel

And new architectures for images:

Vision transformers, MLP Mixers, ...

Regular grid projections

Image-based approaches

Images are pixels arrays

Implicit neighborhoods

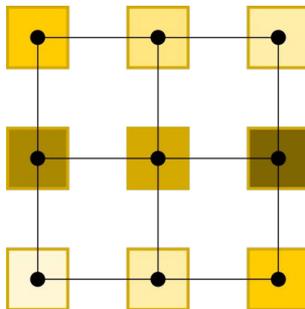
Information is in the color and relative position of the pixels

Thanks to this grid structure

Optimized network architectures

Fast (hardware optimization)

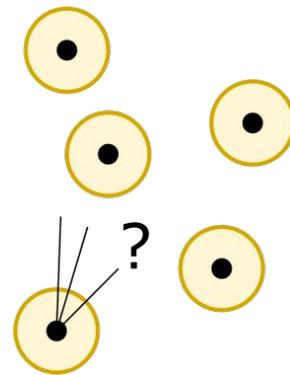
Relatively low memory cost



Point clouds:

~~Implicit neighborhoods~~

~~Information is in the color and relative position of the pixels~~



Idea

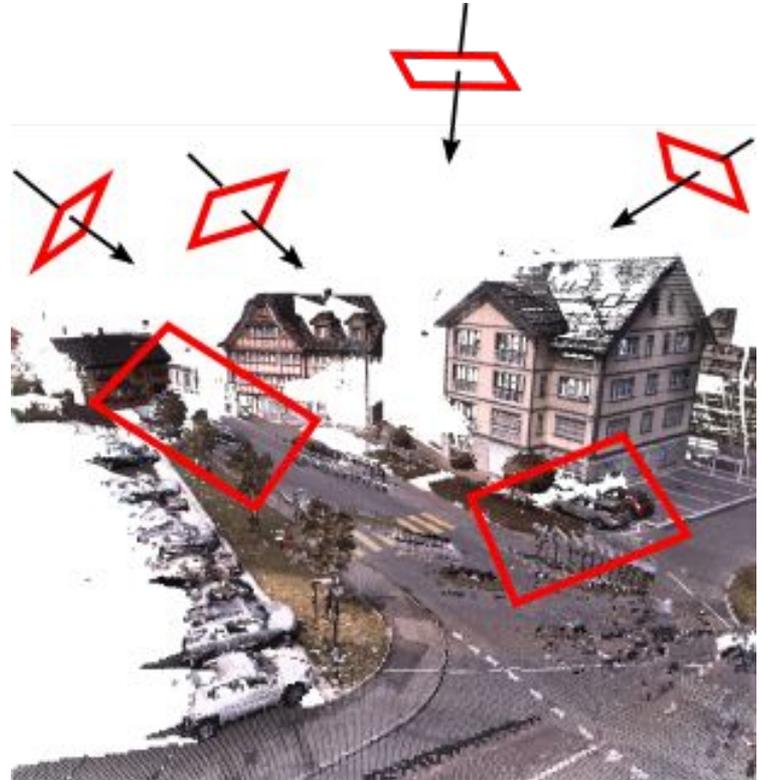
Find a way to create grid data from point cloud

Image projections

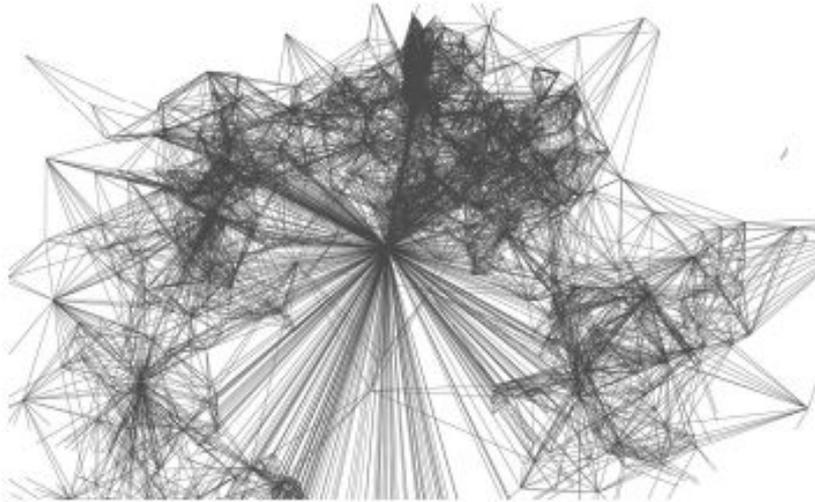
Image-based approaches

Generate images representing the scene

- Use a 3D renderer
- Take virtual snapshots of the scene
- Work in the image



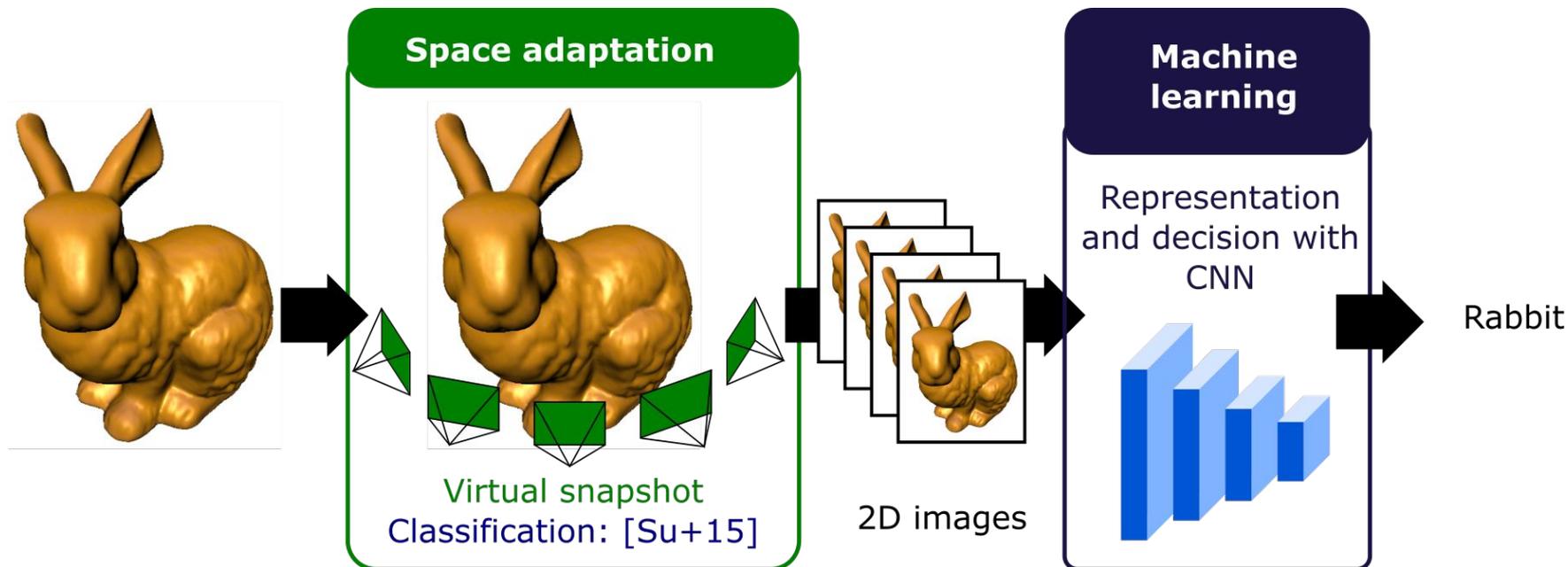
Overview



- I. Image-based approaches
 - A. Principles
 - B. Classification
 - C. Segmentation
- II. Geometric deep learning
 - A. Graph convolution

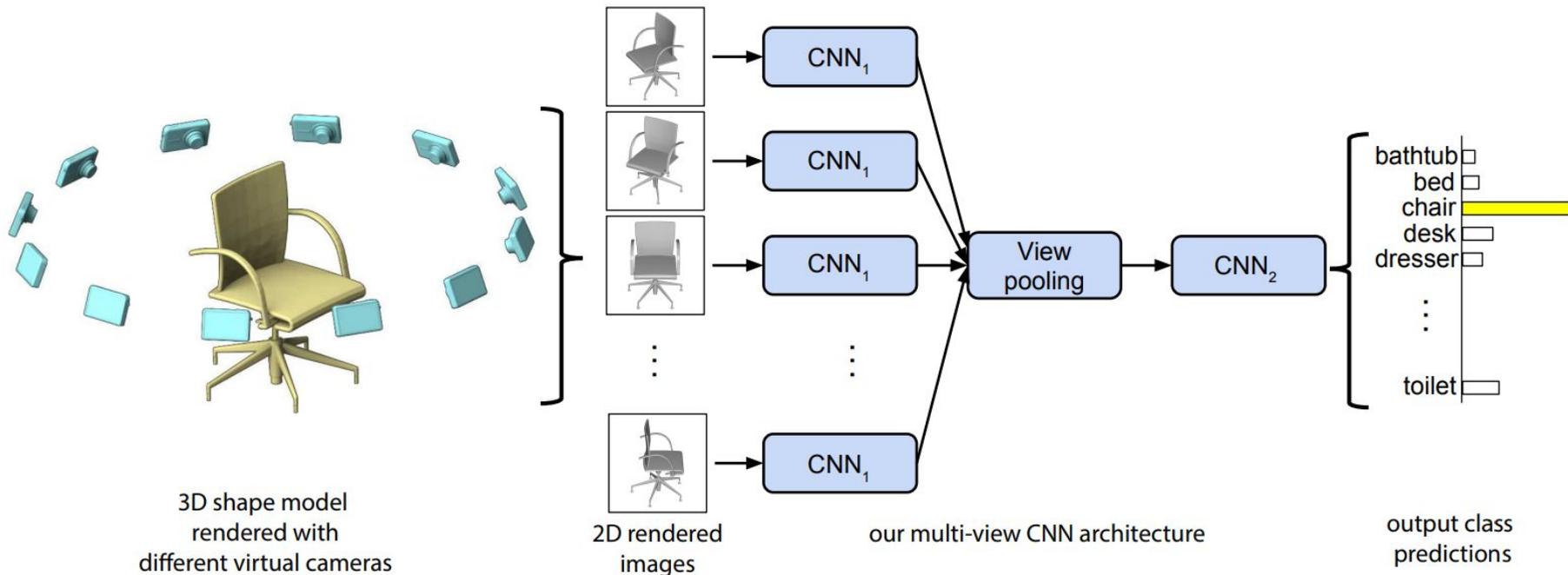
Classification pipeline

Image-based approaches



Classification pipeline

Image-based approaches



Classification

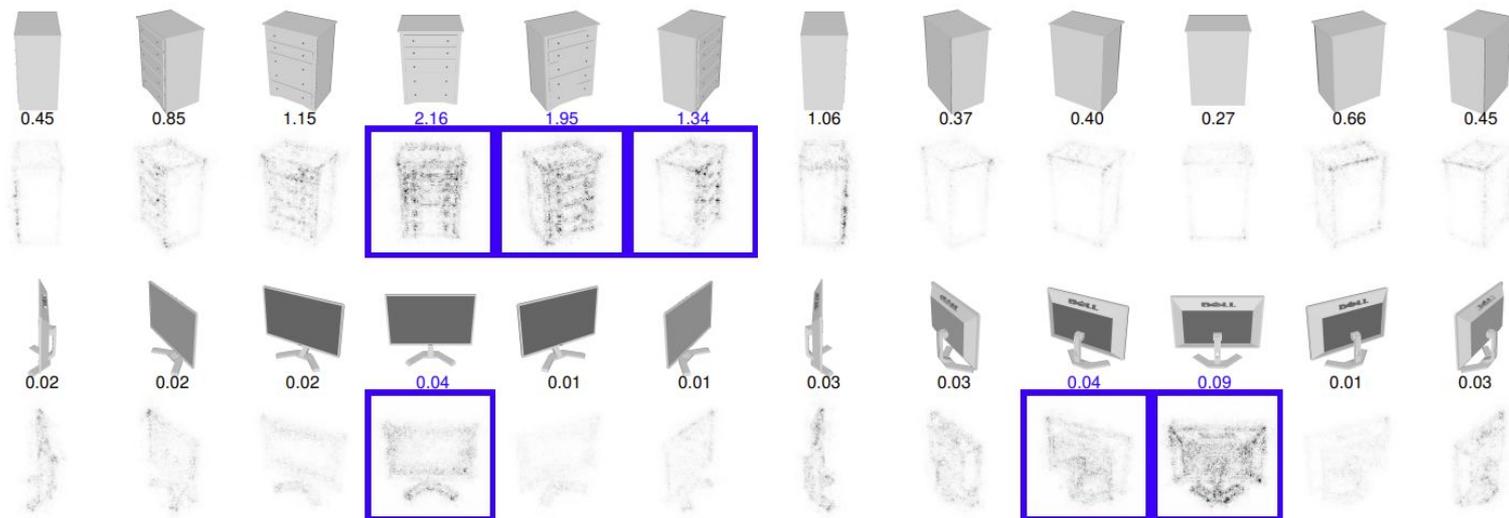
Image-based approaches

Method	Training Config.			Test Config.	Classification (Accuracy)	Retrieval (mAP)
	Pre-train	Fine-tune	#Views	#Views		
(1) SPH [16]	-	-	-	-	68.2%	33.3%
(2) LFD [5]	-	-	-	-	75.5%	40.9%
(3) 3D ShapeNets [37]	ModelNet40	ModelNet40	-	-	77.3%	49.2%
(4) FV	-	ModelNet40	12	1	78.8%	37.5%
(5) FV, 12×	-	ModelNet40	12	12	84.8%	43.9%
(6) CNN	ImageNet1K	-	-	1	83.0%	44.1%
(7) CNN, f.t.	ImageNet1K	ModelNet40	12	1	85.1%	61.7%
(8) CNN, 12×	ImageNet1K	-	-	12	87.5%	49.6%
(9) CNN, f.t., 12×	ImageNet1K	ModelNet40	12	12	88.6%	62.8%
(10) MVCNN, 12×	ImageNet1K	-	-	12	88.1%	49.4%
(11) MVCNN, f.t., 12×	ImageNet1K	ModelNet40	12	12	89.9%	70.1%
(12) MVCNN, f.t.+metric, 12×	ImageNet1K	ModelNet40	12	12	89.5%	80.2%
(13) MVCNN, 80×	ImageNet1K	-	80	80	84.3%	36.8%
(14) MVCNN, f.t., 80×	ImageNet1K	ModelNet40	80	80	90.1%	70.4%
(15) MVCNN, f.t.+metric, 80×	ImageNet1K	ModelNet40	80	80	90.1%	79.5%

* f.t.=fine-tuning, metric=low-rank Mahalanobis metric learning

Classification

Image-based approaches



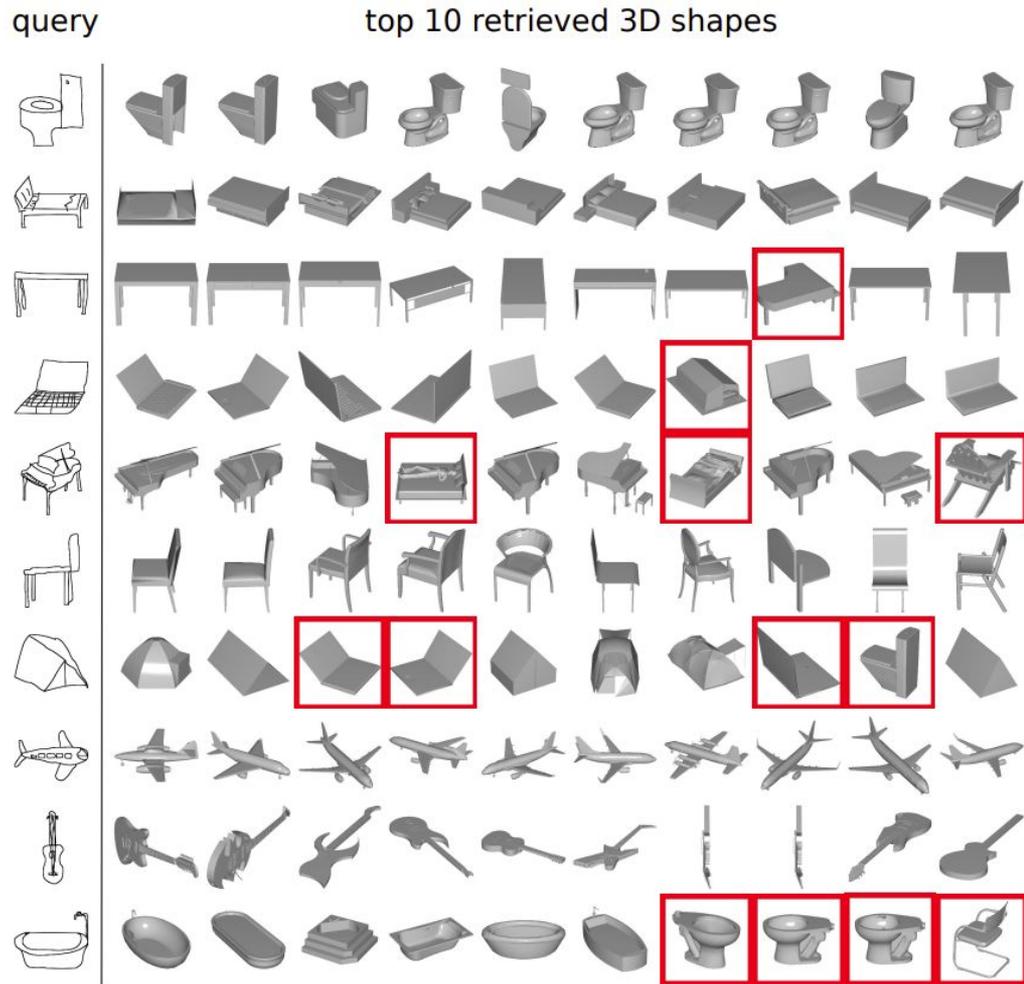
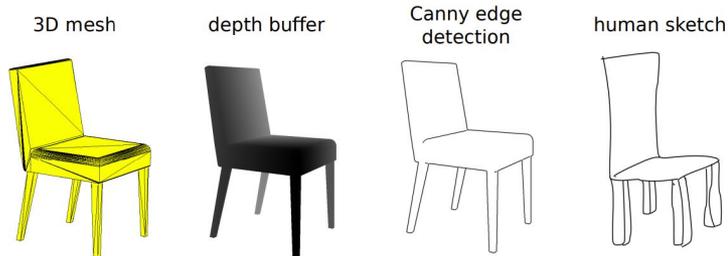
Saliency map (derivative of output w.r.t. the pixel input)

Most discriminative features

Shape retrieval

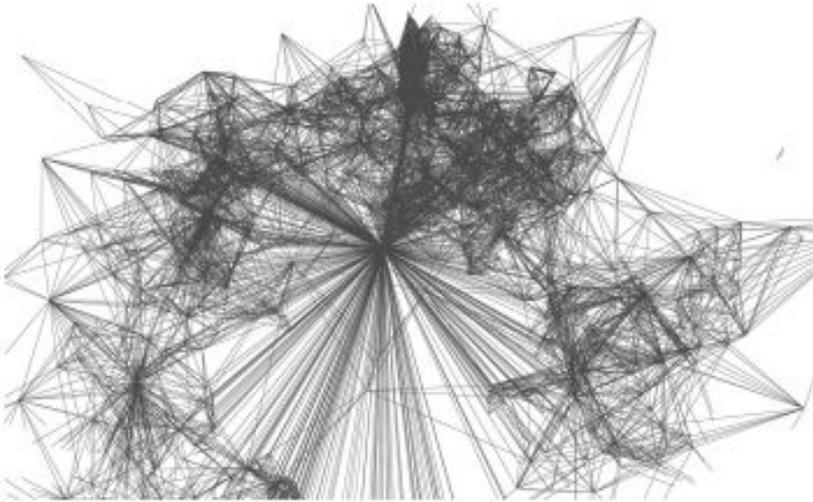
Image-based approaches

Training with different input



Overview

- I. Image-based approaches
 - A. Principles
 - B. Classification
 - C. Segmentation
- II. Geometric deep learning
 - A. Graph convolution

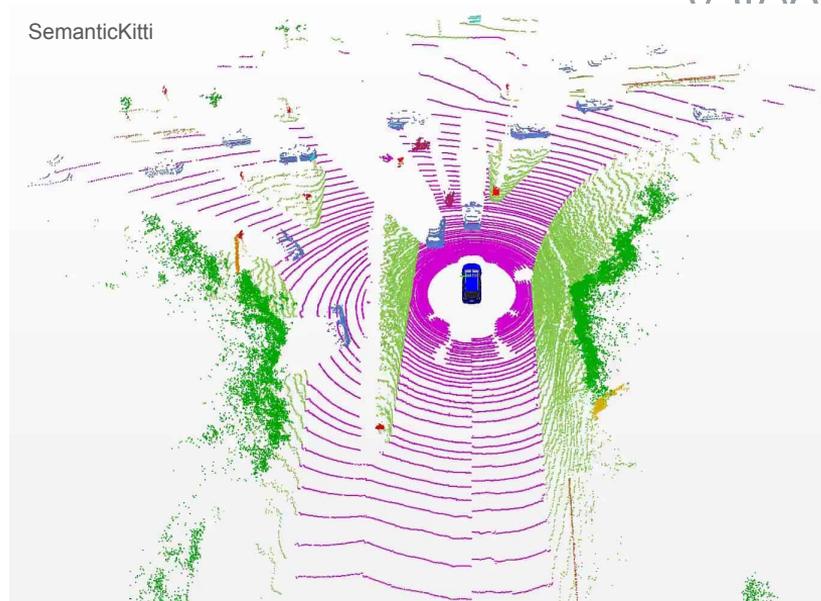


Semantic segmentation

Image-based approaches

One label per point

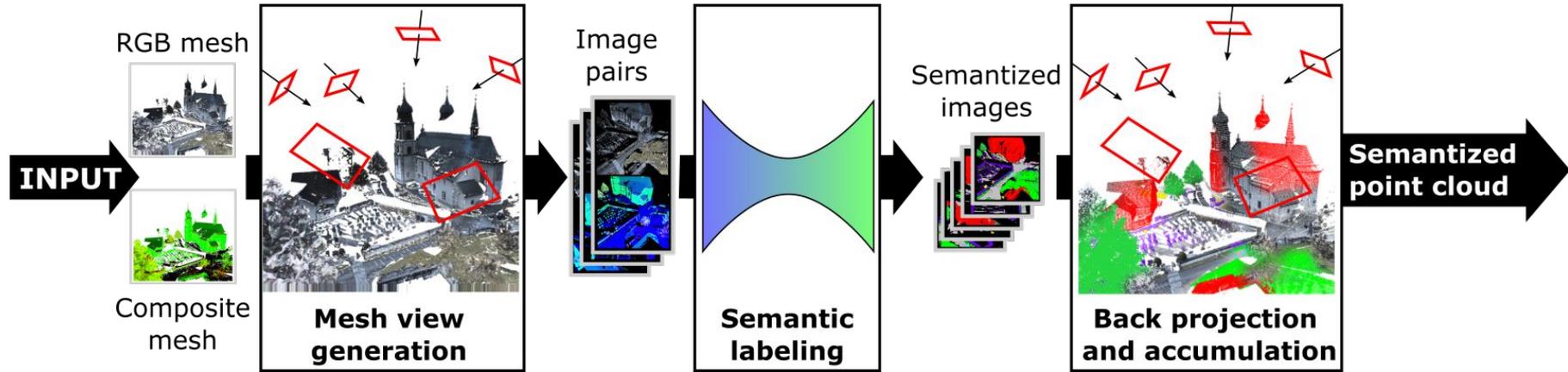
Scannet



ShapeNet part seg

Semantic segmentation pipeline

Image-based approaches



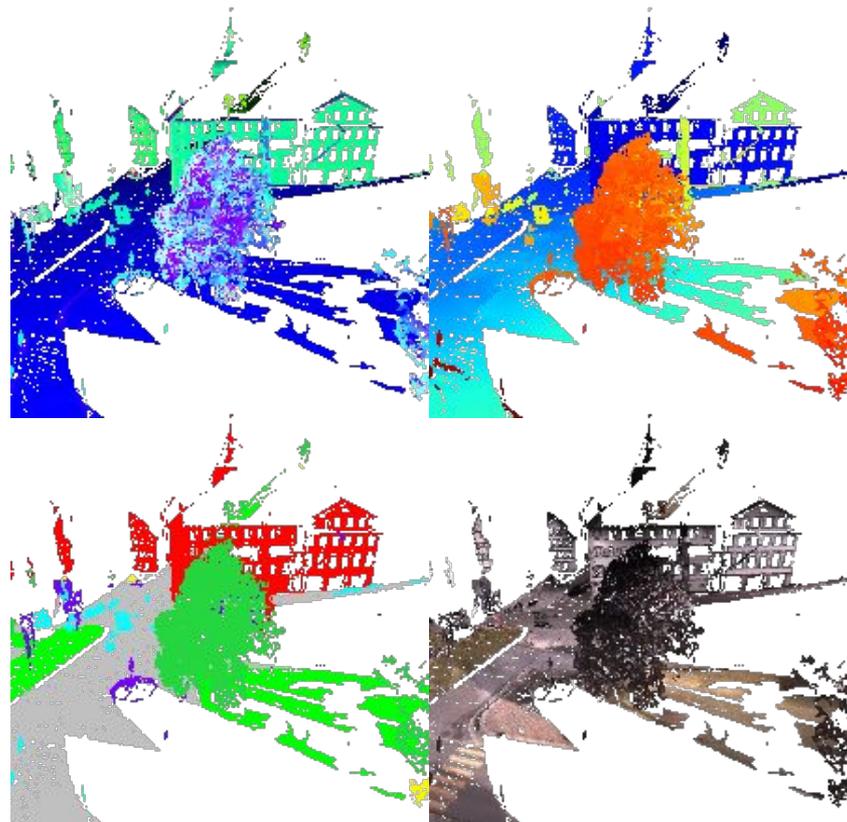
SnapNet

Image-based approaches

Reprojection trick

Generate a snapshot of the scene with fake colors corresponding to point ids

- Allow to generate different snapshots (w / wo colors, geometric features, ground truth at training...)
- Easy reprojection of the results on the original points

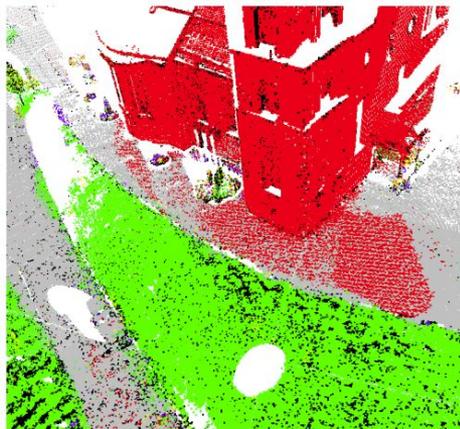


SnapNet

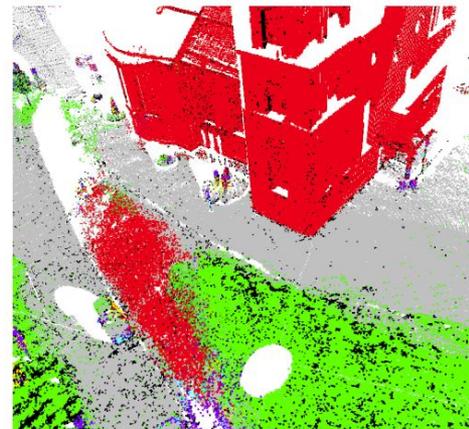
Image-based approaches

Each modality carries different features

→ need for RGB + Geometry



RGB



Composite

SnapNet

Image-based approaches

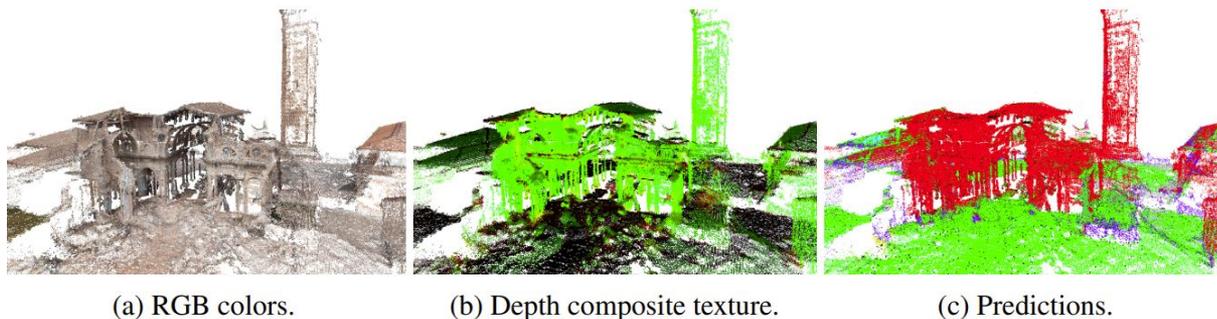
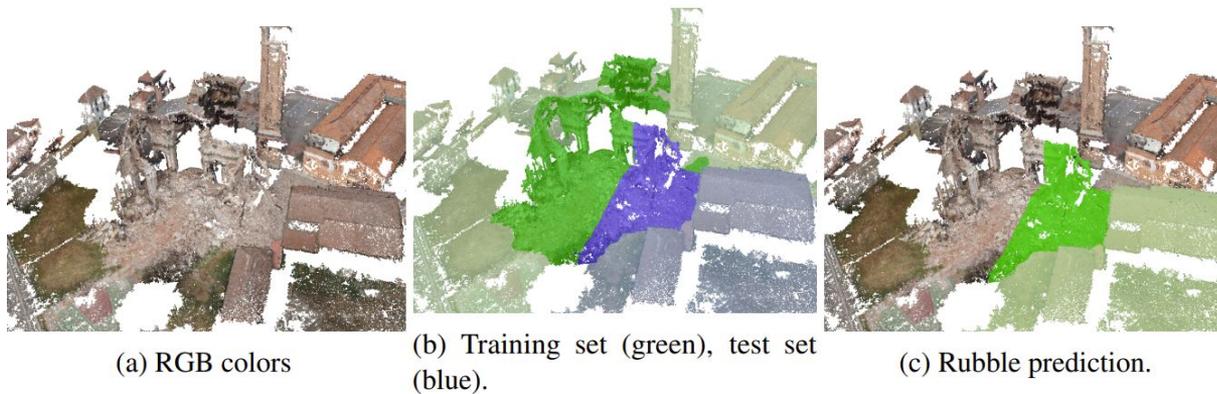


Figure 10: Semantic labeling of photogrammetric data.



SnapNet - advantages and limitations

Image-based approaches

Pros

- Benefit from architectures from image processing
- Use of pre-trained models from large image datasets
- Unlimited number of snapshot a given scene (straightforward data augmentation)

Cons

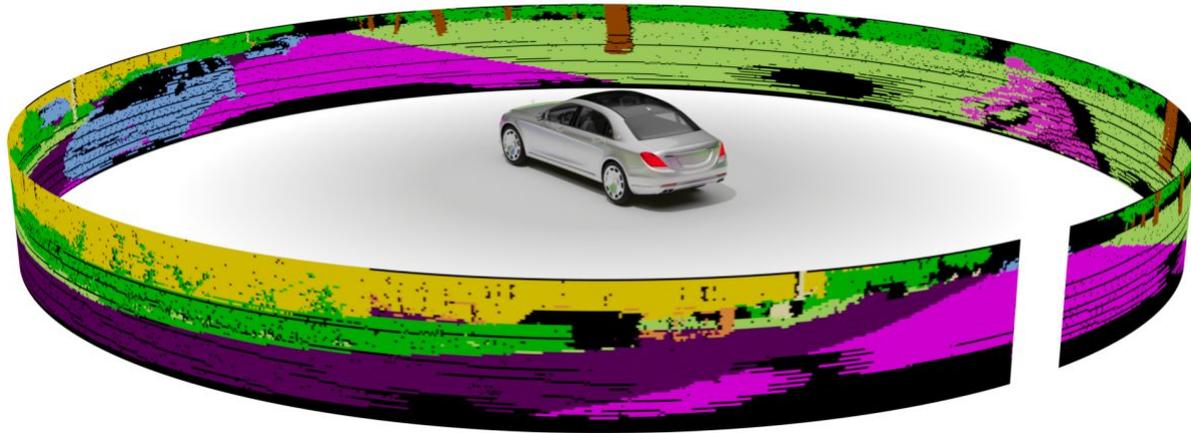
- Good snapshot strategy, which can vary from a dataset to another
- Requires a mesh



Range projection

Image-based approaches

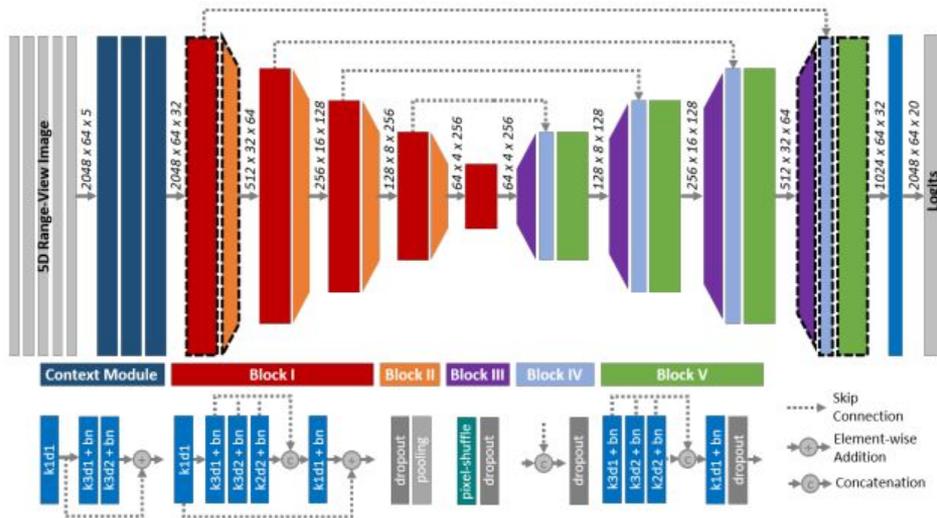
Exploit sensor information to produce images



SalsaNext

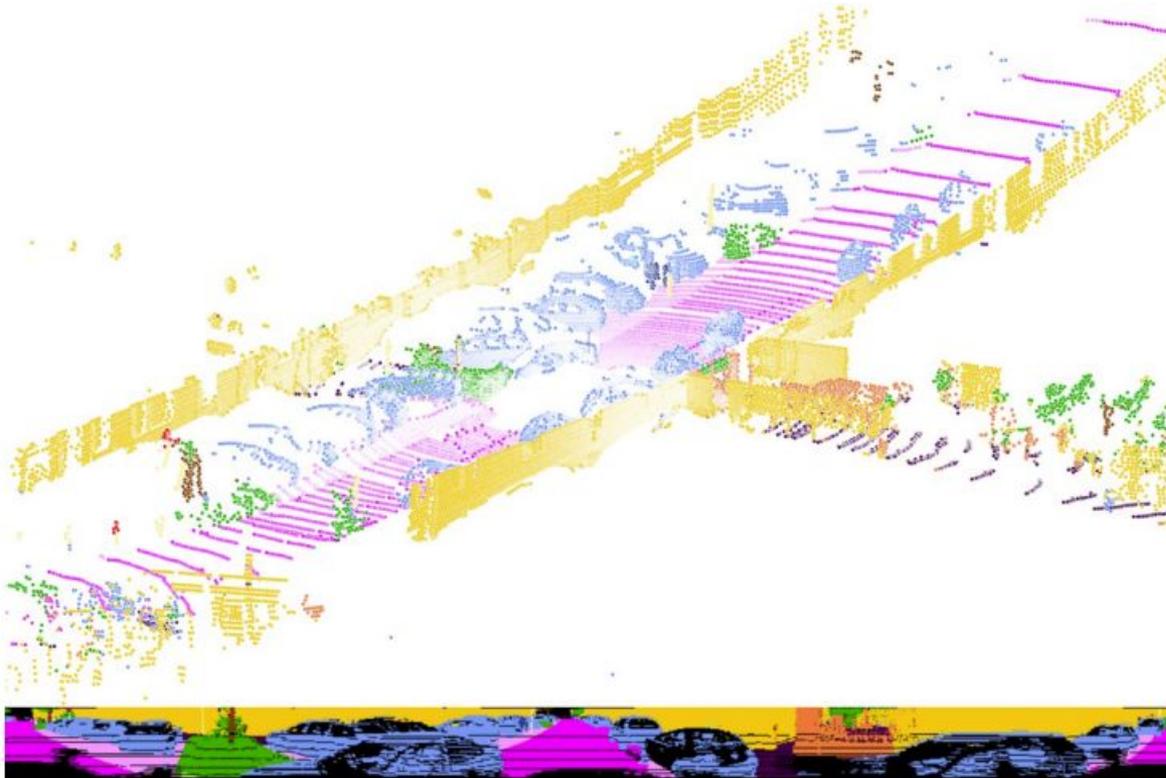
Image-based approaches

Use image backbone (U-Net) for semantic segmentation



SalsaNext

Image-based approaches

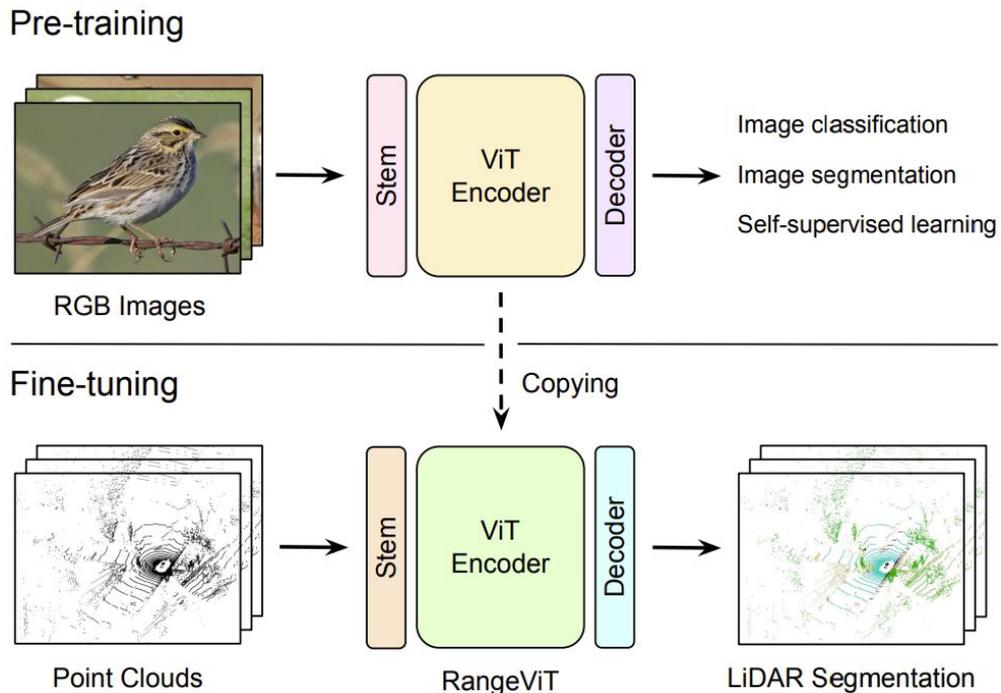


RangeViT

Image-based approaches

LiDAR segmentation based on range images & vision transformers (ViTs)

- Unify architectures in LiDAR and image domain
- Leverage image pre-trained ViTs for LiDAR segmentation

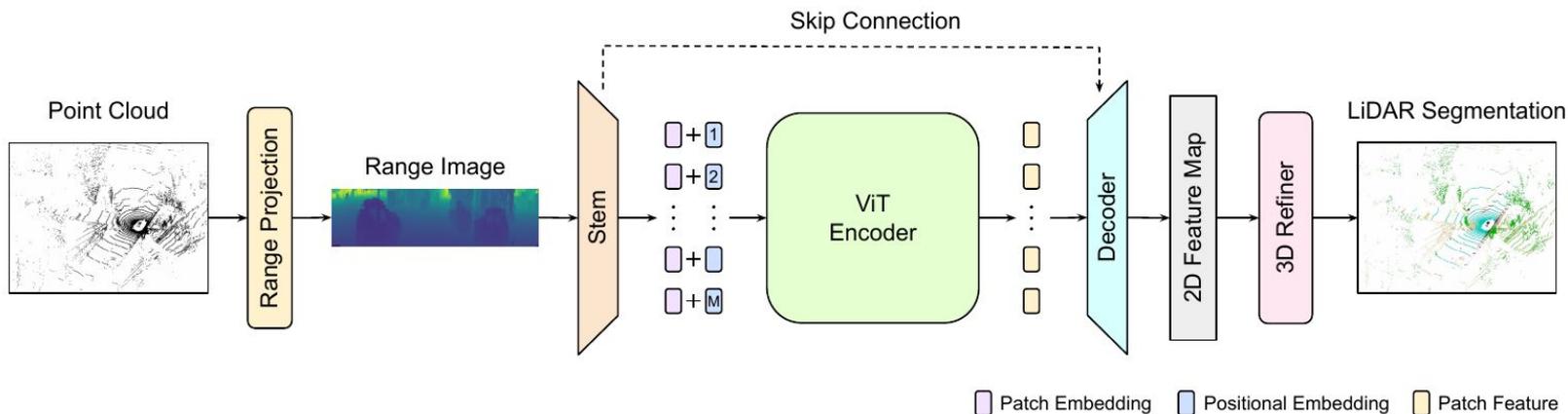


RangeViT

Image-based approaches

Detail of the architecture

Stem + skip connection / 3D refiner



RangeViT

Image-based approaches

Use vision transformers

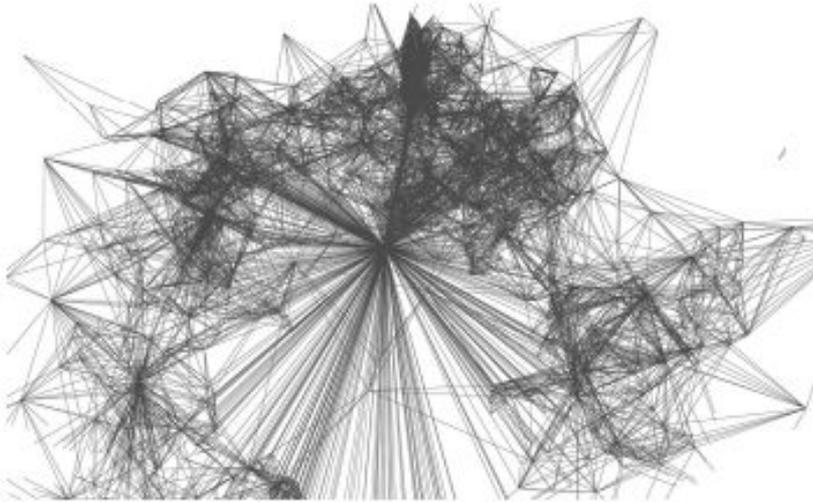
Various initializations

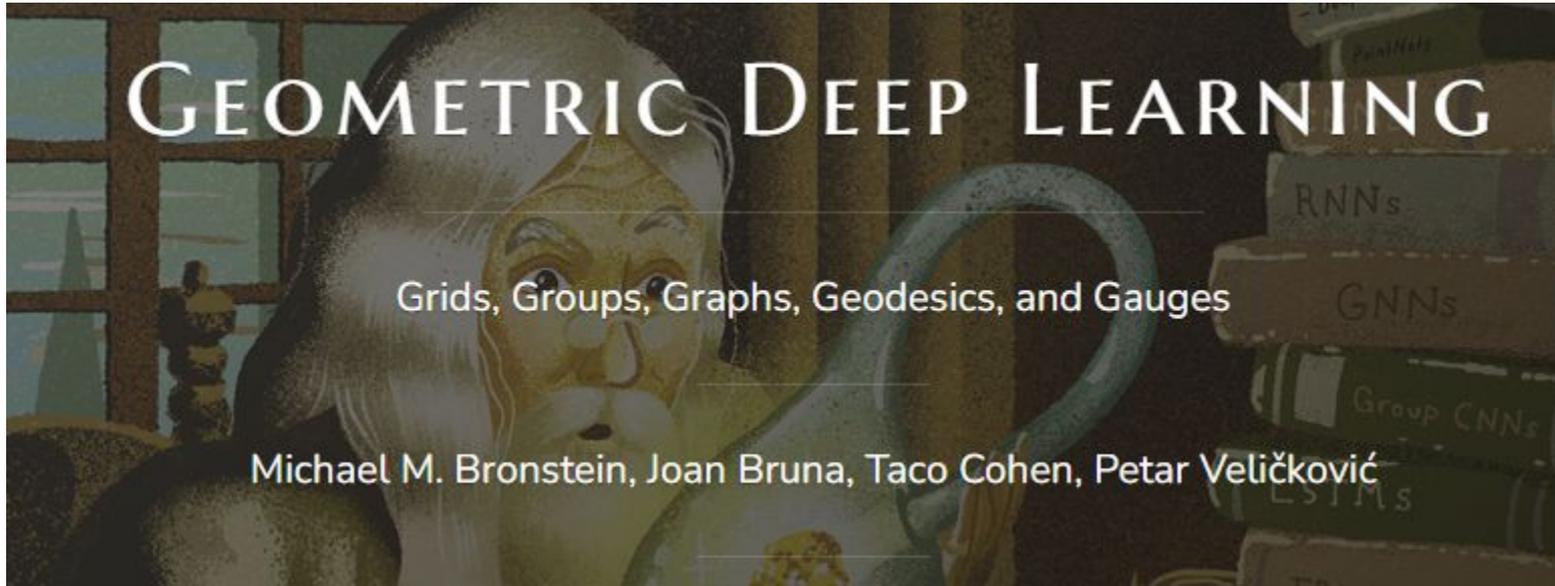
Pre-training	Rand	DINO	IN21k	CS
mIoU	72.37	73.33	74.77	75.21

Method	mIoU (%)
Voxel-based	
Cylinder3D [69]	76.1
2D Projection-based	
RangeNet++ [36]	65.5
PolarNet [66]	71.0
SalsaNext [11]	72.2
RangeViT-IN21k (ours)	74.8
RangeViT-CS (ours)	75.2

Overview

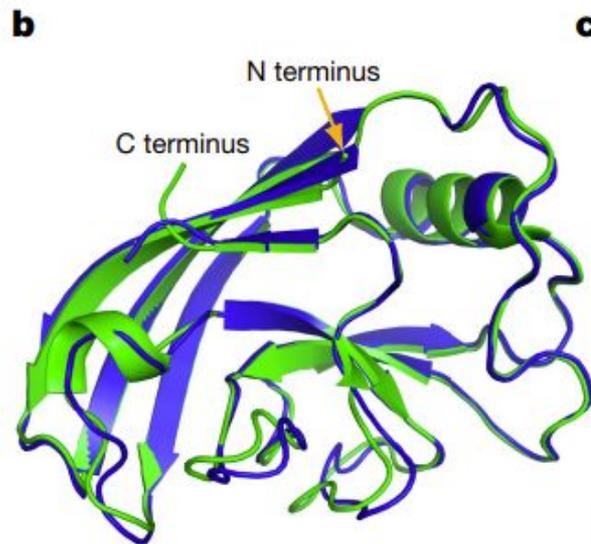
- I. Image-based approaches
 - A. Principles
 - B. Classification
 - C. Segmentation
- II. Geometric deep learning
 - A. Graph convolution



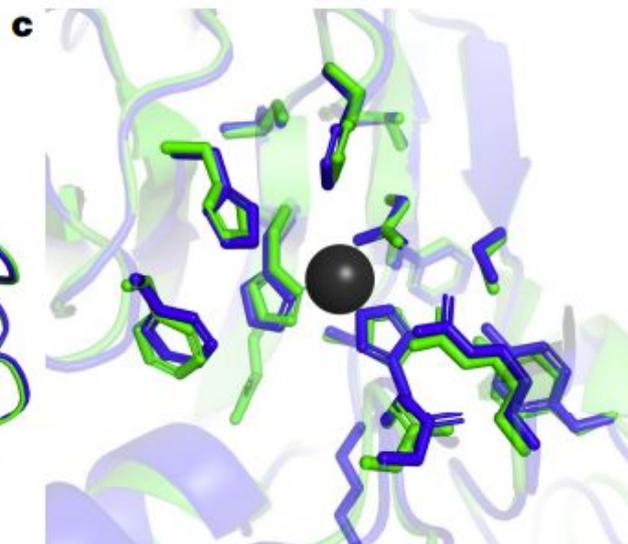


<https://geometricdeeplearning.com/>

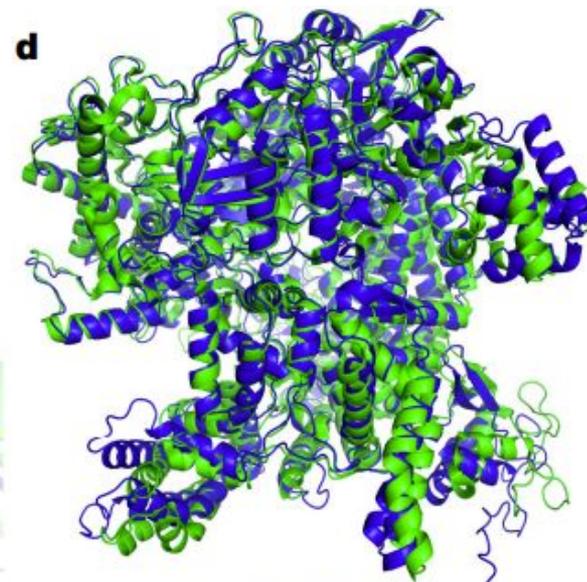
AlphaFold



AlphaFold Experiment
r.m.s.d.₉₅ = 0.8 Å; TM-score = 0.93



AlphaFold Experiment
r.m.s.d. = 0.59 Å within 8 Å of Zn



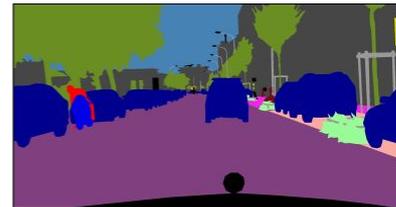
AlphaFold Experiment
r.m.s.d.₉₅ = 2.2 Å; TM-score = 0.96

Geometric deep learning

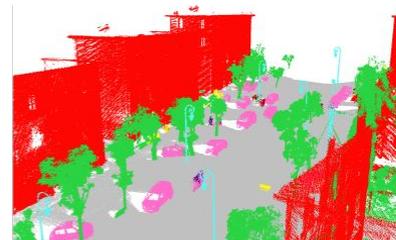
Geometric deep learning



Image
processing



Point cloud
processing



Main problem

Geometric deep learning

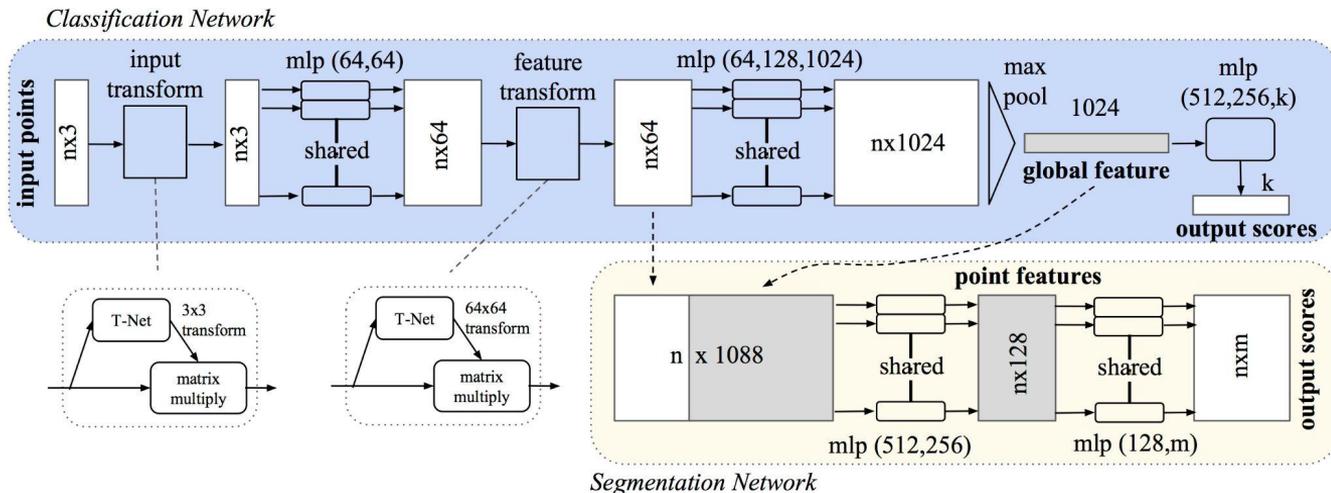
Point cloud are permutation invariant

$$P = [p_1, p_2, \dots, p_i, \dots, p_j, \dots, p_n] \iff P = [p_1, p_2, \dots, p_j, \dots, p_i, \dots, p_n]$$

Neural networks (as usual) are not permutation invariant: swapping features has an influence on the output (e.g., shuffling the pixels of an image)

PointNet

Geometric deep learning



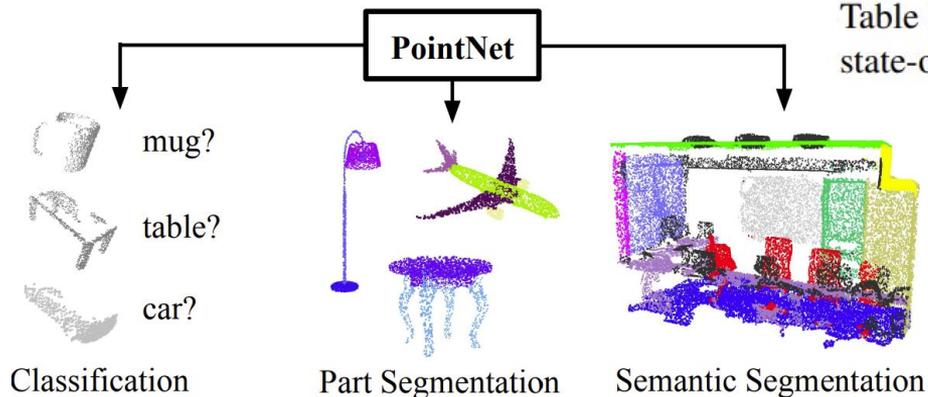
- Use point-wise operation: MLP on feature vector (e.g., coordinates of the points)
- Aggregate with a permutation invariant operation: global pooling (max or average)

PointNet

Geometric deep learning

Multiple applications

- Classification
- Segmentation
- ...



	input	#views	accuracy avg. class	accuracy overall
SPH [11]	mesh	-	68.2	-
3DShapeNets [28]	volume	1	77.3	84.7
VoxNet [17]	volume	12	83.0	85.9
Subvolume [18]	volume	20	86.0	89.2
LFD [28]	image	10	75.5	-
MVCNN [23]	image	80	90.1	-
Ours baseline	point	-	72.6	77.4
Ours PointNet	point	1	86.2	89.2

Table 1. **Classification results on ModelNet40.** Our net achieves state-of-the-art among deep nets on 3D input.

PointNet++

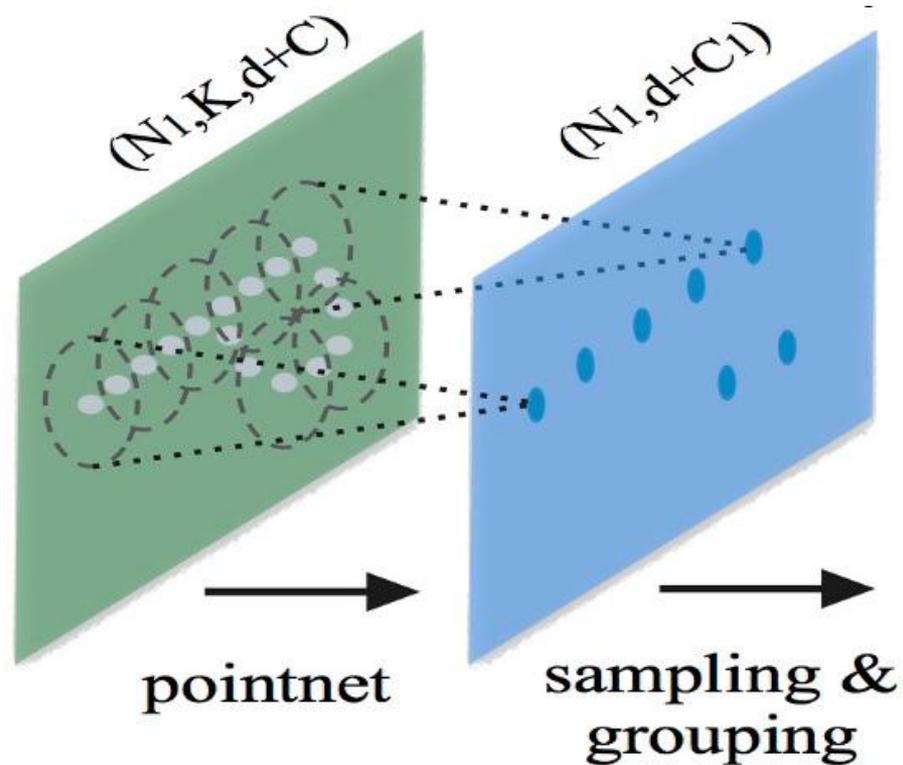
Geometric deep learning

PointNet limitation

Global operations induce a loss in detail analysis

Solution

- Apply local PointNets
- Aggregate results on support points
- Iterate



PointNet++

Geometric deep learning

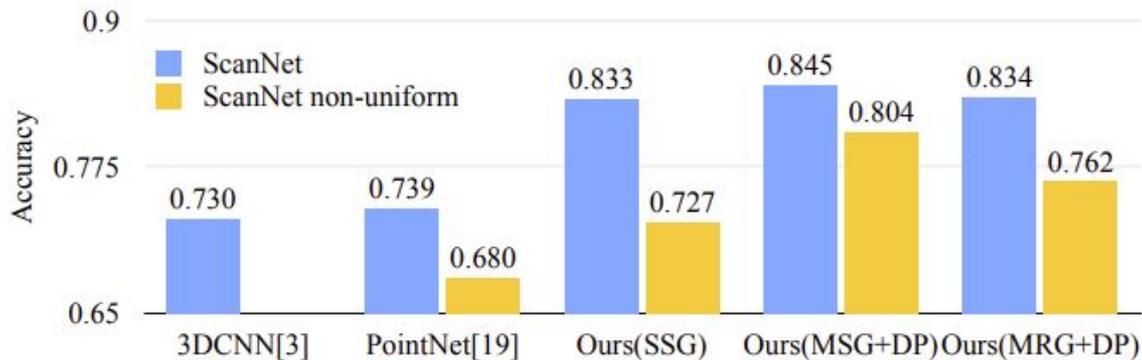
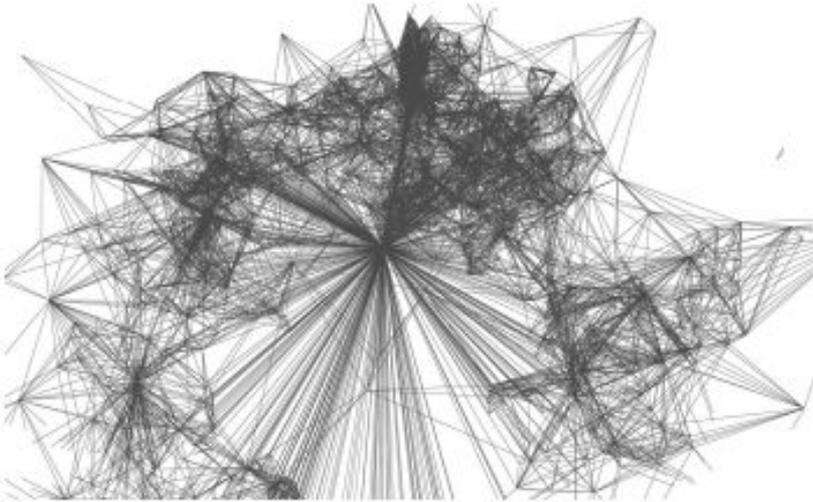


Figure 5: Scannet labeling accuracy.

Big boost, particularly, in semantic segmentation.

Overview

- I. Image-based approaches
 - A. Principles
 - B. Classification
 - C. Segmentation
- II. Geometric deep learning
 - A. Graph convolution

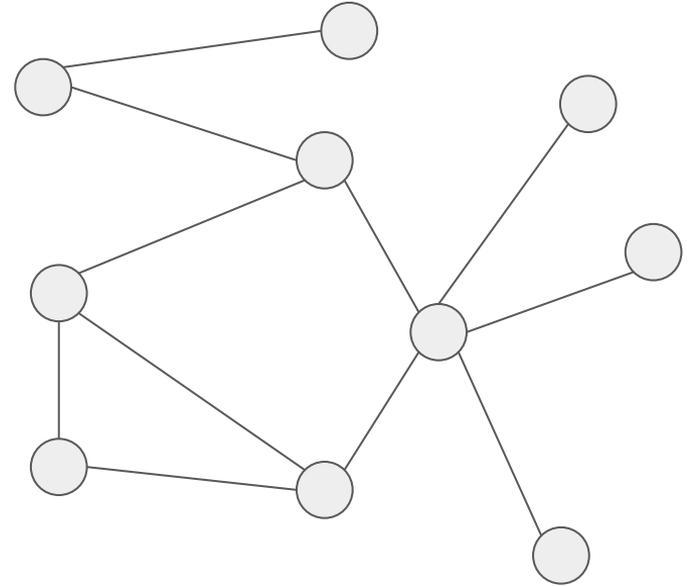


Graph

Graph message passing

Points + edges

Undirected

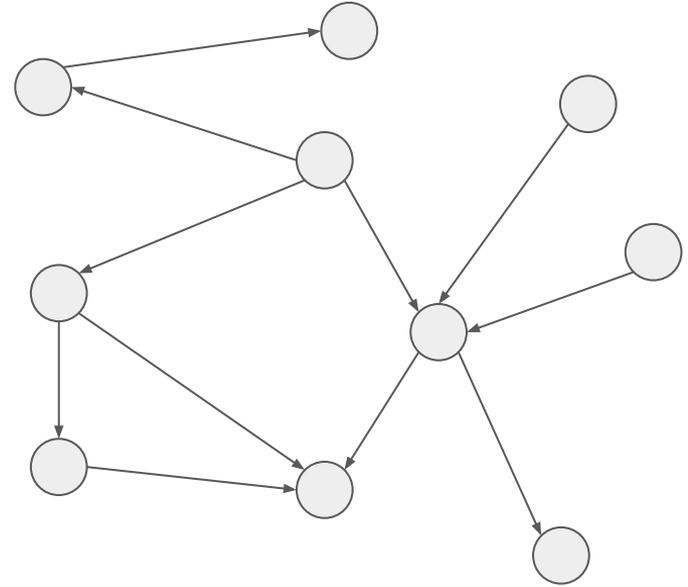


Graph

Graph message passing

Points + edges

Directed



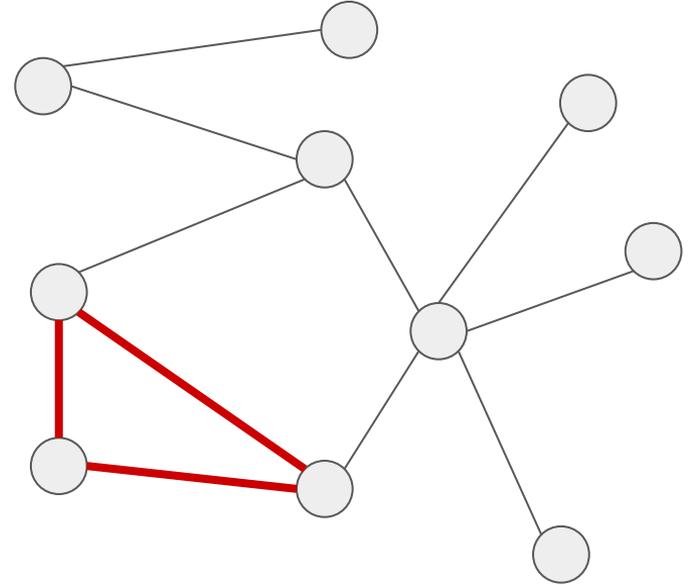
Graph

Graph message passing

Points + edges

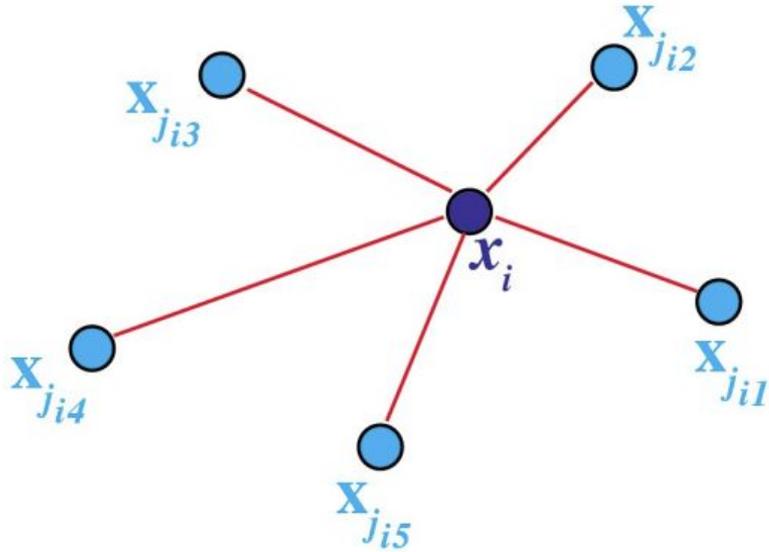
Undirected

Cycles



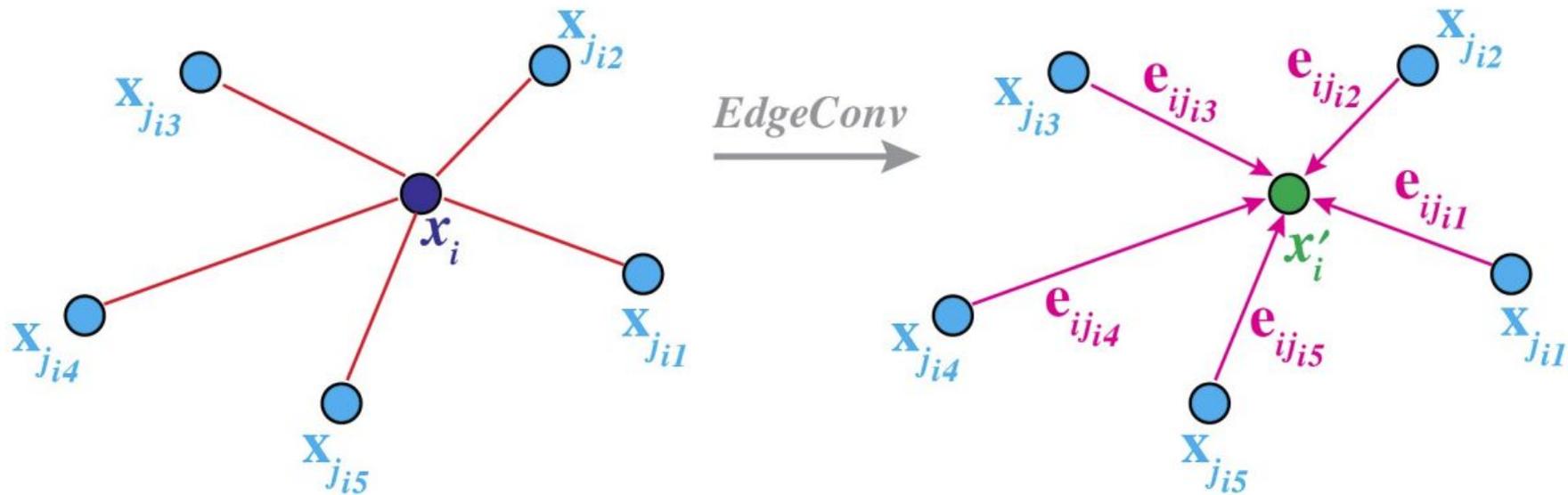
DGCNN

Graph convolution



DGCNN

Graph convolution

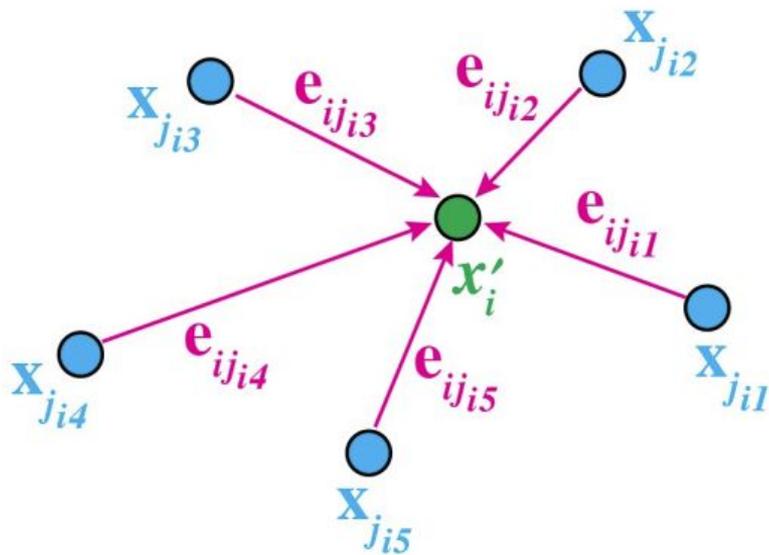


DGCNN

Graph convolution

Similar to images that look at pixel neighborhoods, edge convolution looks at graph neighborhoods.

$$\mathbf{x}'_i = \square_{j:(i,j) \in \mathcal{E}} h_{\Theta}(\mathbf{x}_i, \mathbf{x}_j).$$



PointNet / PointNet++

Graph convolution

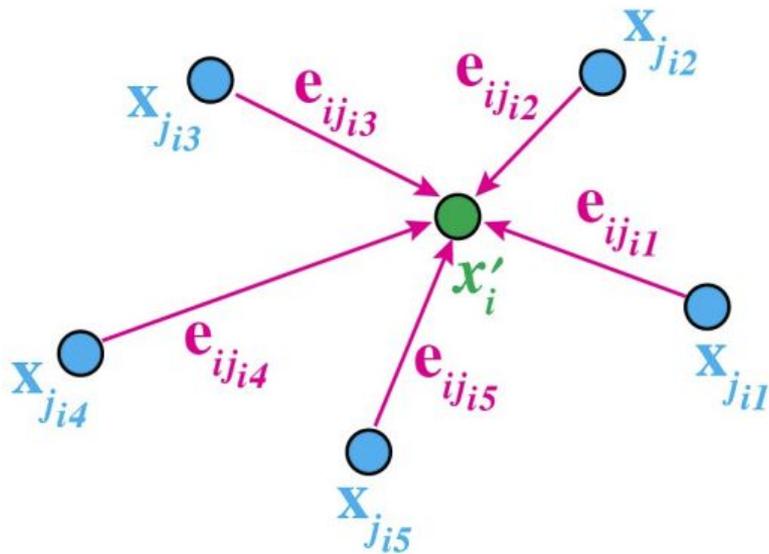
PointNet / PointNet++

$$h_{\Theta}(\mathbf{x}_i, \mathbf{x}_j) = h_{\Theta}(\mathbf{x}_i),$$

And $\square_{j:(i,j) \in \mathcal{E}}$ is a max pooling

for PointNet++ (Null for PointNet)

(The last aggregation is max pooling for all)



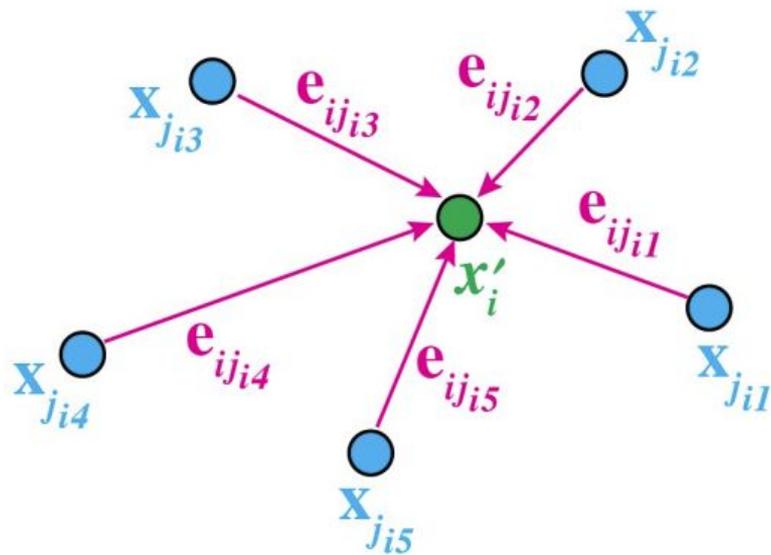
PCNN

Graph convolution

PCNN uses Gaussian kernel an weighting of the sum of the neighbors

$$h_{\Theta}(\mathbf{x}_i, \mathbf{x}_j) = h_{\Theta}(\mathbf{x}_j)$$

$$\mathbf{x}'_{im} = \sum_{j \in \mathcal{V}} (h_{\Theta}(\mathbf{x}_j)) g(u(\mathbf{x}_i, \mathbf{x}_j)),$$



DGCNN

Graph convolution

DGCNN

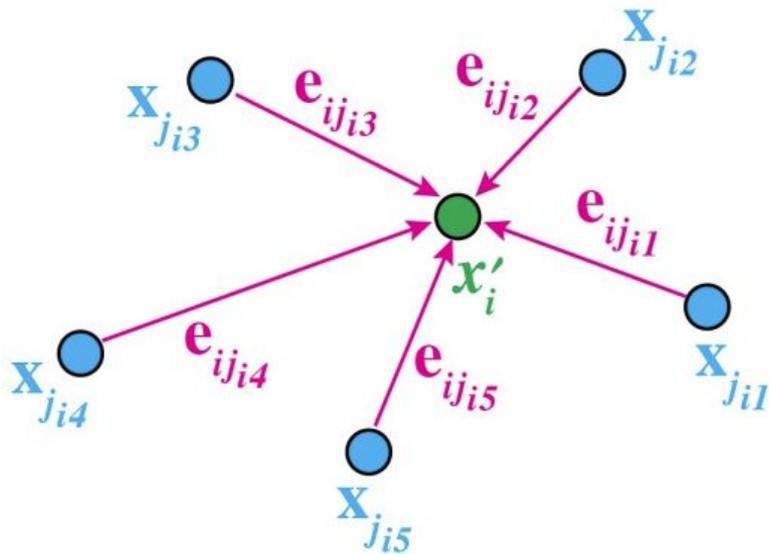
$$h_{\Theta}(\mathbf{x}_i, \mathbf{x}_j) = \bar{h}_{\Theta}(\mathbf{x}_i, \mathbf{x}_j - \mathbf{x}_i).$$

Then (in practice we also put a batchnorm)

$$e'_{ijm} = \text{ReLU}(\theta_m \cdot (\mathbf{x}_j - \mathbf{x}_i) + \phi_m \cdot \mathbf{x}_i),$$

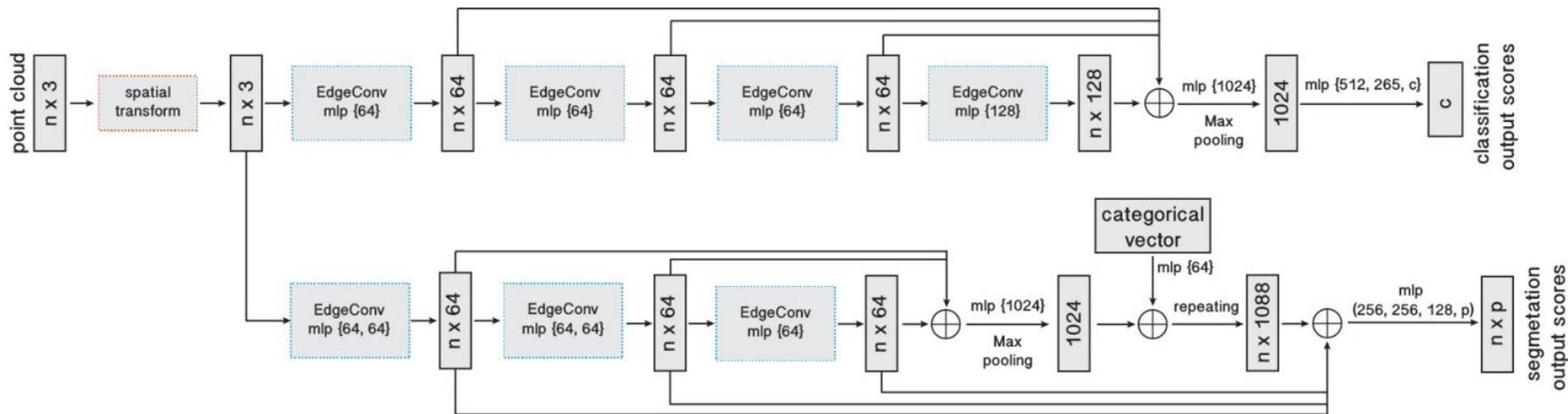
and

$$x'_{im} = \max_{j:(i,j) \in \mathcal{E}} e'_{ijm},$$



Design CNN

Graph convolution

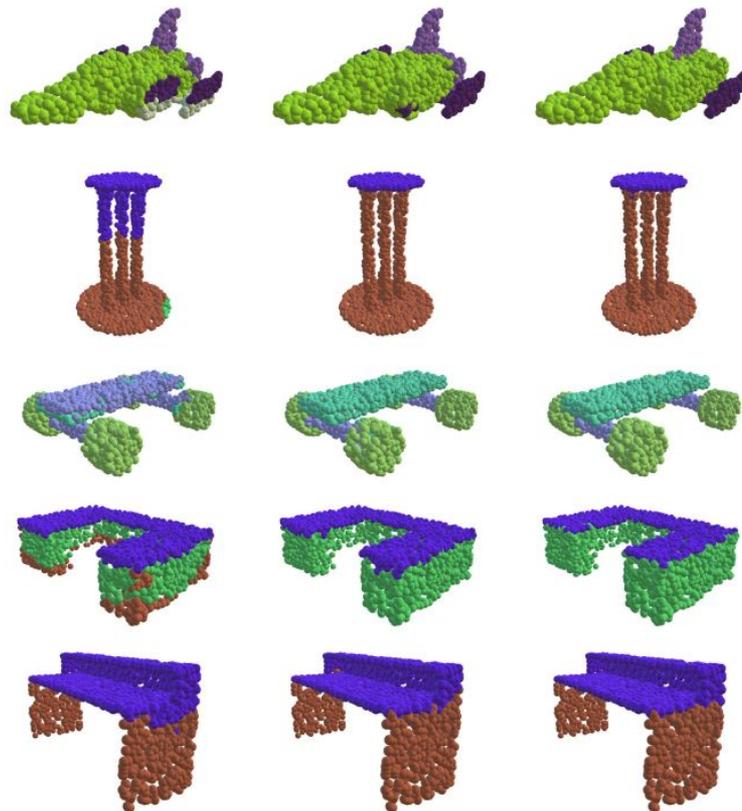


Design CNN

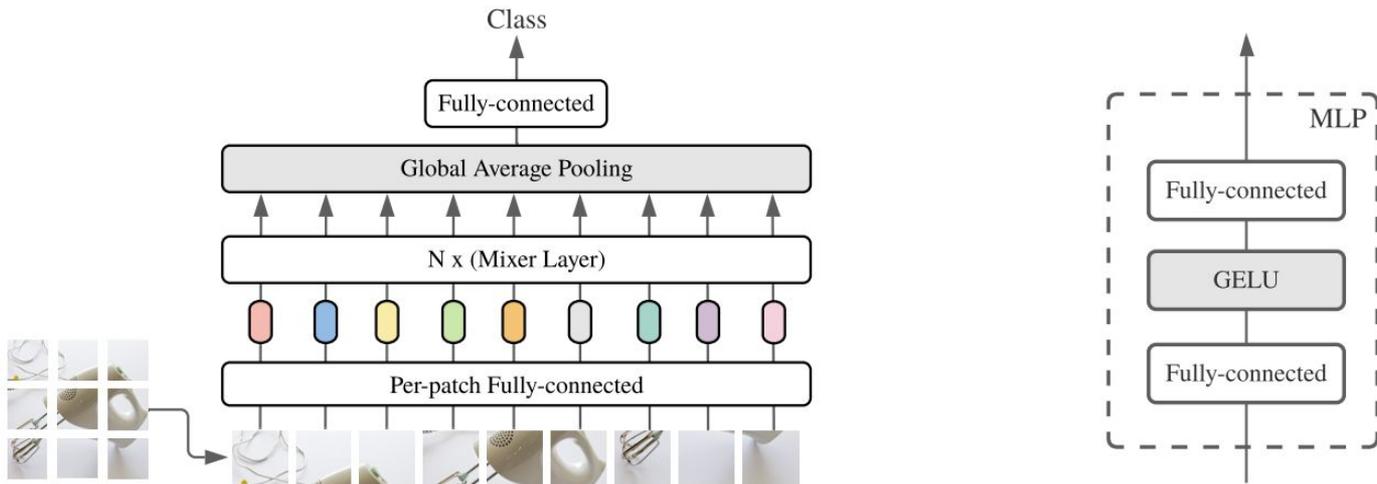
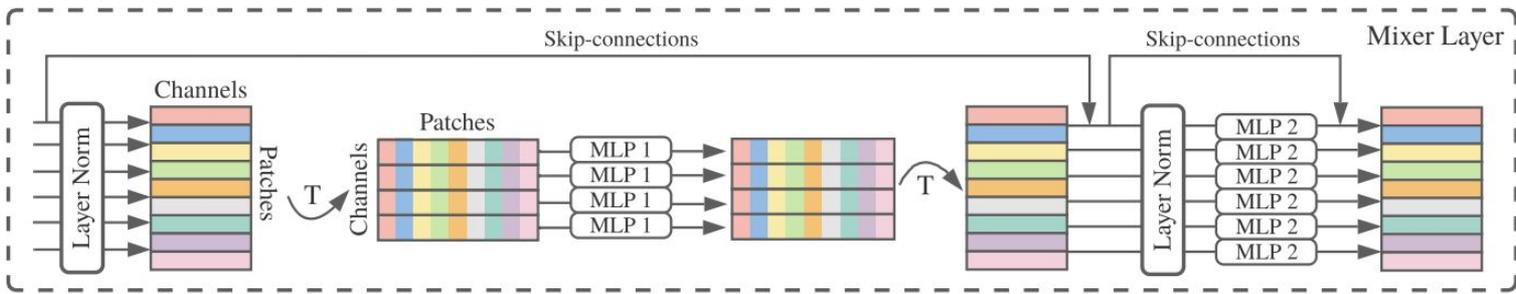
Graph convolution

Local features are better recognized than with only global pooling (PointNet)

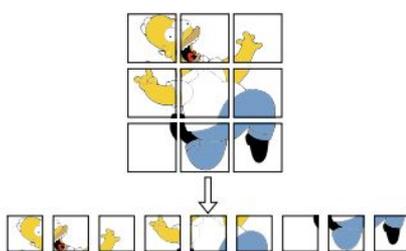
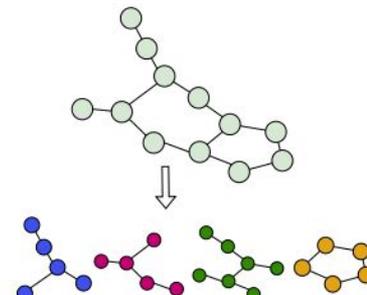
Better performances in segmentation (classification also).



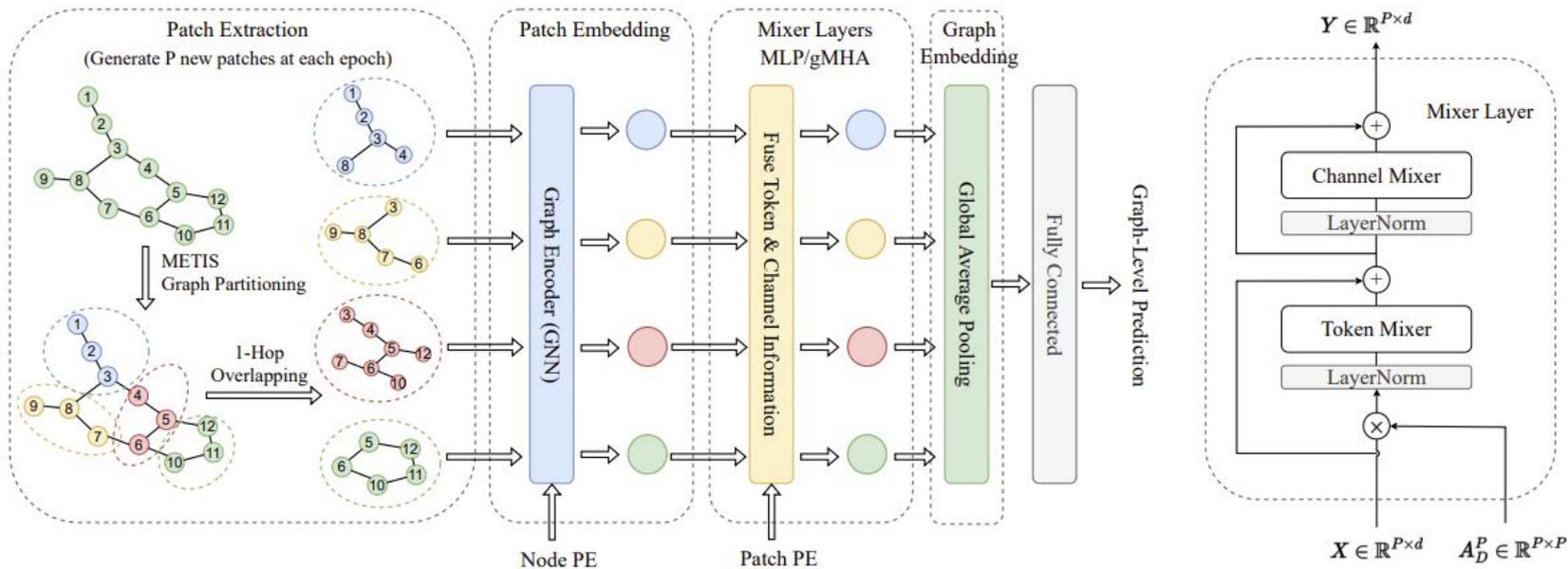
Mixers on graph



Mixers on graph

	Images	Graphs
		
Input	Regular grid Same data resolution (Height, Width)	Irregular domain Variable data structure (# Nodes and # Edges)
Patch Extraction	Via pixel reordering Non-overlapping patches Same patches at each epoch	Via graph clustering algorithm Overlapping patches Different patches at each epoch
Patch Encoder	Same patch resolution (Patch Height, Patch Width) MLP (equivalently CNN)	Variable patch structure (# Nodes and # Edges) GNN (e.g. GCN, GAT, GT)
Positional Information	Implicitly ordered (No need for explicit PE)	No universal ordering Node PE for patch encoder Patch PE for token mixer
ViT / MLP-Mixer	MLP / Channel mixer MHA / Token mixer	MLP / Channel mixer gMHA / Token mixer

Mixers on graph



He et al., A Generalization of ViT/MLP-Mixer to Graphs, ICML, 2023.

Super Point Graph



(a) RGB point cloud



(b) Geometric partition

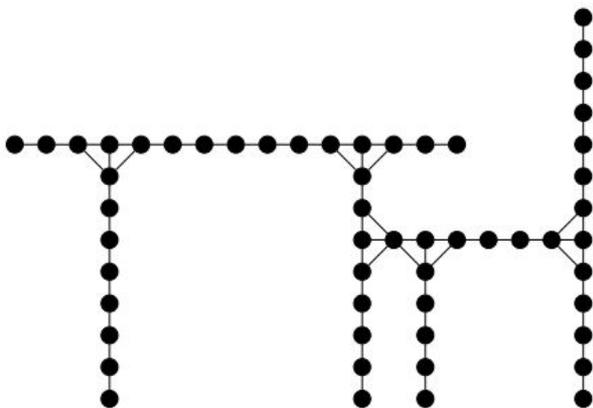


(c) Superpoint graph

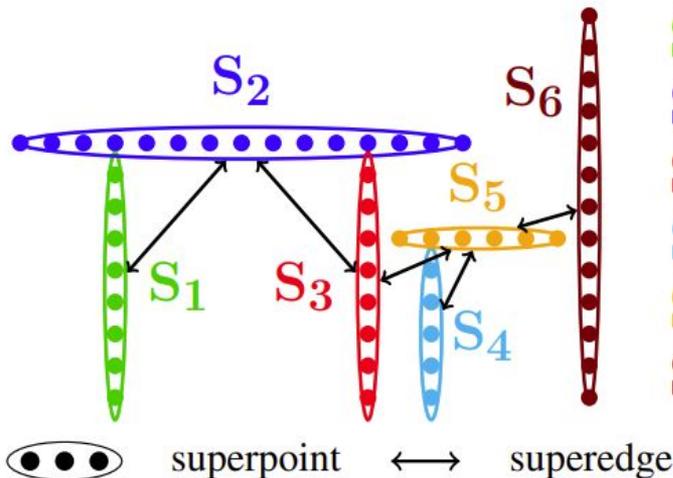


(d) Semantic segmentation

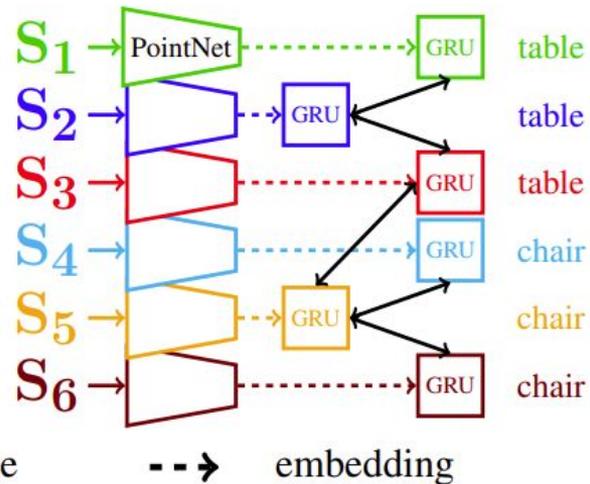
Super Point Graph



(a) Input point cloud

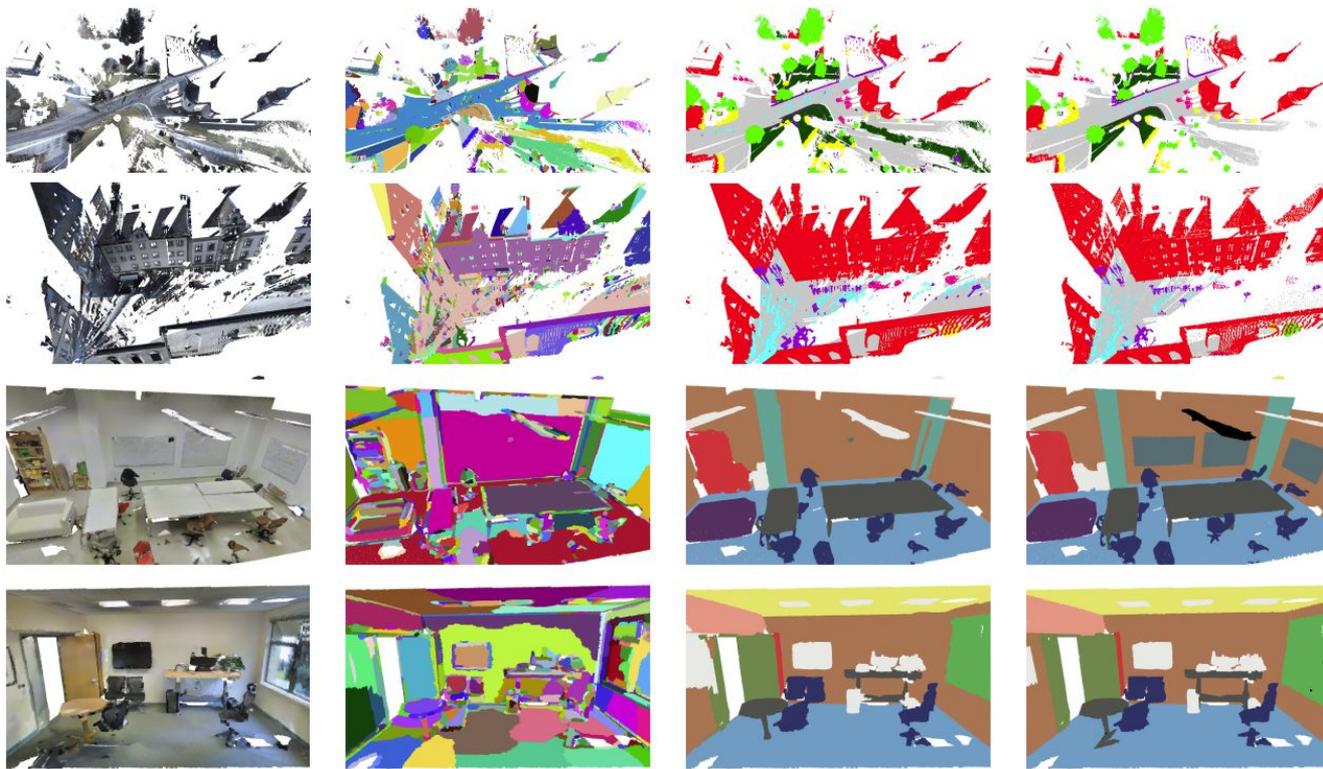


(b) Superpoint graph



(c) Network architecture

Super Point Graph



(a) RGB point cloud

(b) Geometric partitioning

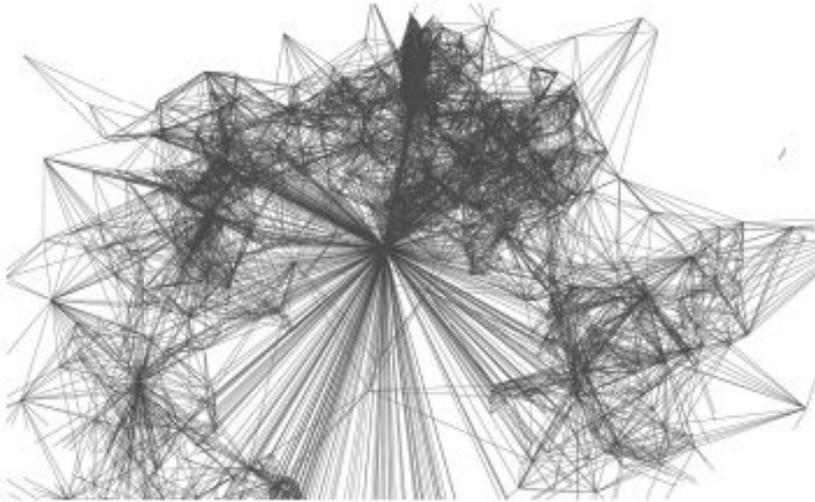
(c) Prediction

(d) Ground truth

Overview

- I. Image-based approaches
 - A. Principles
 - B. Classification
 - C. Segmentation
- II. Geometric deep learning
 - A. Graph convolution

Conclusion



What we didn't talk about

Image structure

- ++ efficient / benefit from image knowledge
- ++ if an easy image rendering
- - - rendering can be hard

Pure graph structure

- ++ very high performances
- ++ evolution toward graph attention (AlphaFold)
- - - when the graph structure is not obvious, additional computing

Practical session

First neural network training

Classification on ModelNet40

→ PointNet (light)

→ DGCNN (light)

We still run them CPU !