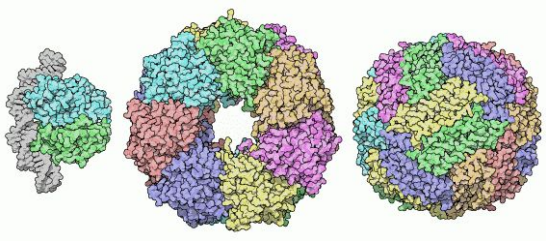
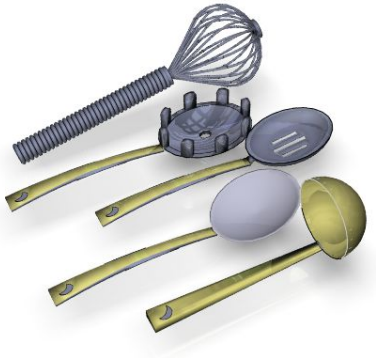
جامعة الملك عبدالله
للعلوم والتقنية
King Abdullah University of
Science and Technology

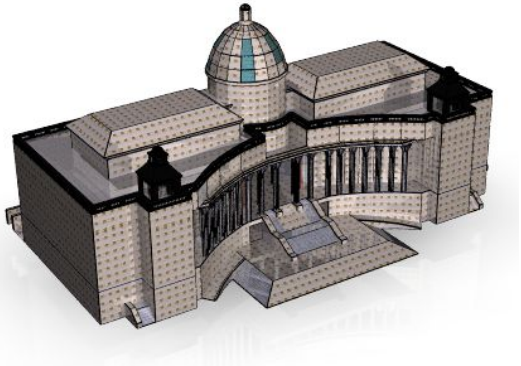
3D Deep Learning

PROF. BERNARD GHANEM

We live in a 3D world





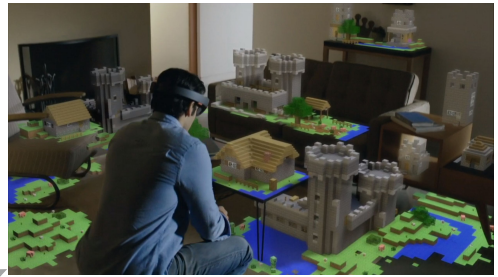


Broad Application of 3D data

Robotics

Broad Application of 3D data

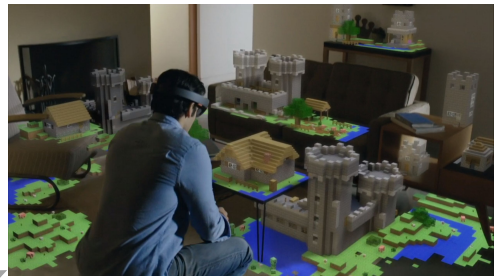
Robotics



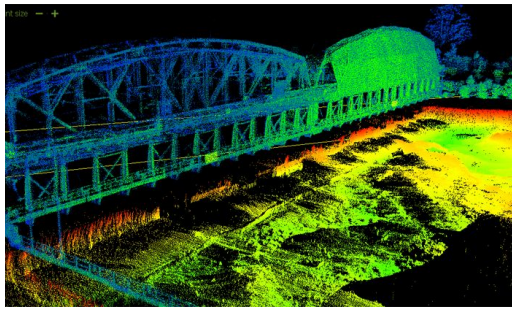
Augmented Reality

Broad Application of 3D data

Robotics



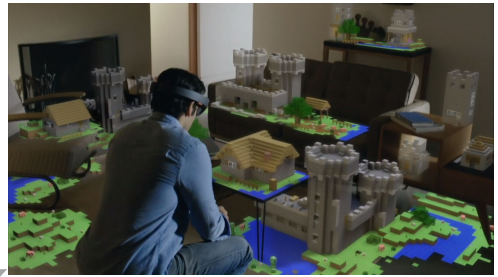
Augmented Reality



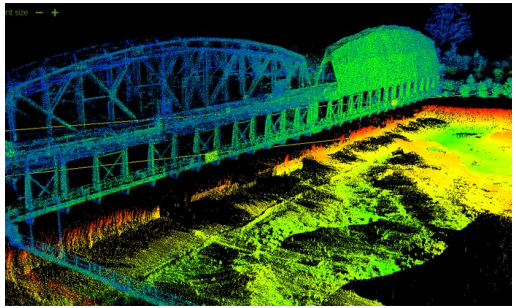
Autonomous driving

Broad Application of 3D data

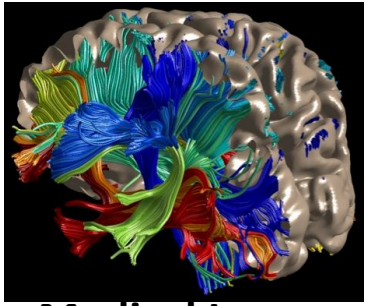
Robotics



Augmented Reality

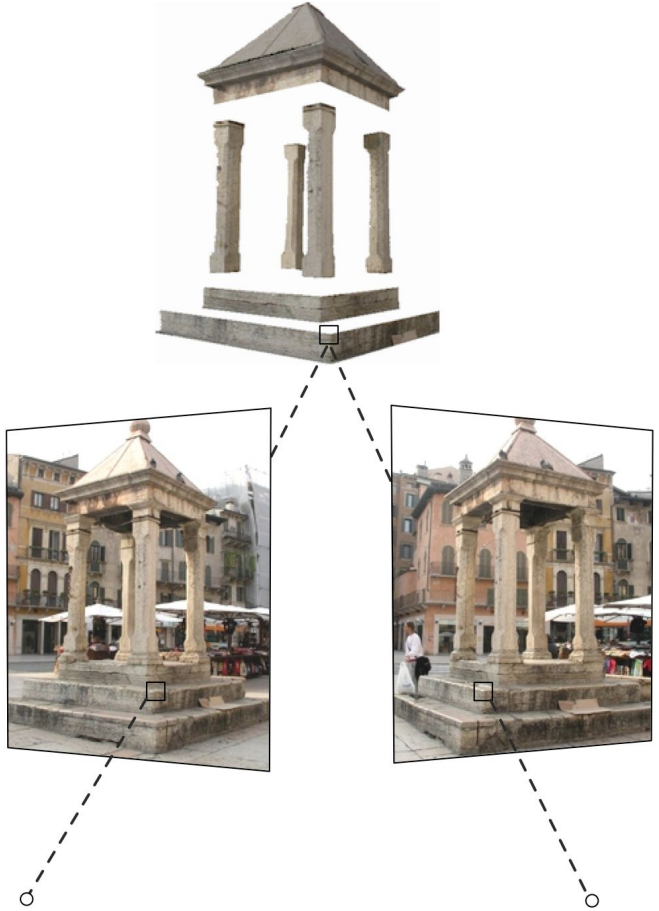


Autonomous driving



Medical Image Processing

Traditional 3D Vision: Physics based



Multi-view Geometry

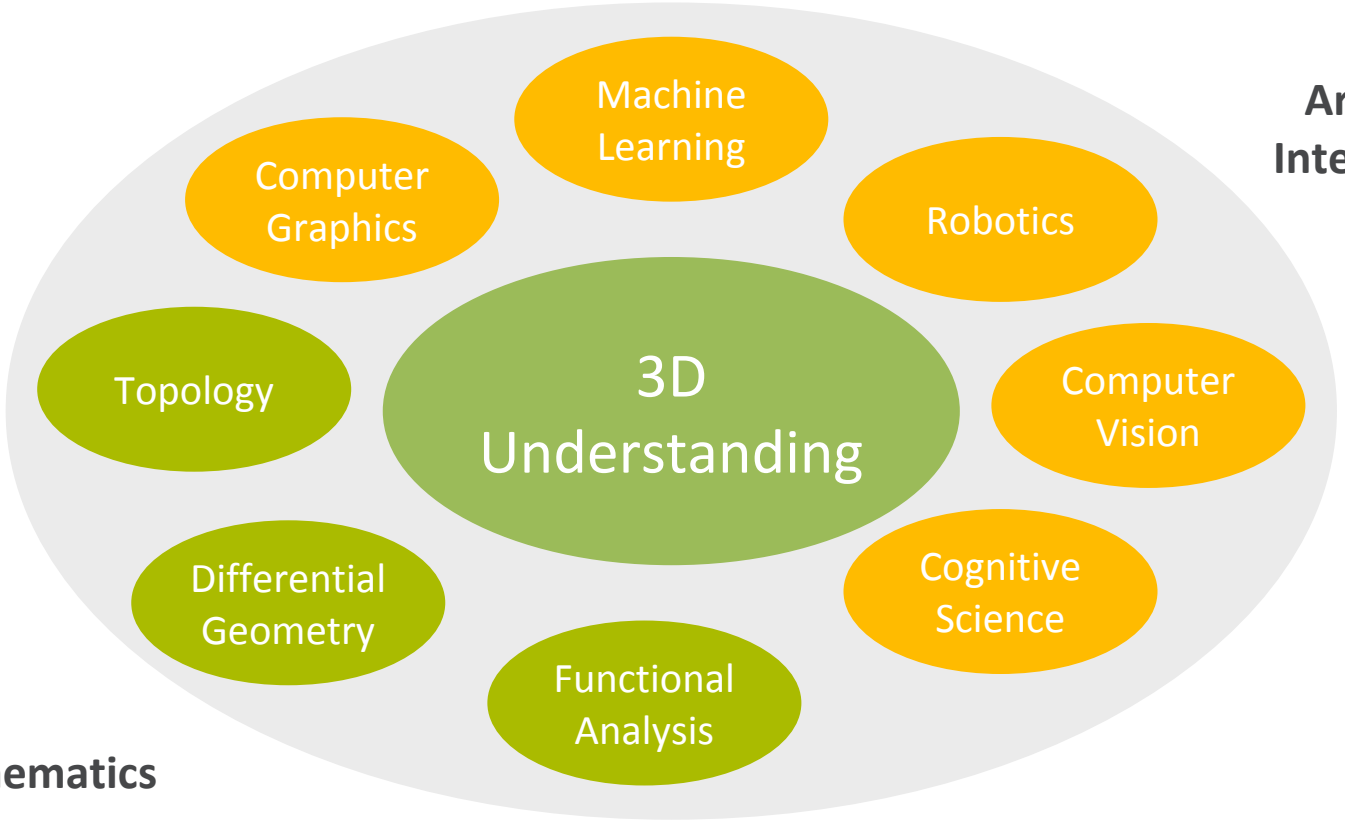
3D Learning: Knowledge Based

3D Learning: Knowledge Based

Acquire Knowledge of 3D World by Learning

A priori knowledge of
the 3D world

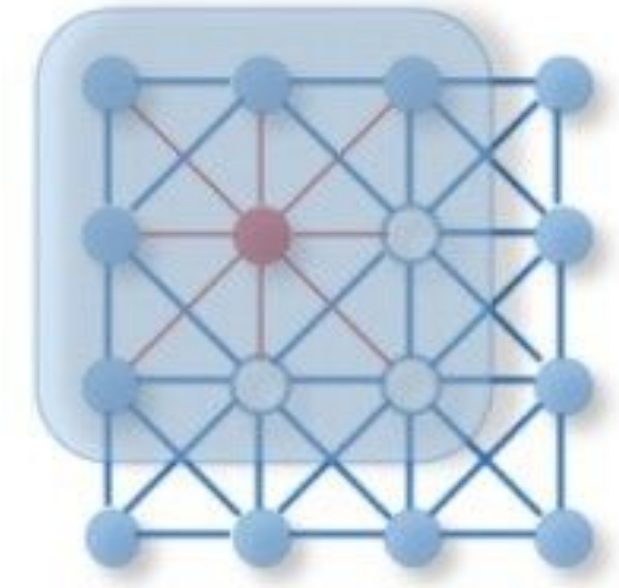
A New Rising Field



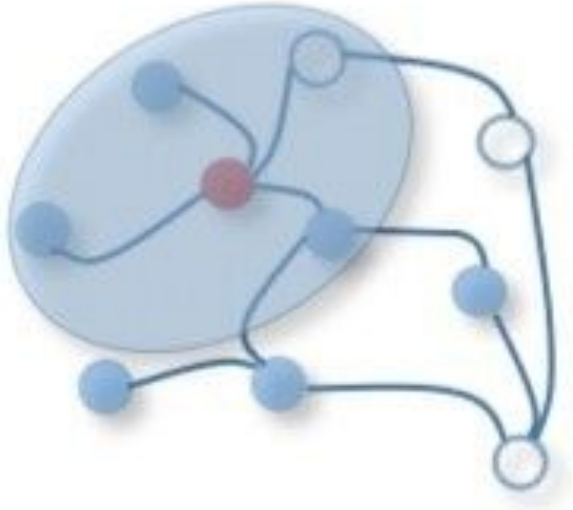
Outline

- **3D Data**
- Classification
- Segmentation and Detection
- 3D Data synthesis

The 3D Representation Challenge

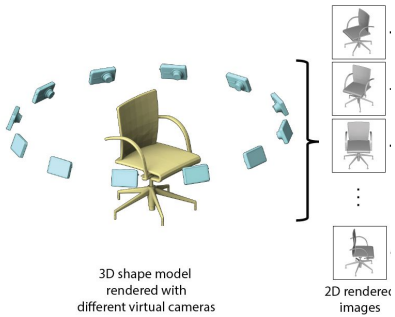


Rasterized form (regular grids)

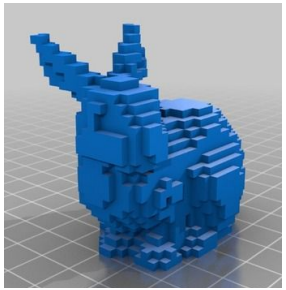


Geometric form (irregular)

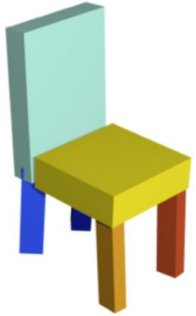
The 3D Representation Challenge



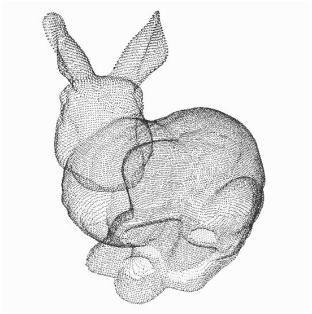
Multi-view



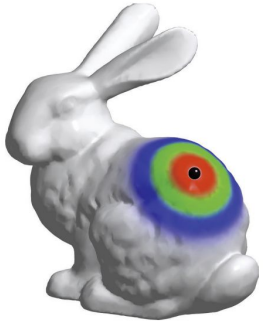
Volumetric



Part Assembly



Point Cloud



Mesh (Graph CNN)

$$F(x) = 0$$

Implicit Shapes

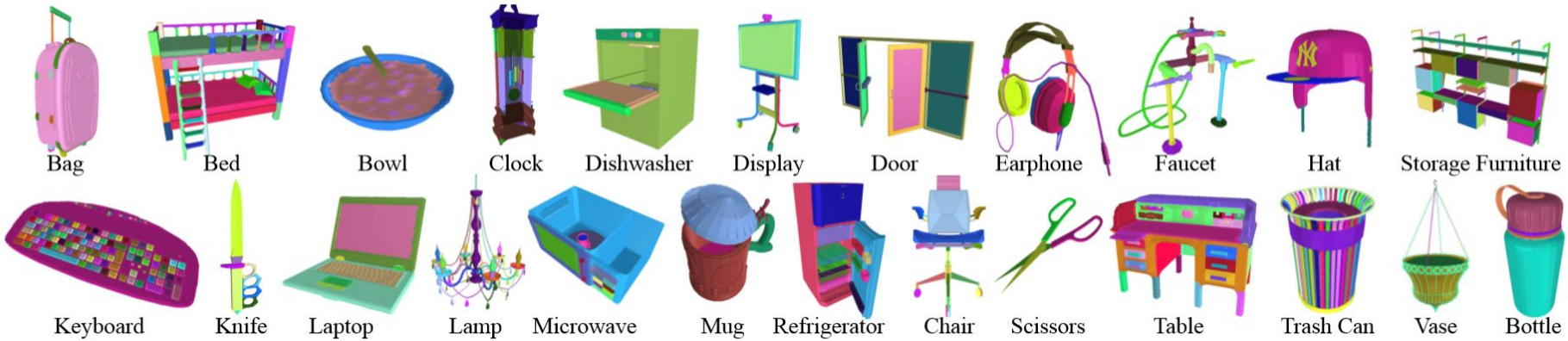
Datasets for 3D Objects

- **ModelNet**: Classification - absorbed by ShapeNet
- **ShapeNet**: Large-scale Synthetic Objects
- **3DScan**: Consumer-grade 3D Scanning

Chang et al., “**ShapeNet: An Information-Rich 3D Model Repository**” , *arXiv 2015*
Wu et al., "**3D ShapeNets: A deep representation for volumetric shapes**", *CVPR 2015*
Choi et al., “**A Large Dataset of Object Scans**", *arXiv 2016*

Datasets for 3D Object Parts

- **ShapeNetPart2019**: Fine-grained Part
  - Fine-grained (towards mobility)
  - Instance-level / Hierarchical



Mo et al., "**PartNet: A Large-Scale Benchmark for Fine-Grained and Hierarchical Part-Level 3D Object Understanding** ", *CVPR 2019*

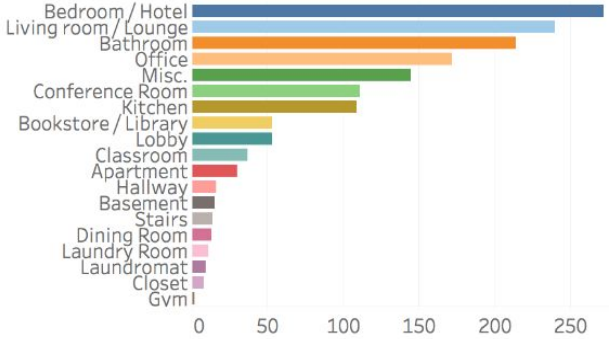
Datasets for Indoor 3D Scenes

- **SceneNet**: Large-scale Synthetic Scenes
  - 3D meshes / 5M Photorealistic Images

Ankur et al., "**Understanding RealWorld Indoor Scenes with Synthetic Data**", *CVPR 2016*
McCormac et al., "**SceneNet RGB-D: Can 5M Synthetic Images Beat Generic ImageNet Pre-training on Indoor Segmentation?**", *ICCV 2017*

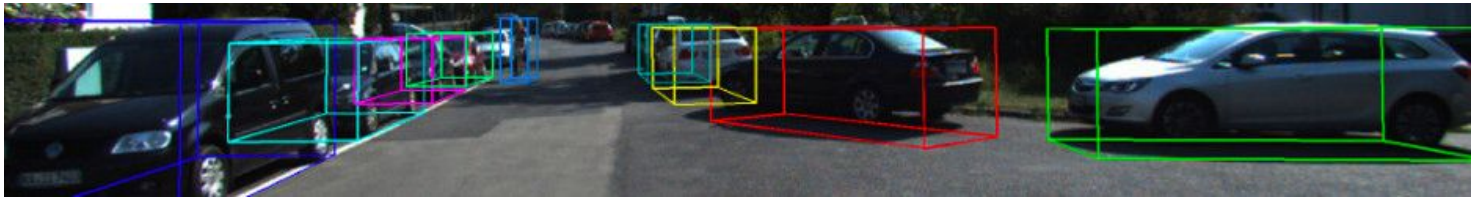
Datasets for Indoor 3D Scenes

- **ScanNet**: Large-scale Scanned Real Scenes
  - 2.5 M Views
  - 1500 RGBD scans
  - 3D camera poses
  - Surface reconstruction
  - Instance-level semantic segmentation

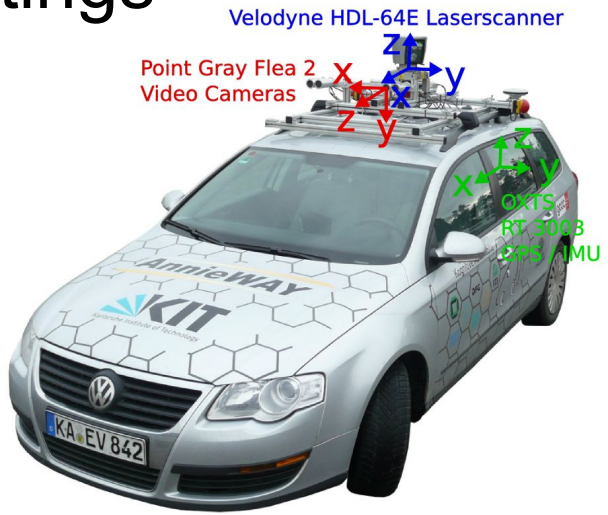


Dai et al., "**ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes**", *CVPR 2017*

Datasets for Outdoor 3D Scenes



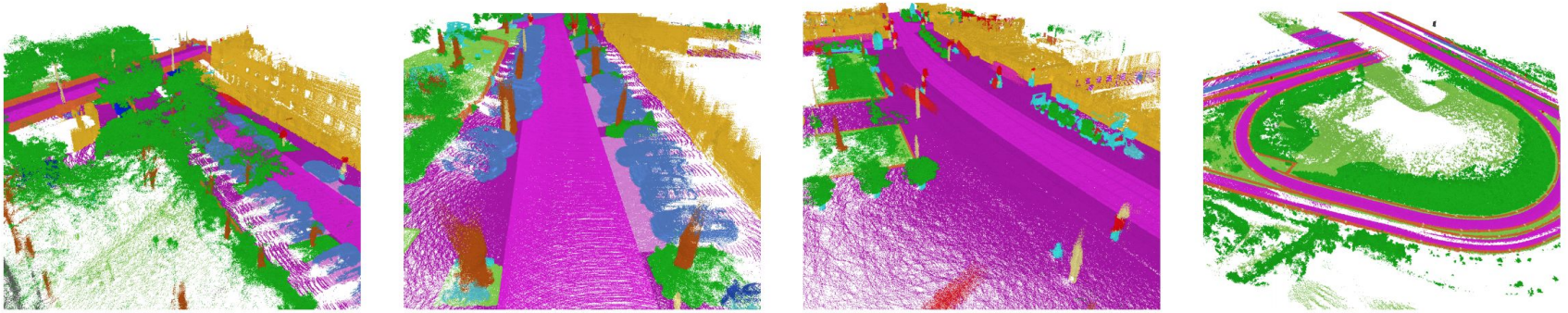
- **KITTI**: Autonomous navigation settings
  - Camera, LiDAR, GPS, IMU
  - 14000 frames training/testing
  - Stereo, Flow, Depth, Odometry, ...
  - Detection, Tracking, Road, Semantic, ...



Geiger et al., "**Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite**", *CVPR 2012*
Geiger et al., "**Vision meets Robotics: The KITTI Dataset**", *IJRR 2013*

Datasets for Outdoor 3D Scenes

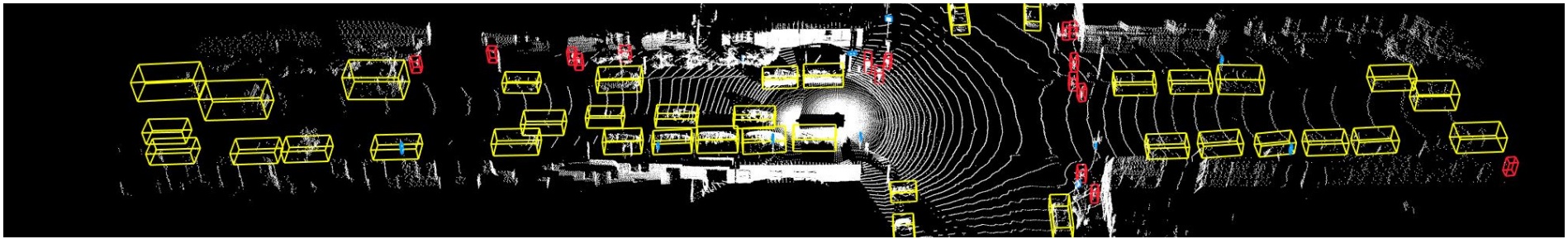
- **Semantic KITTI**: LiDAR data, labeled per point
  - Semantic Segmentation
  - Panoptic Segmentation
  - Semantic Scene Completion



Behley et al., "**SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences**", *ICCV 2019*

Datasets for Outdoor 3D Scenes

- **Waymo Open Dataset**: Autonomous Navigation at Large Scale
  - Camera / LiDAR data
  - 230.000 frames (1150 temporal sequences, 12M BB)
  - 2D/3D Object Detection/Tracking
  - Domain Adaptation (700 extra sequences from different locations)



Sun et al., "**Scalability in Perception for Autonomous Driving: Waymo Open Dataset**", *Arxiv 2019*

Outline

- 3D Data
- **Classification**
- Segmentation and Detection
- 3D Data synthesis

Task: 3D Classification

This is a chair!

- **Covered methods**: Volumetric CNN, OctNet, O-CNN, SparseConvNet, PointNet, PointNet++, RS CNN, DGCNN, Point ConvNet, KPConv, Monte Carlo Point Convolution, PConv, Multi-View CNN, Spectral CNN, Synchronized Spectral CNN, Spherical CNN

Multi-View CNN

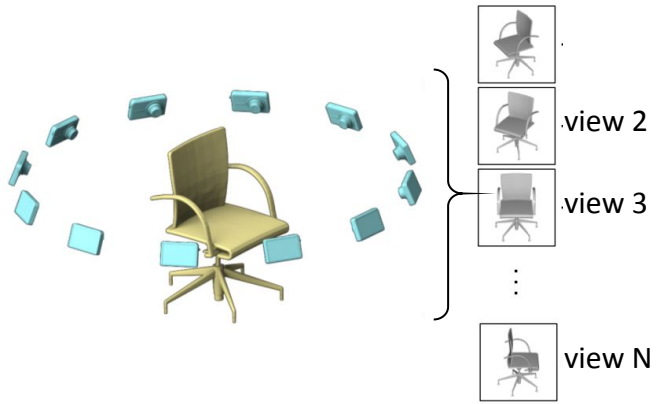
- Given an Input Shape

Su et al., "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *ICCV 2015*

view 1

Multi-View CNN

- Render with Multiple Virtual Cameras

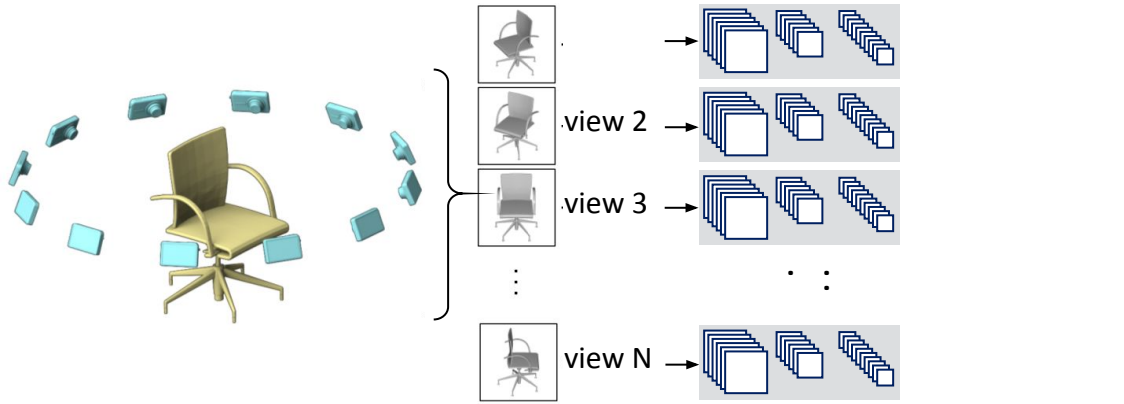


Su et al., "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *ICCV 2015*

view 1

Multi-View CNN

- The Rendered Images are Passed through CNN_1 for Image Features



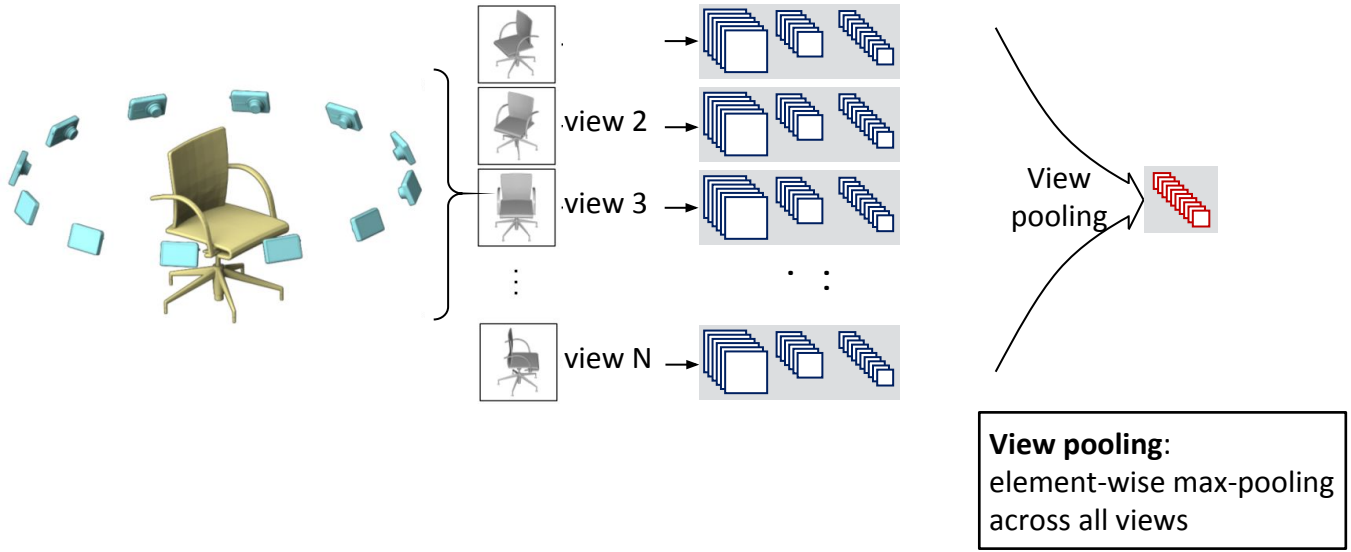
CNN_1: a ConvNet extracting image features

Su et al., "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *ICCV 2015*

view 1

Multi-View CNN

- All Image Features are Combined by View Pooling



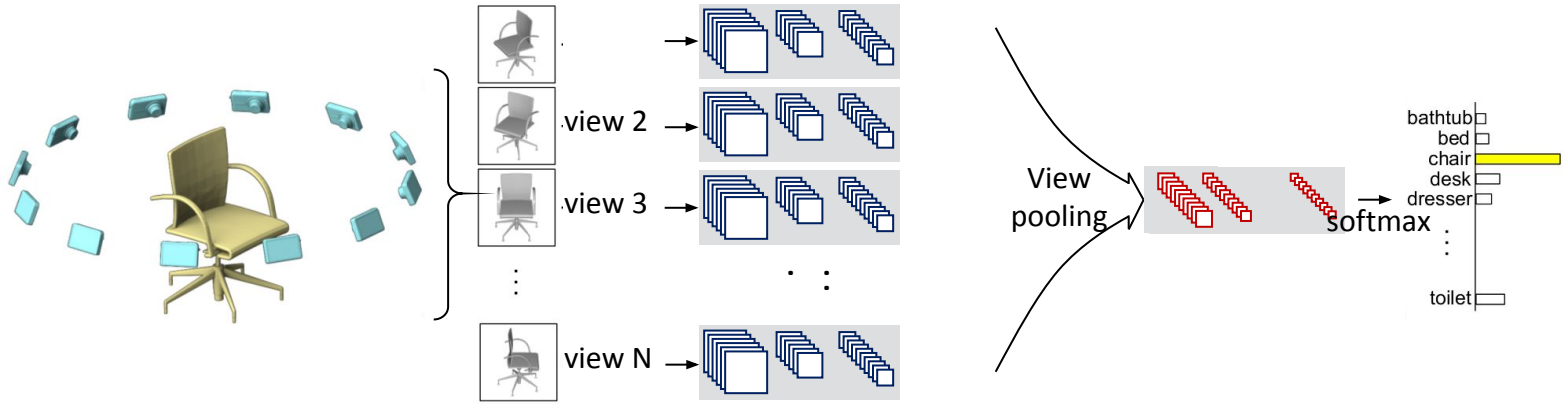
View pooling:
element-wise max-pooling across all views

Su et al., "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *ICCV 2015*

view 1

Multi-View CNN

- … and then Passed through CNN_2 and to Generate Final Predictions



CNN_2: a second ConvNet producing shape descriptors

Su et al., "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *ICCV 2015*

Multi-View CNN

- Experiments – Classification & Retrieval

	Method	Classification (Accuracy)	Retrieval (mAP)
Non-deep	SPH [16]	68.2%	33.3%
	LFD [5]	75.5%	40.9%
	3D ShapeNets [37]	77.3%	49.2%
	FV, 12 views	84.8%	43.9%
	CNN, 12 views	88.6%	62.8%
	MVCNN, 12 views	**89.9%**	70.1%
	MVCNN+metric, 12 views	89.5%	**80.2%**
	MVCNN, 80 views	90.1%	70.4%
	MVCNN+metric, 80 views	**90.1%**	**79.5%**

On ModelNet40

[credit: Hang Su]

Su et al., "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *ICCV 2015*

Multi-View CNN

- Indeed gives good performance
- Can leverage vast literature of image classification
- Can use pre-trained features
- Needs projection
- What if the input is noisy and/or incomplete? e.g., point cloud

Su et al., "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *ICCV 2015*

Volumetric CNN

- Can we use CNNs without 2D-3D projection?

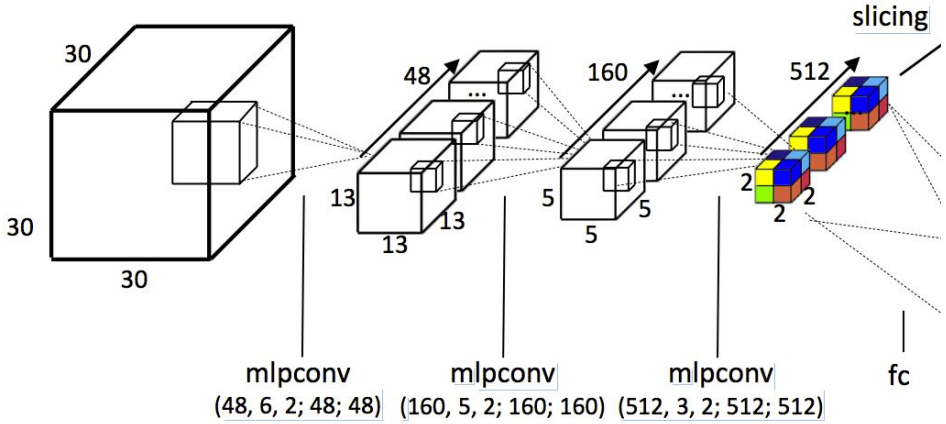
- Straightforward idea: 3D native convolution

Volumetric CNN: Voxelization

- Represent the occupancy of regular 3D grids

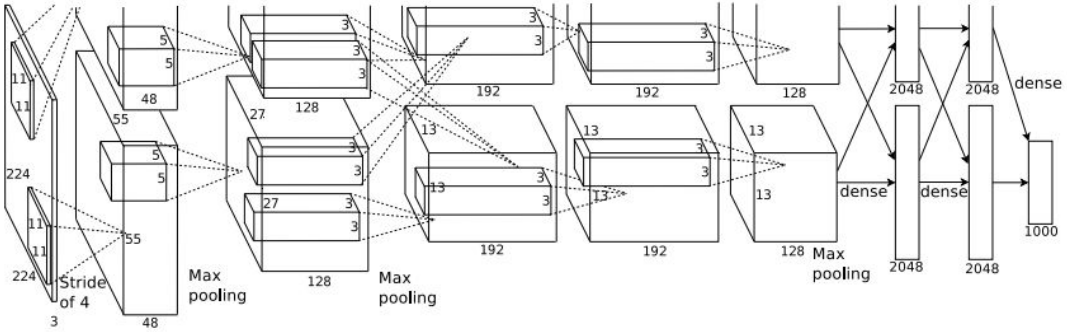
Volumetric CNN: Voxelization

- 3D convolution uses 4D kernels



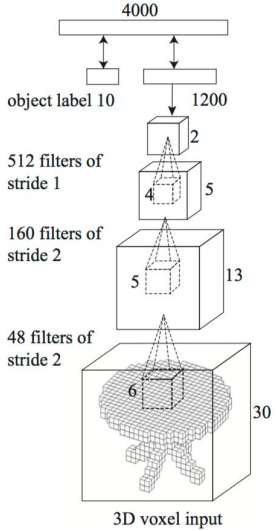
Volumetric CNN: Voxelization

- Complexity Issue



AlexNet, 2012

Input resolution:
224x224=50176

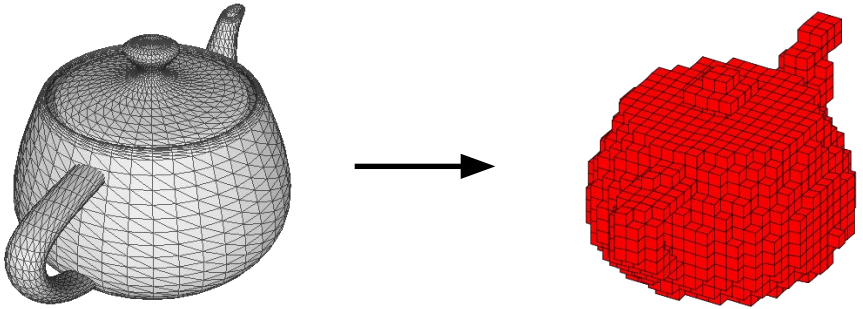


3DShapeNets, 2015

Input resolution:
30x30x30 =27000

Volumetric CNN: Voxelization

- Complexity Issue
- Information loss in voxelization



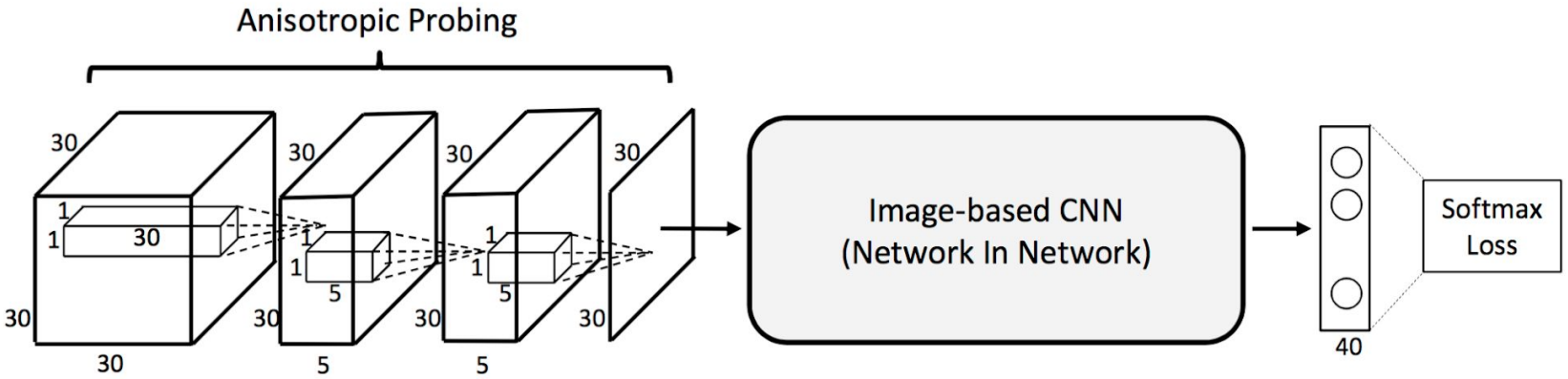
Polygon Mesh

Occupancy Grid
30x30x30

Volumetric CNN: Voxelization

- Idea 1: Learn to Project

Idea: "X-ray" rendering + Image (2D) CNNs
very low #param, very low computation

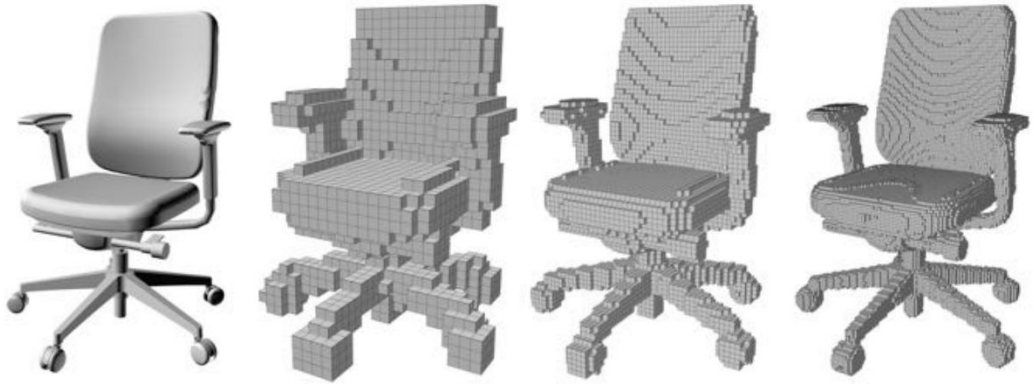


Su et al., "**Volumetric and Multi-View CNNs for Object Classification on 3D Data**", *CVPR 2016*

Many other works in autonomous driving that uses bird's eye view for object detection.

Volumetric CNN: Voxelization

- More Principled: Leverage Sparsity of 3D Surface



$\frac{\#occupied\ grid}{\#total\ grid}$

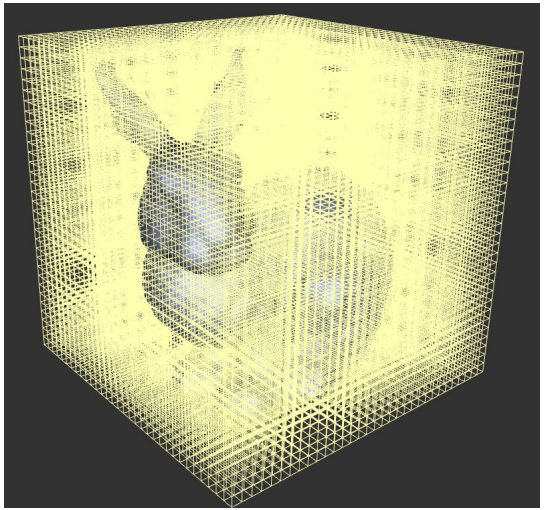
Occupancy:	10.41%	5.09%	2.41%
Resolution:	32	64	128

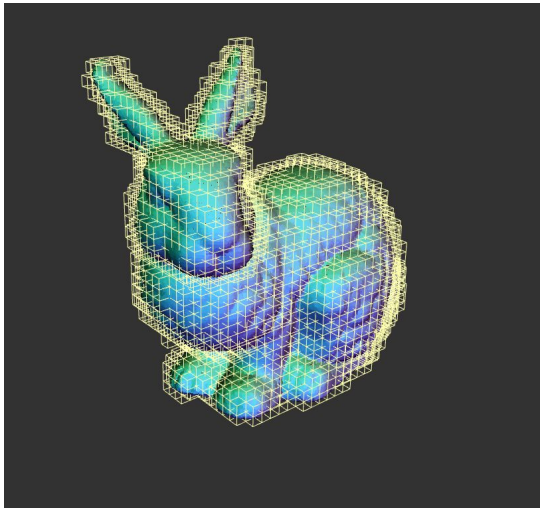
Su et al., "**Volumetric and Multi-View CNNs for Object Classification on 3D Data**", *CVPR 2016*

Many other works in autonomous driving that uses bird's eye view for object detection

Volumetric CNN: Voxelization

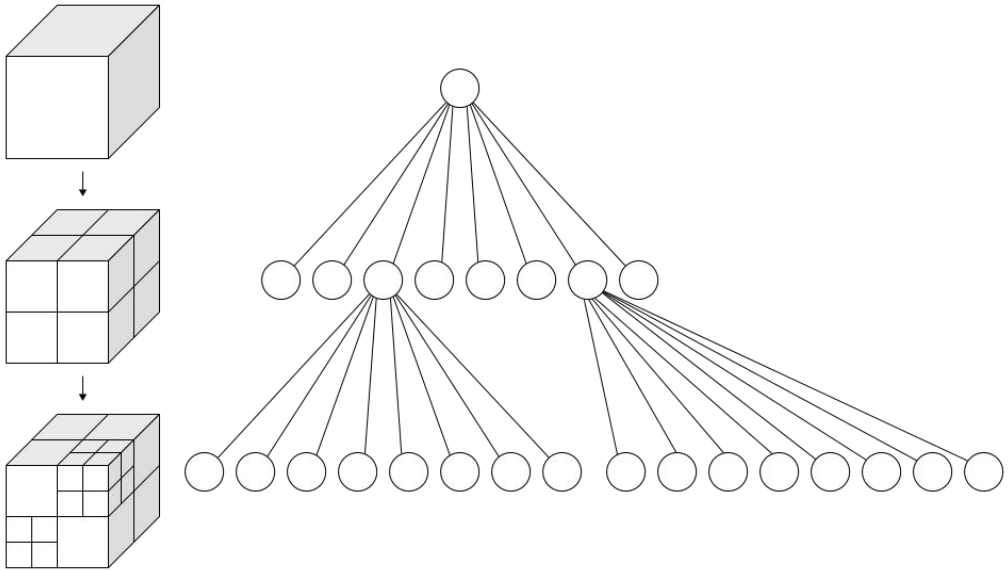
- Store only the Occupied Grids
  - Store the sparse surface signals
  - Constrain the computation near the surface





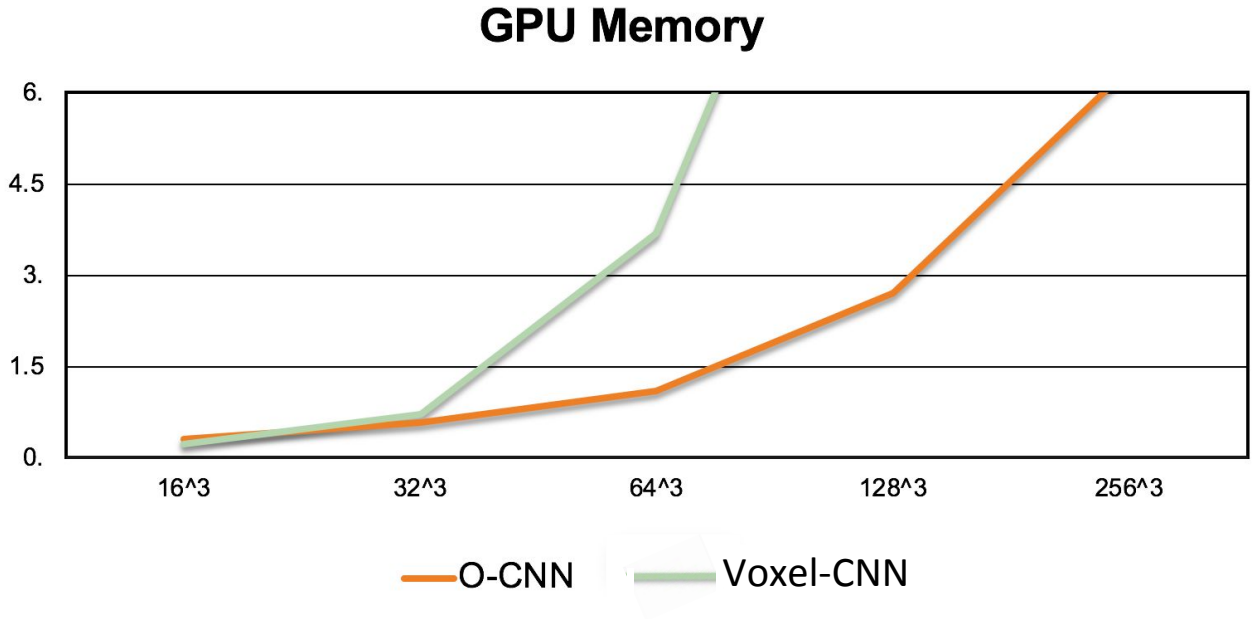
Volumetric CNN: Voxelization

- Octree: Recursively Partition the Space
  - Each internal node has exactly eight children
  - Neighborhood searching: Hash table



Volumetric CNN: Voxelization

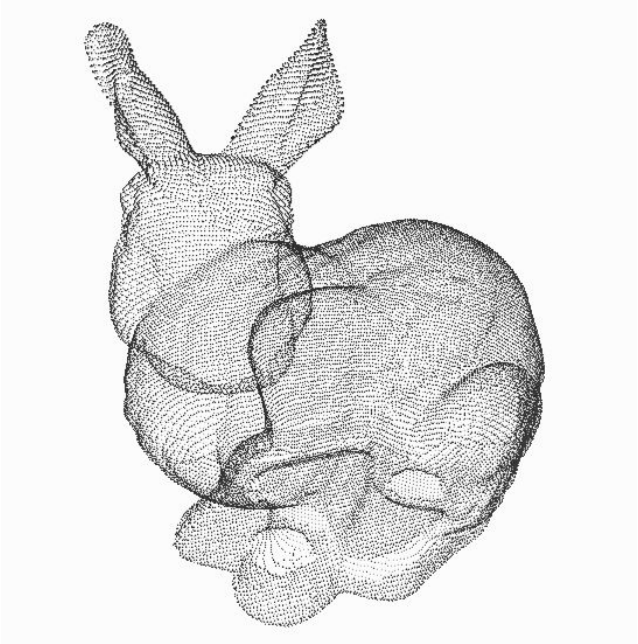
- Octree: Recursively Partition the Space
  - Memory Efficiency



Volumetric CNN: Voxelization

- Implementation
  - SparseConvNet
  - https://github.com/facebookresearch/SparseConvNet
  - Uses ResNet architecture
  - State-of-the-art for 3D analysis
  - Takes time to train

Point Networks

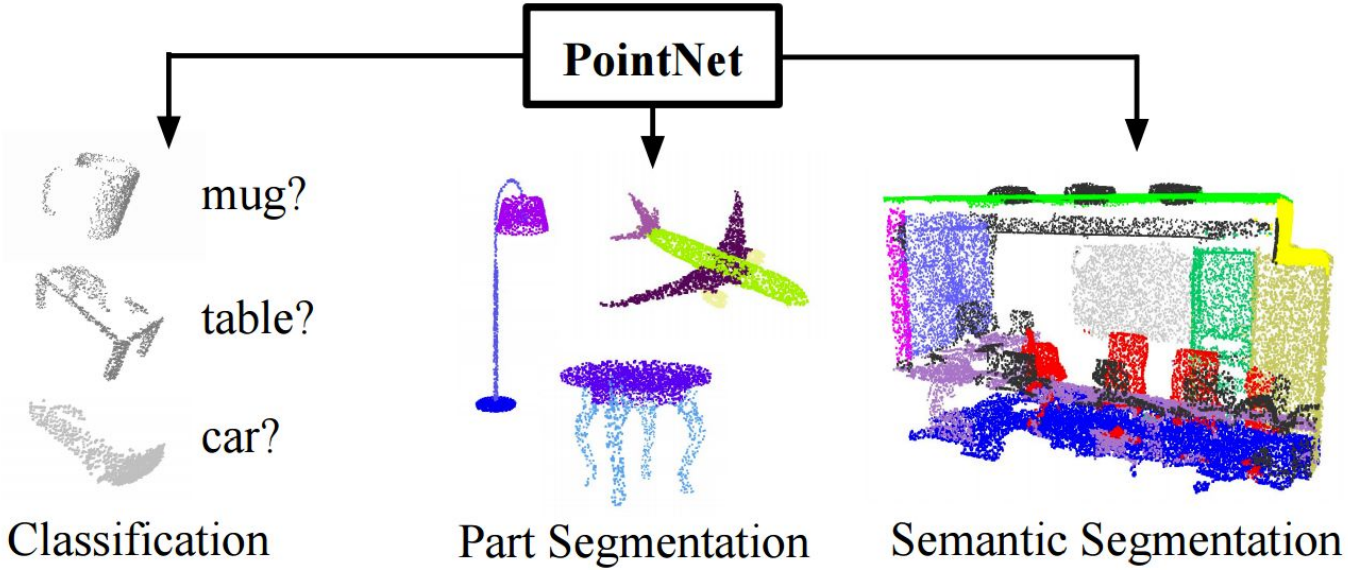


Point cloud

(The most common 3D sensor data)

Point Networks

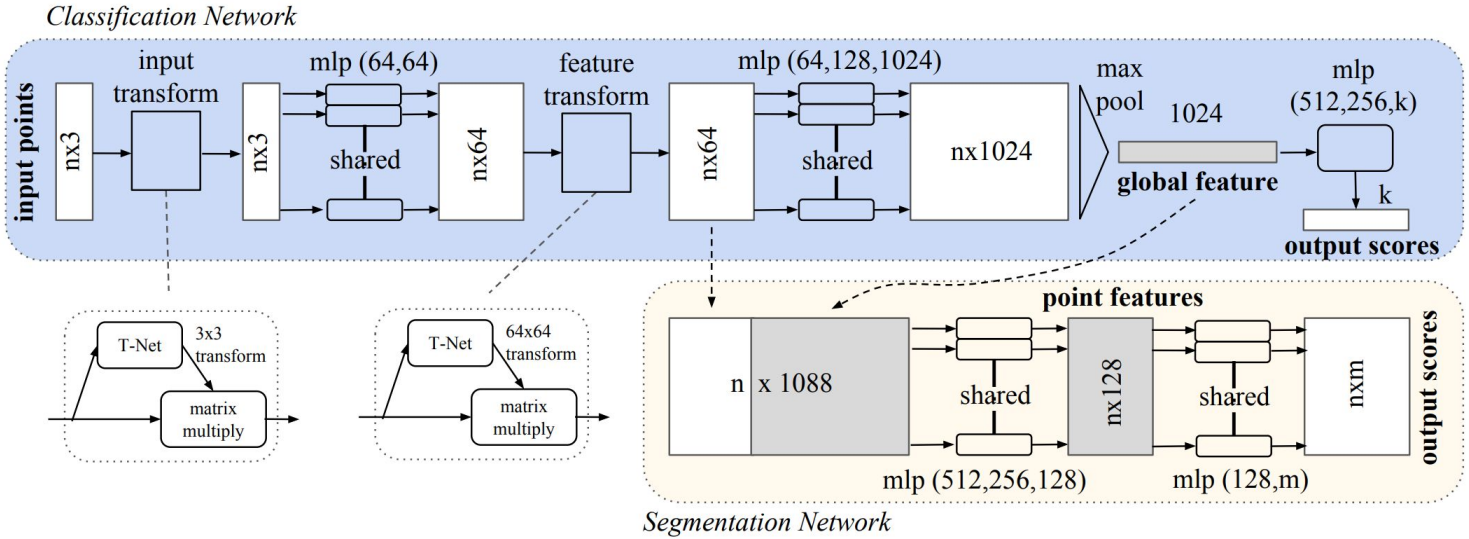
- Directly Process Point Cloud Data
  - End-to-end learning for **unstructured, unordered** point data



Qi, Charles R., et al. "**Pointnet: Deep learning on point sets for 3d classification and segmentation**", CVPR 2017
Zaheer, Manzil, et al. "**Deep sets**", NeurIPS 2017

Point Networks

- Directly Process Point Cloud Data
  - **Segmentation**: MLP for each points
  - **Classification**: MaxPooling between point features

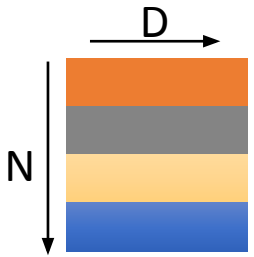


Qi, Charles R., et al. "**Pointnet: Deep learning on point sets for 3d classification and segmentation**", CVPR 2017

Zaheer, Manzil, et al. "**Deep sets**", NeurIPS 2017

Point Networks

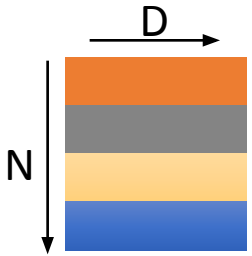
- Permutation invariance:
  - Point cloud: N **orderless** points, each represented by a D dim coordinate



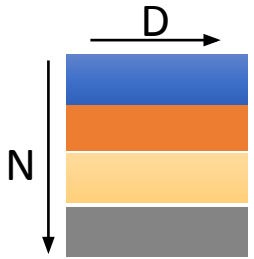
2D array representation

Point Networks

- Permutation invariance:
  - Point cloud: N **orderless** points, each represented by a D dim coordinate



represents the same **set** as

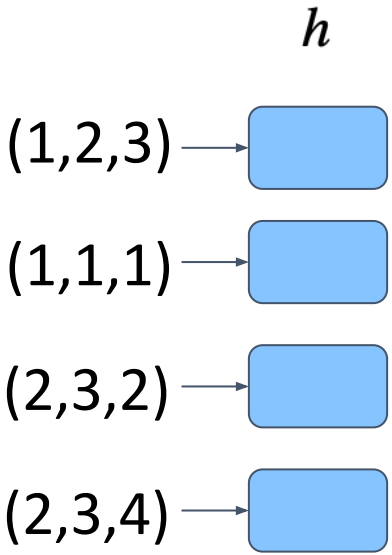


2D array representation

Point Networks

- Construct a Symmetric Function

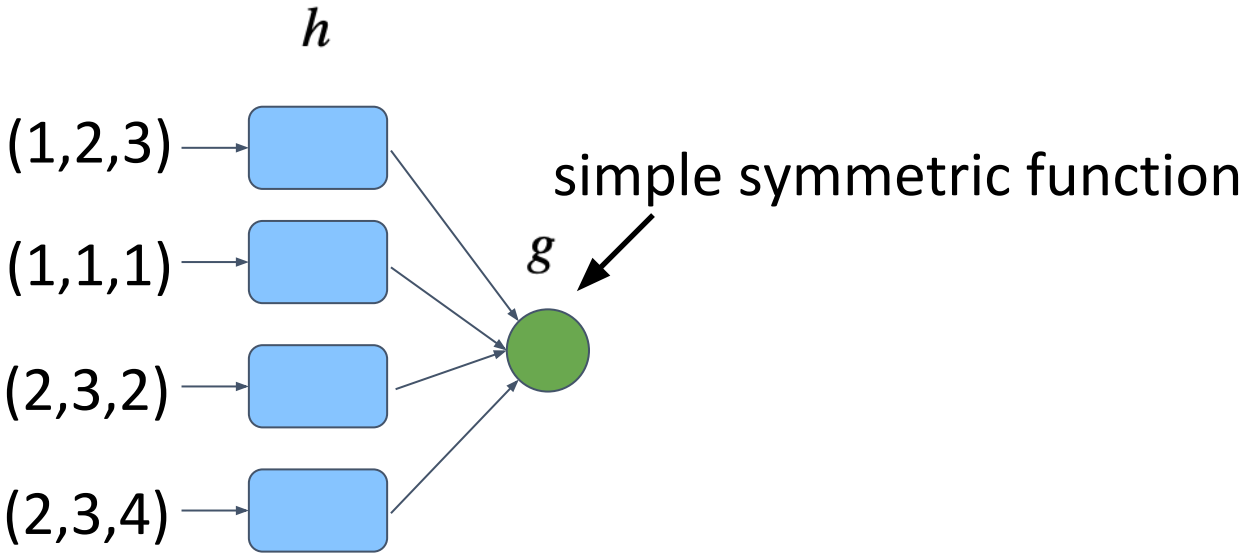
$f(x_1,x_2,\ldots,x_n) = \gamma \circ g(h(x_1),\ldots,h(x_n))$ is symmetric if g is symmetric



Point Networks

- Construct a Symmetric Function

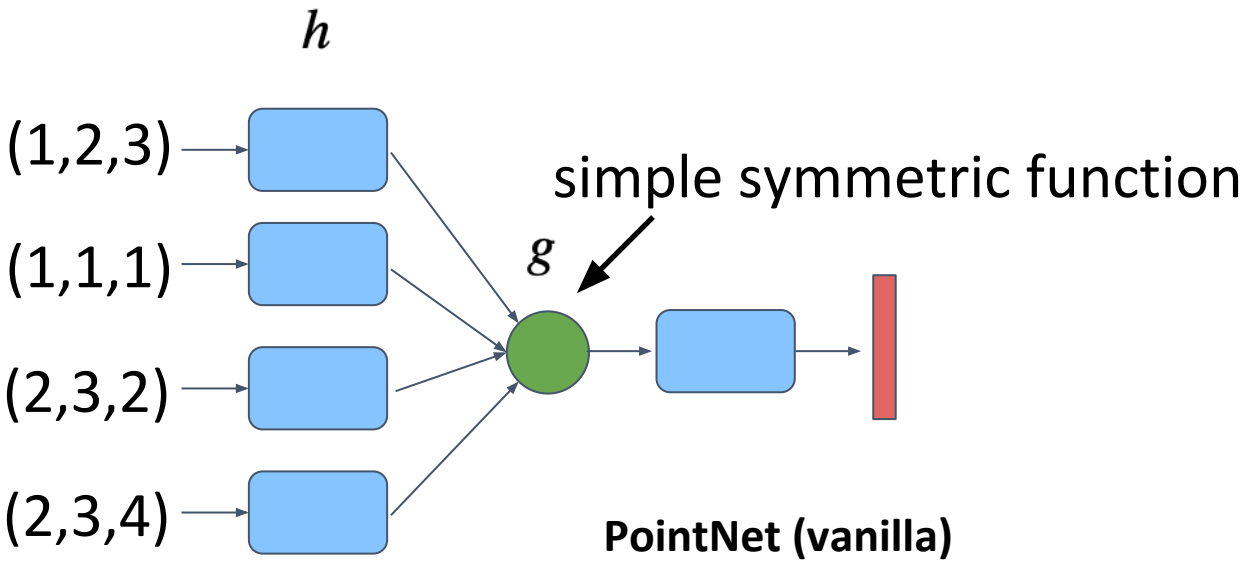
$f(x_1, x_2, \ldots, x_n) = \gamma \circ g(h(x_1), \ldots, h(x_n))$ is symmetric if g is symmetric



Point Networks

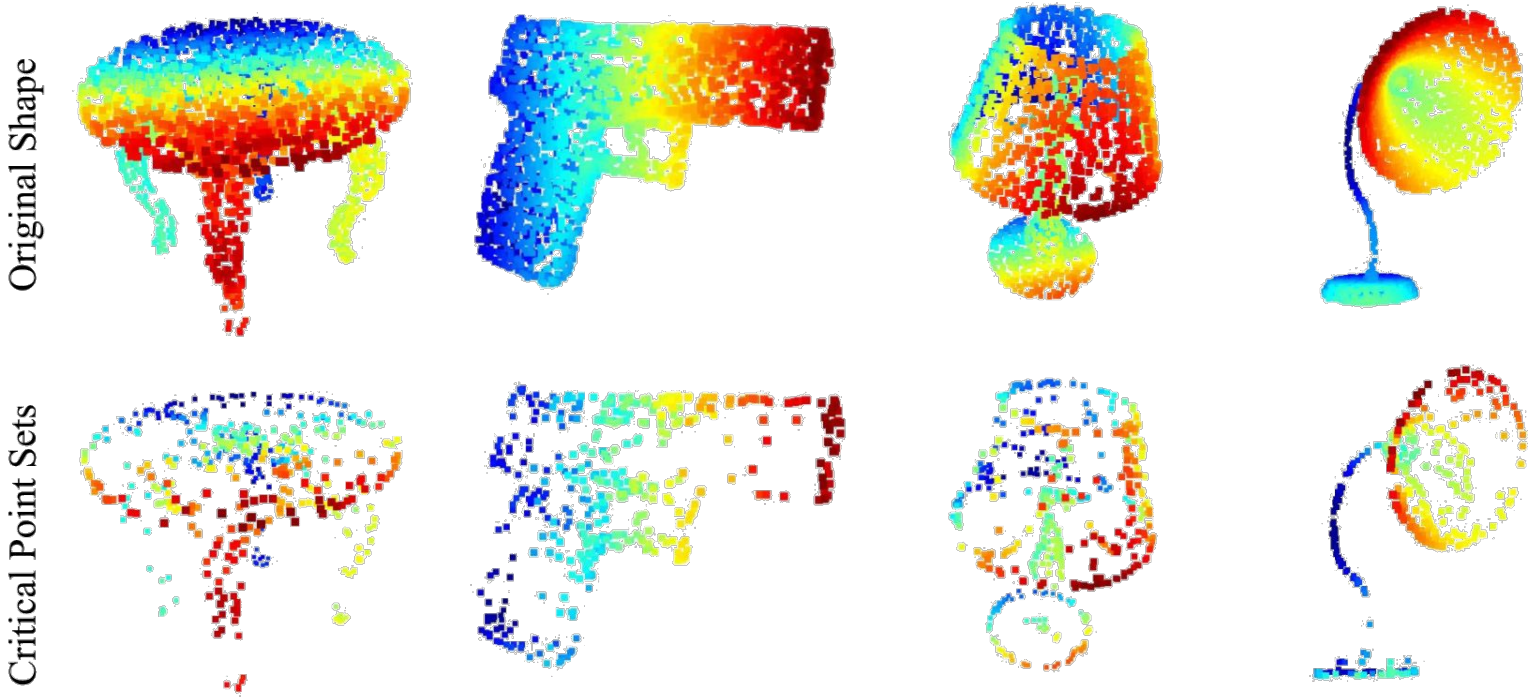
- Construct a Symmetric Function

$f(x_1, x_2, \ldots, x_n) = \gamma \circ g(h(x_1), \ldots, h(x_n))$ is symmetric if g is symmetric



Point Networks

- Salient points are discovered!

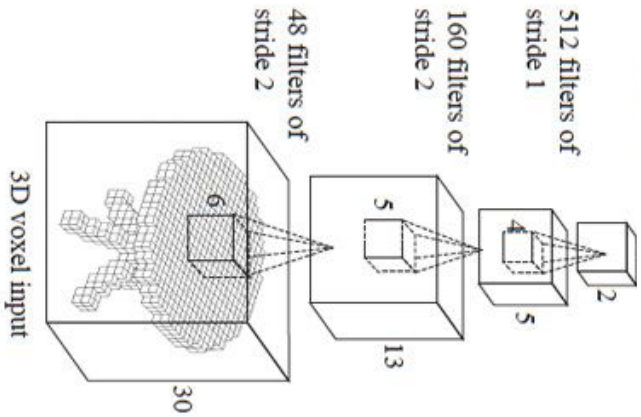


Point Networks: Limitations

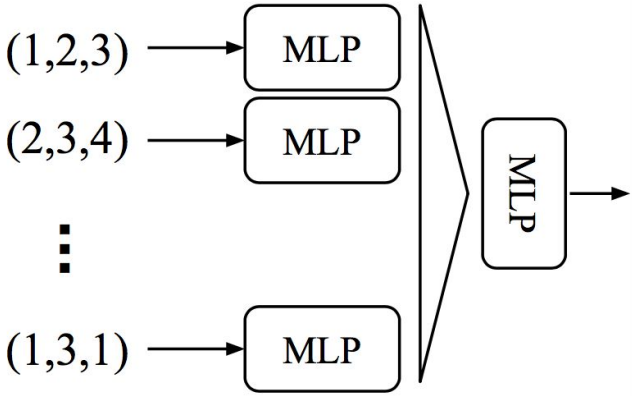
Hierarchical feature learning
Multiple levels of abstraction

v.s.

Global feature learning
Either one point or all points



3D CNN (Wu et al.)



PointNet (vanilla) (Qi et al.)

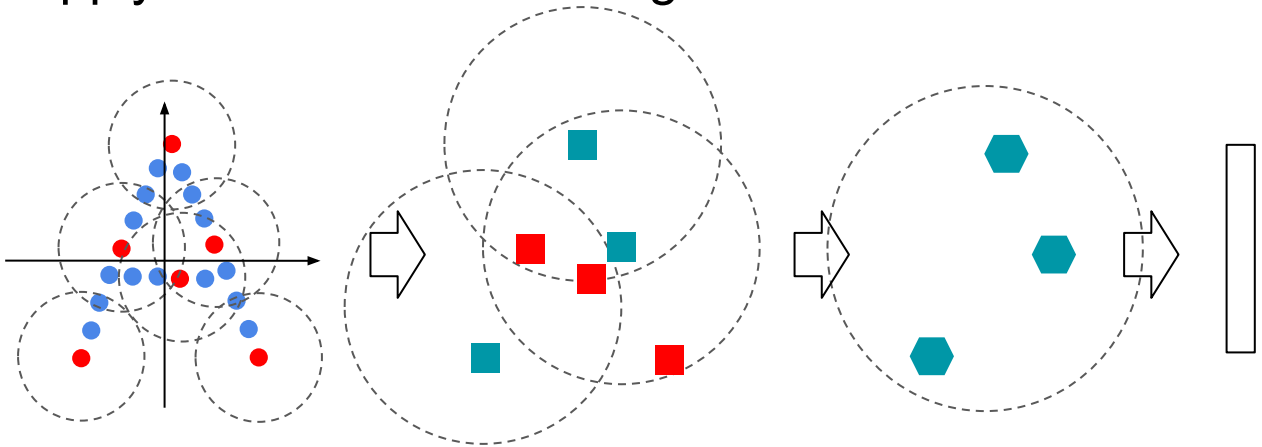
No local context for each point!
Global feature depends on absolute coordinate.
Hard to generalize to unseen scene configurations!

Point Networks

- Points in Metric Space:
  - Learn “kernels” in 3D space and conduct convolution
  - Kernels have compact spatial support
  - For convolution, we need to find neighboring points
  - Possible strategies for range query
    - Ball query (results in more stable features)
    - k-NN query (faster)

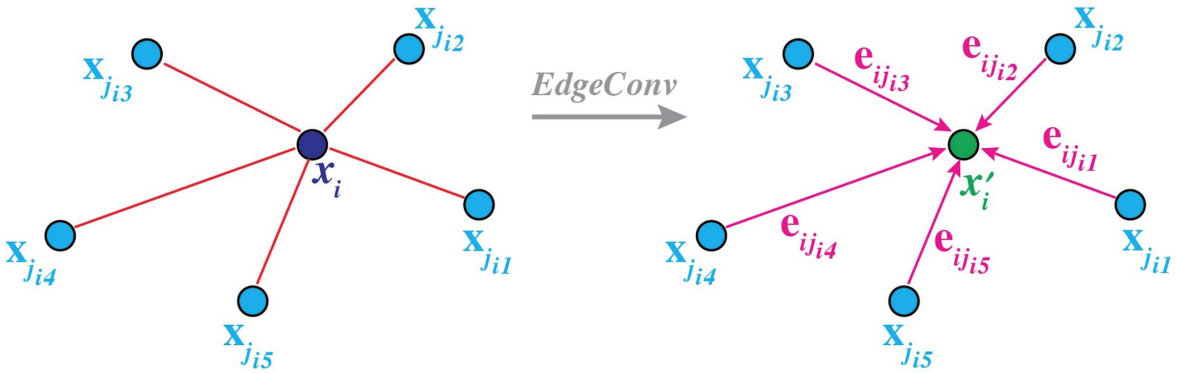
PointNet++: Multi-Scale PointNet

- Repeat
  - Sample anchor points
  - Find neighborhood of anchor points
  - Apply PointNet in each neighborhood to mimic convolution



Point Convolution As Graph Convolution

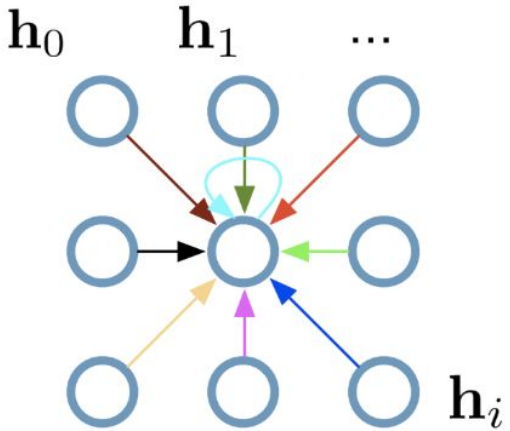
- Points -> Nodes
- Neighborhood -> Edges
- Graph CNN for point cloud processing



Wang et al., "**Dynamic Graph CNN for Learning on Point Clouds**", *Transactions on Graphics, 2019*
Liu et al., "**Relation-Shape Convolutional Neural Network for Point Cloud Analysis**", *CVPR 2019*

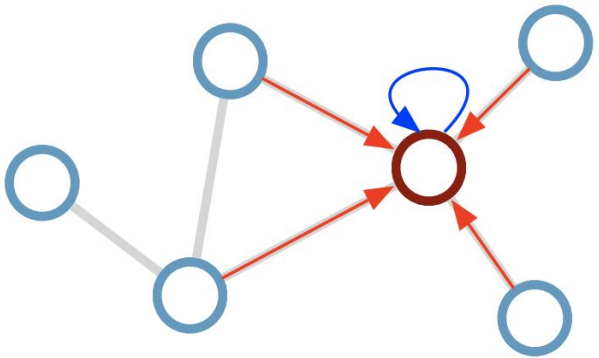
Point Convolution As Graph Convolution

Convolutional Neural Network (CNN)



$$\mathbf{h}_4^{(l+1)} = \sigma\left(\mathbf{W}_0^{(l)}\mathbf{h}_0^{(l)} + \mathbf{W}_1^{(l)}\mathbf{h}_1^{(l)} + \cdots + \mathbf{W}_8^{(l)}\mathbf{h}_8^{(l)}\right)$$

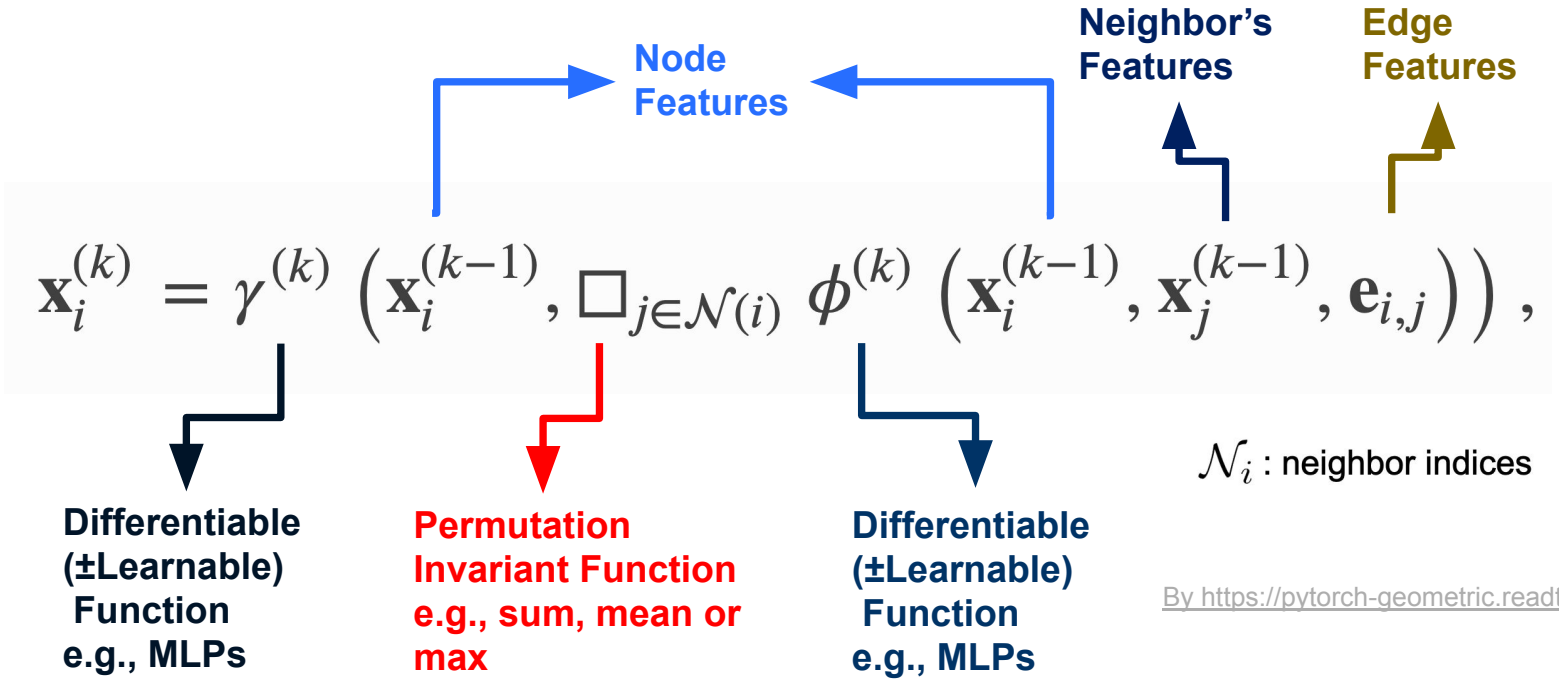
Graph Convolutional Network (GCN)



$$\mathbf{h}_i^{(l+1)} = \sigma\left(\mathbf{h}_i^{(l)}\mathbf{W}_0^{(l)} + \sum_{j\in\mathcal{N}_i}\frac{1}{c_{ij}}\mathbf{h}_j^{(l)}\mathbf{W}_1^{(l)}\right)$$

$\mathcal{N}_i$: neighbor indices c_{ij} : norm. constant (fixed/trainable)

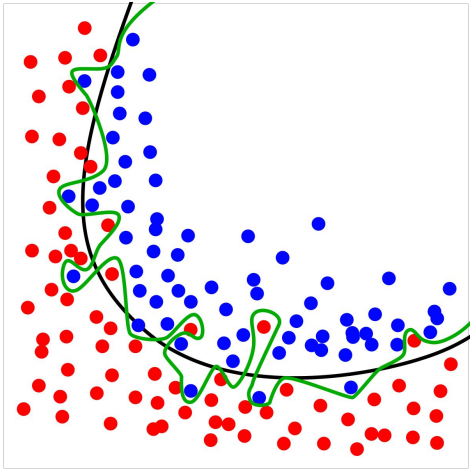
Point Convolution As Graph Convolution



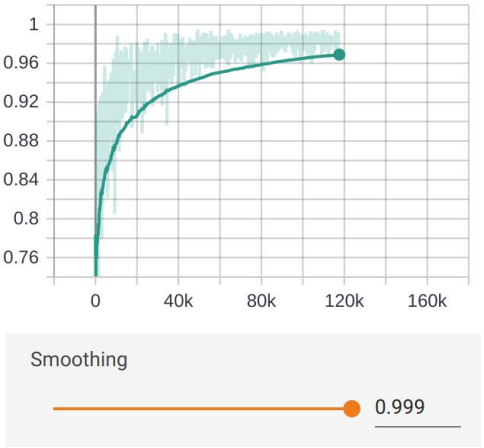
By https://pytorch-geometric.readthedocs.io

Point Convolution As Graph Convolution

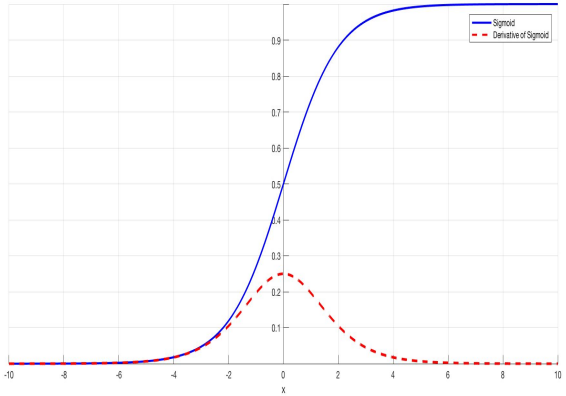
- GCN: Main Limitations



Over-fitting



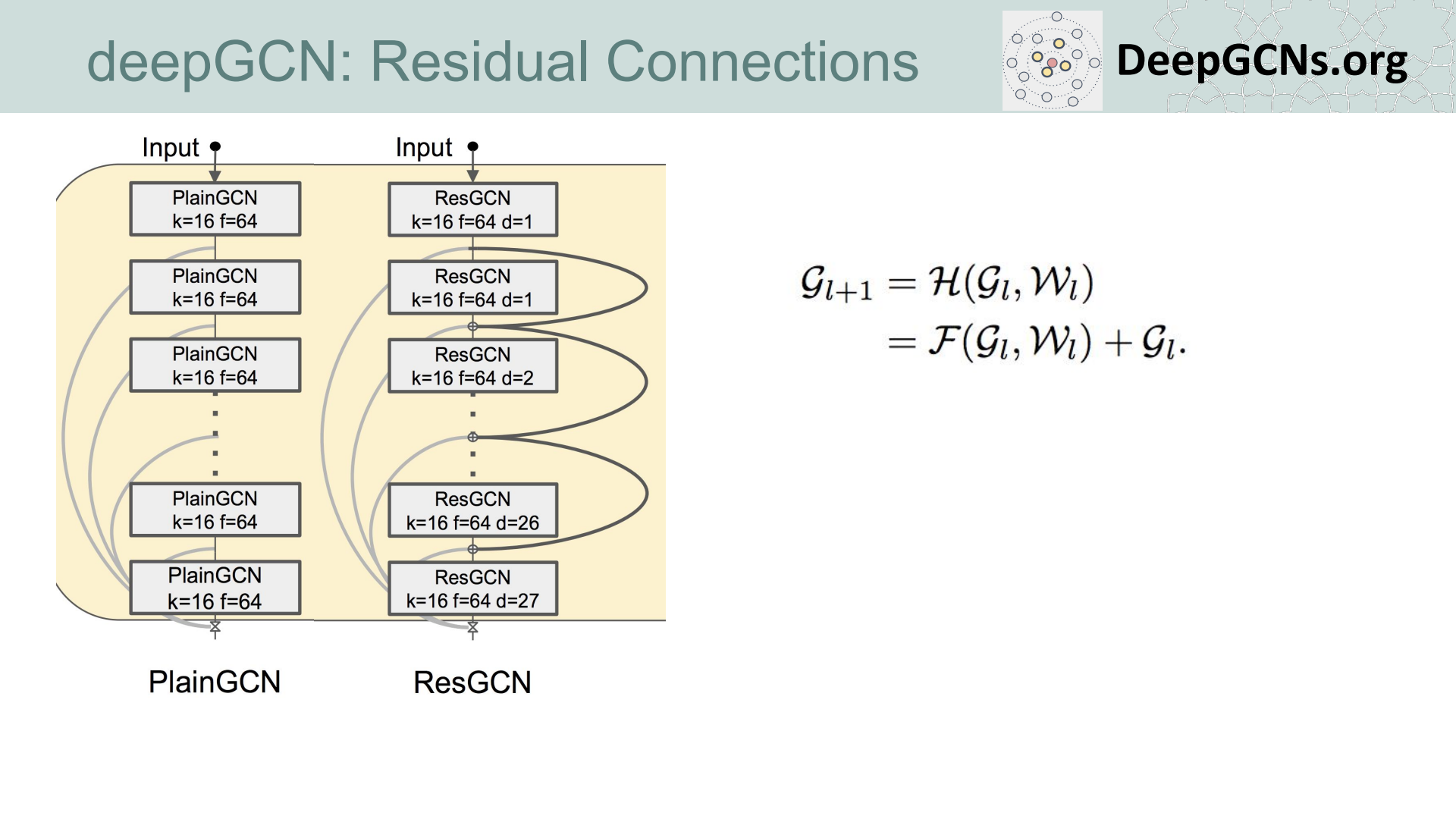
Over-smoothing



Vanishing Gradient

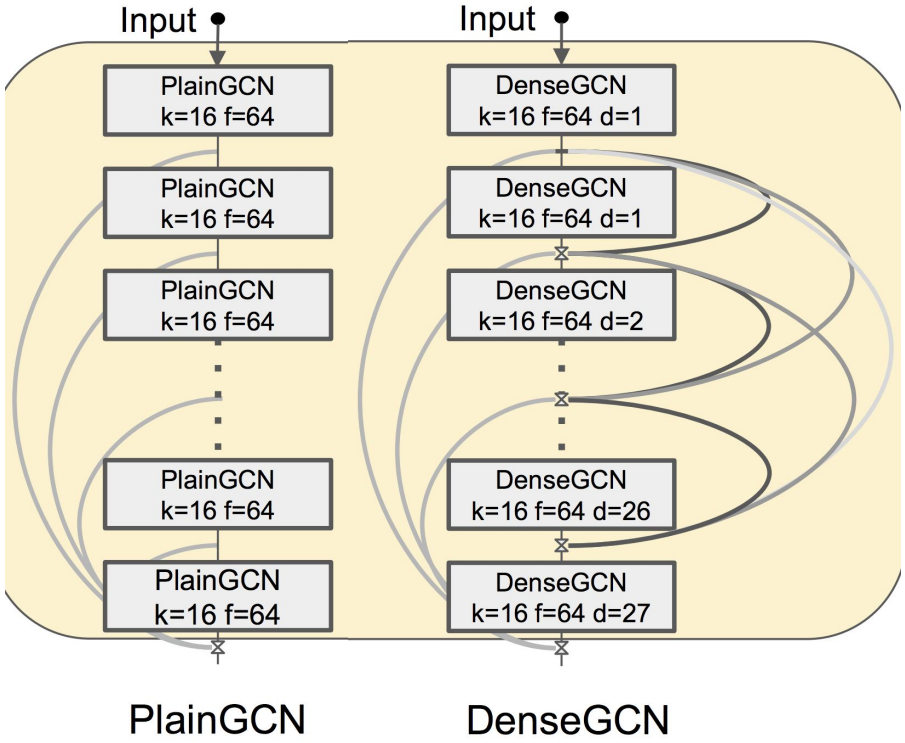
Figures from https://towardsdatascience.com/the-vanishing-gradient-problem-69bf08b15484

deepGCN: Residual Connections



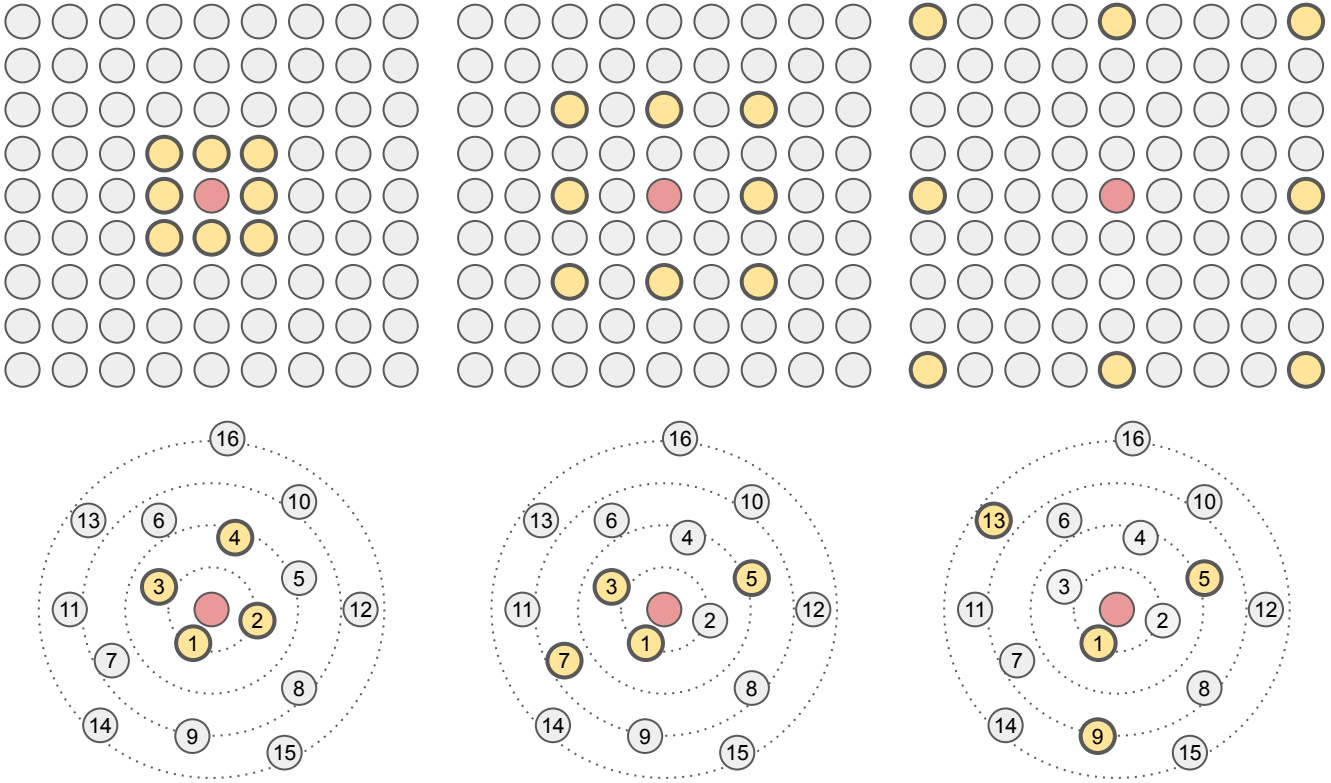
$$\mathcal{G}_{l+1} = \mathcal{H}(\mathcal{G}_l, \mathcal{W}_l)$$
$$= \mathcal{F}(\mathcal{G}_l, \mathcal{W}_l) + \mathcal{G}_l.$$

deepGCN: Dense Connections



$$\begin{aligned}\mathcal{G}_{l+1} &= \mathcal{H}(\mathcal{G}_l, \mathcal{W}_l) \\ &= \mathcal{T}(\mathcal{F}(\mathcal{G}_l, \mathcal{W}_l), \mathcal{G}_l) \\ &= \mathcal{T}(\mathcal{F}(\mathcal{G}_l, \mathcal{W}_l), ..., \mathcal{F}(\mathcal{G}_0, \mathcal{W}_0), \mathcal{G}_0).\end{aligned}$$

deepGCN: Dilated GCN



Standard GCNs are not Geometry-Aware

- Note that points are **sampled** from surfaces
- Ideally, features describe the geometry of underlying surface. Should be sample invariant
- But GCNs lack design to address sample invariance
- Remind us “density estimation” from a population
- Rescue: Estimate the continuous kernel and point density for continuous convolution

Mathematically Proper Conv. Discretization

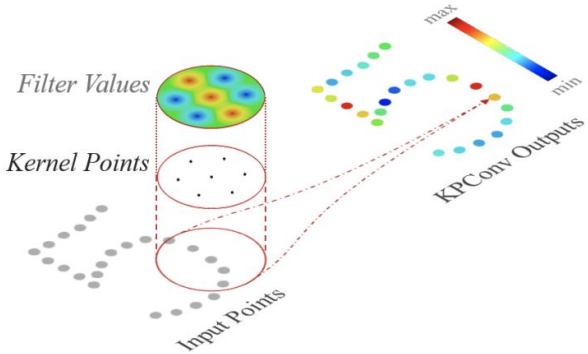
- Continuous conv: $$(\mathcal{F} * g)(x) = \int g(y - x) f(y) dy$$

- Empirical conv: $$(\mathcal{F} * g)(x) = \sum_{x_i \in \mathcal{N}_x} g(x_i - x) f_i$$

- Learn kernel value at anchor points and interpolate to build continuous kernel

$$\kappa_{jm}(z) = \sum_l k_{ljm} \Phi(|z - y_l|)$$

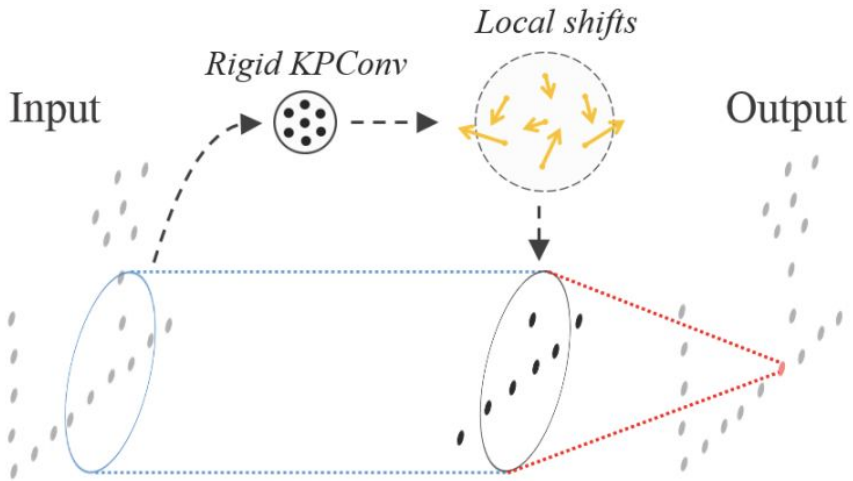
Φ: RBF kernel



Atzmon et al., "**Point Convolutional Neural Networks by Extension Operators**", *Trans. on Graphics, 2018*
Thomas et al., "**KPConv: Flexible and Deformable Convolution for Point Clouds**", *ICCV 2019*

Deformable Kernel for Deformable Objects

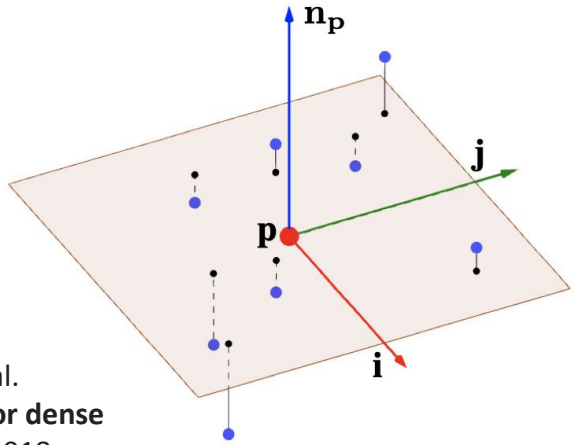
- Deformable point-based kernel
- The 3D version of 2D deformable convolution



Thomas et al., "**KPConv: Flexible and Deformable Convolution for Point Clouds**", *ICCV 2019*

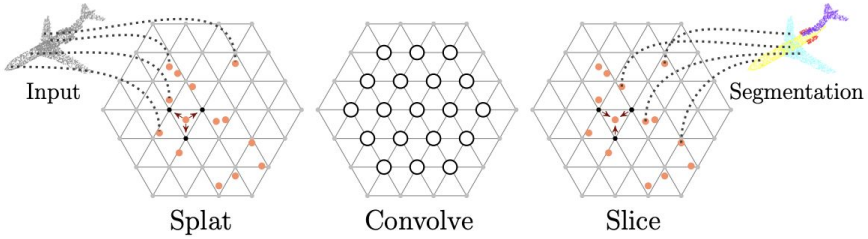
Other Ways to Address 3D Conv

- Tangent convolution:
  - Project & interpolate local points to the tangent plane (PCA for orientation)



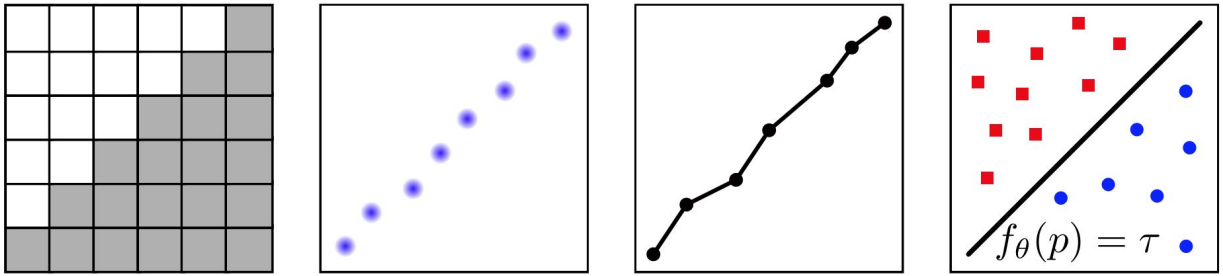
Tatarchenko, Maxim, et al. "**Tangent convolutions for dense prediction in 3d**", *CVPR* 2018.

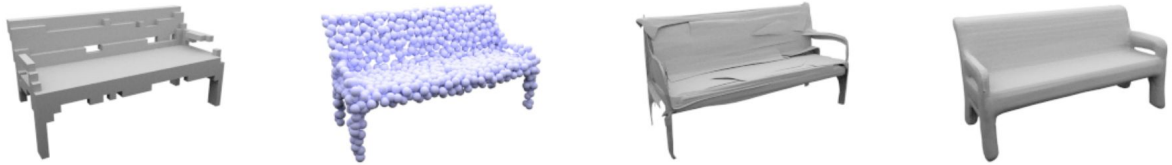
- Lattice
  - high-dimensional space



Su, Hang, et al. "**Splatnet: Sparse lattice networks for point cloud processing**", CVPR 2018

Implicit Representation?

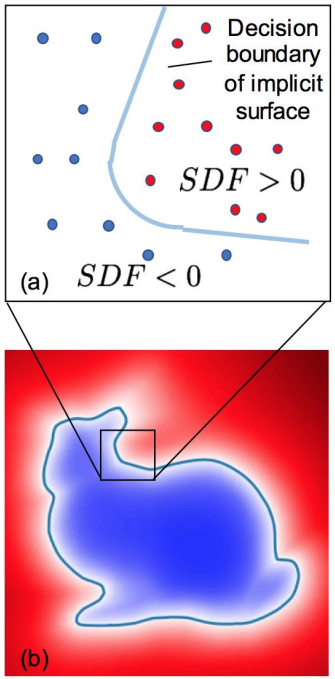




$$F(x) = 0$$

Mescheder et al., “**Occupancy Networks: Learning 3D Reconstruction in Function Space**”, *CVPR 2019*
Park et al., “**DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation**”, *CVPR 2019*
Chen et al., “**Learning Implicit Fields for Generative Shape Modeling**”, *CVPR 2019*

Implicit Representation?

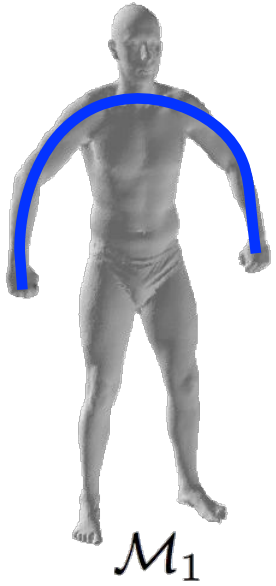


$F(x) = 0$

Mescheder et al., "**Occupancy Networks: Learning 3D Reconstruction in Function Space**", *CVPR 2019*
Park et al., "**DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation**", *CVPR 2019*
Chen et al., "**Learning Implicit Fields for Generative Shape Modeling**", *CVPR 2019*

Spectral Convolution

- Recognition with Isometric Invariance?



$\mathcal{M}_1$

$\mathcal{M}_2$

geodesic =
intrinsic
isometry = length-preserving

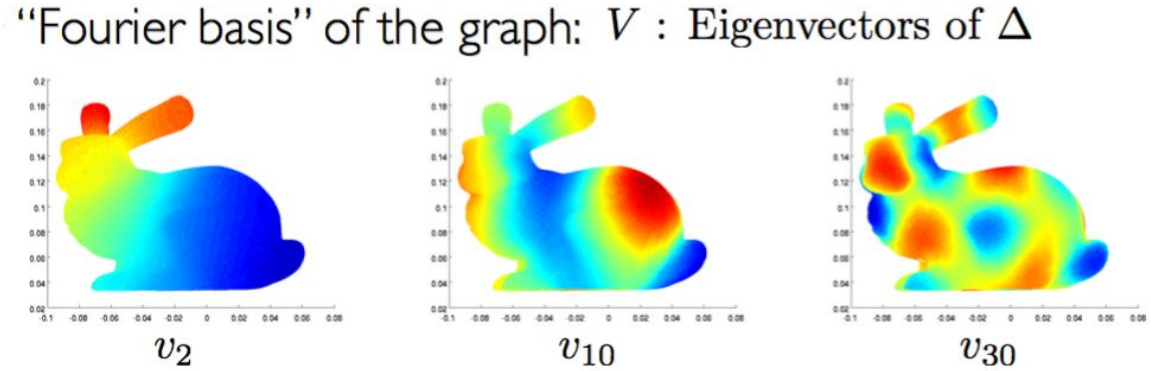
Spectral Convolution

- Convolution done in the spectral domain
- Kernels are also built in spectral domain
- Activation done in the spatial domain

Masci et al., "**Geometric deep learning on graphs and manifolds using mixture model CNNs**", *CVPR 2017*

Spectral Convolution

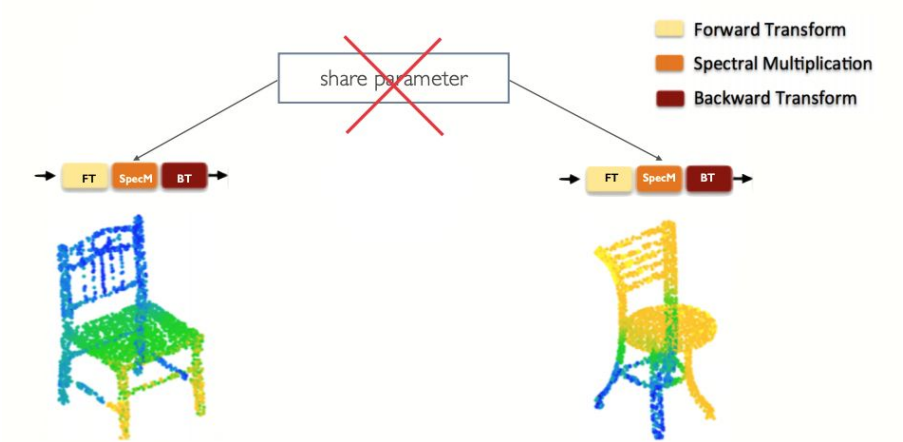
- Challenge: Obtain Fourier Basis
  - Derived by eigenfunctions of self-adjoint operators, e.g. Laplacian-Bertrami or Dirac operator



Masci et al., "**Geometric deep learning on graphs and manifolds using mixture model CNNs**", *CVPR 2017*

Spectral Convolution

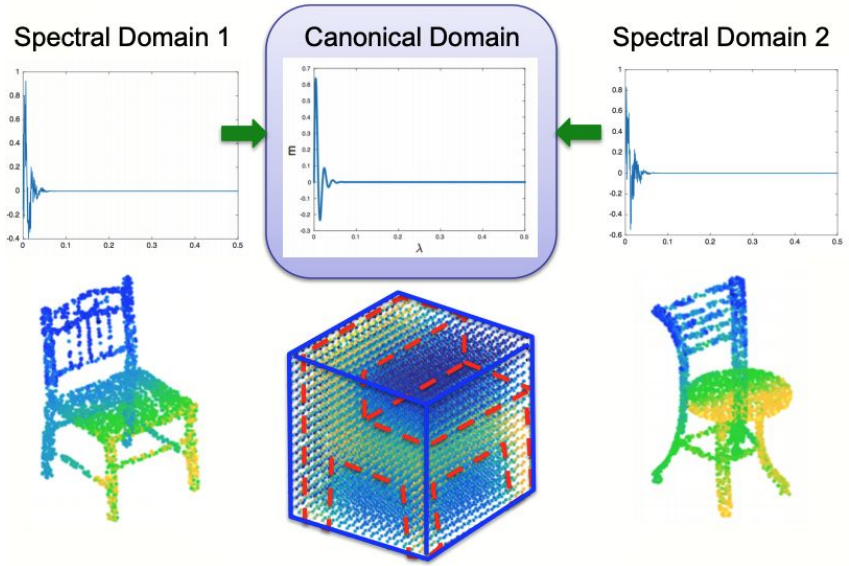
- Fundamental Challenge of Spectral CNN
  - If the shapes to compare are not isometric, their spectral domains are not aligned
  - Function bases are derived by Laplacian operator, which is geometry dependent



Yi et al., "**SyncSpecCNN: Synchronized Spectral CNN for 3D Shape Segmentation**", *CVPR 2017*

Spectral Convolution

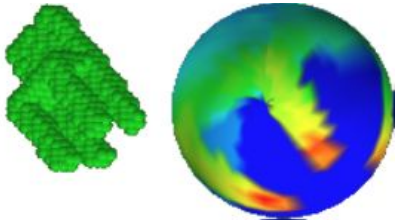
- Domain Synchronization
  - Rescue: Synchronize spectral domains by functional maps



Yi et al., "**SyncSpecCNN: Synchronized Spectral CNN for 3D Shape Segmentation**", *CVPR 2017*

Spectral Convolution

- A Special Case: Spherical CNN
  - If the surface is always a SPHERE, no worry about the functional space alignment anymore
  - Generate a spherical representation

input  convert input to a function on sphere (spherical repr.)

  - Do Spherical CNN
    - Has numerical tricks exploiting the symmetry of sphere

Cohen et al., "**Spherical CNN**", *ICLR 2018*
Esteves et al., "**Learning SO(3) Equivariant Representations with Spherical CNNs**", *ECCV 2018*

Spectral Convolution

- Learned Filters

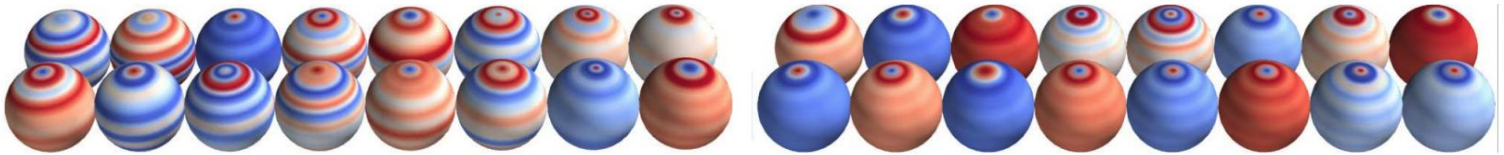


Fig. 4: Filters learned in the first layer. The filters are zonal. *Left:* 16 nonlocalized filters. *Right:* 16 localized filters. Nonlocalized filters are parameterized by all spectral coefficients (16, in the example). Even though locality is not enforced, some filters learn to respond locally. Localized filters are parameterized by a few points of the spectrum (4, in the example), the rest of the spectrum is obtained by interpolation.

Cohen et al., "**Spherical CNN**", *ICLR 2018*
Esteves et al., "**Learning SO(3) Equivariant Representations with Spherical CNNs**", *ECCV 2018*

Spectral Convolution

- Rotation invariance guaranteed
- Can be used to improve the rot. invariance of MVCNN, as well

Table 1: ModelNet40 classification accuracy per instance. Spherical CNNs are robust to arbitrary rotations, even when not seen during training, while also having one order of magnitude fewer parameters and faster training.

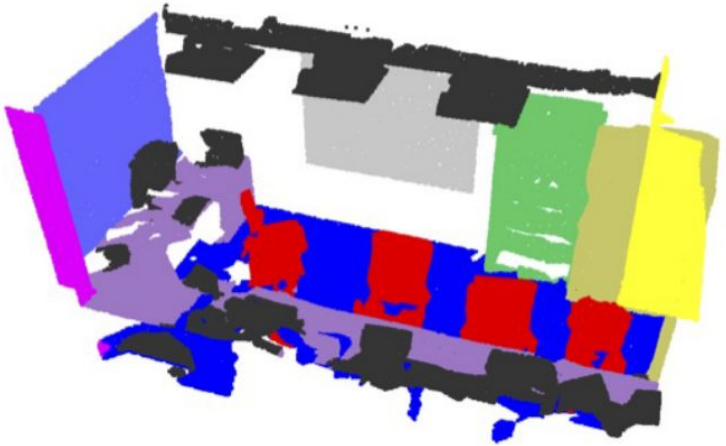
Method	z/z	SO3/SO3	z/SO3	params	inp. size
PointNet [7]	89.2	83.6	14.7	3.5M	2048 x 3
PointNet++ [38]	89.3	85.0	28.6	1.7M	1024 x 3
VoxNet [29]	83.0	73.0	-	0.9M	30^3
SubVolSup [8]	88.5	82.7	36.6	17M	30^3
SubVolSup MO [8]	89.5	85.0	45.5	17M	20×30^3
MVCNN 12x [9]	89.5	77.6	70.1	99M	12×224^2
MVCNN 80x [9]	**90.2**	86.0	- [2]	99M	80×224^2
RotationNet 20x [30]	**92.4**	80.0	20.2	58.9M	20×224^2
Ours	88.9	**86.9**	**78.6**	**0.5M**	$\mathbf{2 \times 64^2}$

Esteves et al., "**Learning SO(3) Equivariant Representations with Spherical CNNs**", *ECCV 2018*
Esteves et al., "**Equivariant Multi-View Networks**", *ICCV 2019*

Outline

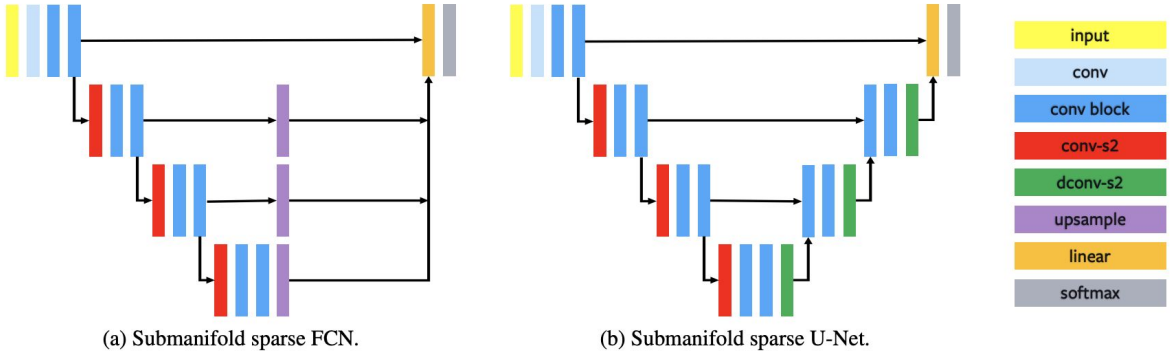
- 3D Data
- Classification
- **Segmentation and Detection**
- 3D Data synthesis

Task: 3D Semantic Segmentation



Task: 3D Semantic Segmentation

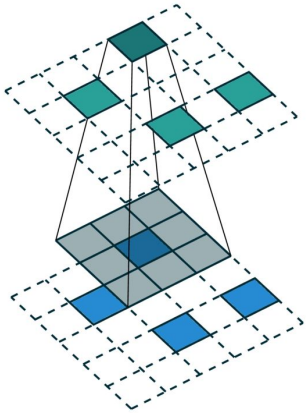
- Encoder – Decoder



(a) Submanifold sparse FCN.

(b) Submanifold sparse U-Net.

Sparse Convolution



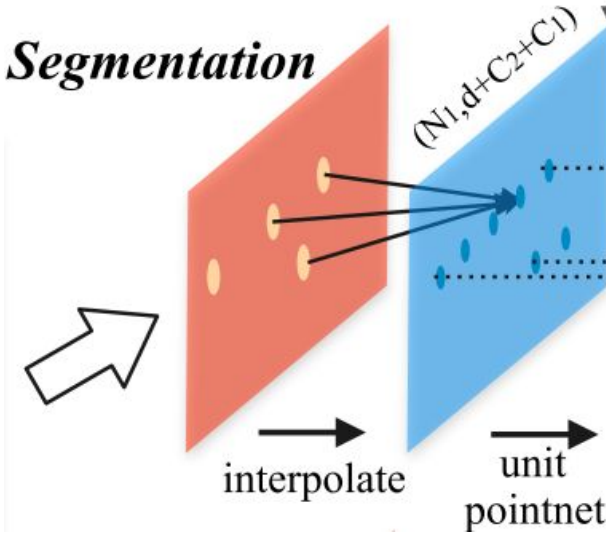
Sparse Deconvolution

Graham, Benjamin, Martin Engelcke, and Laurens van der Maaten. **"3d semantic segmentation with submanifold sparse convolutional networks."** CVPR 2018.

Choy, Christopher, et al. **"4D Spatio-Temporal Convnets: Minkowski Convolutional Neural Networks."** CVPR 2019

Task: 3D Semantic Segmentation

- Decoder



Upsample point cloud features by interpolating from 3 nearest points in the point cloud of lower resolution.

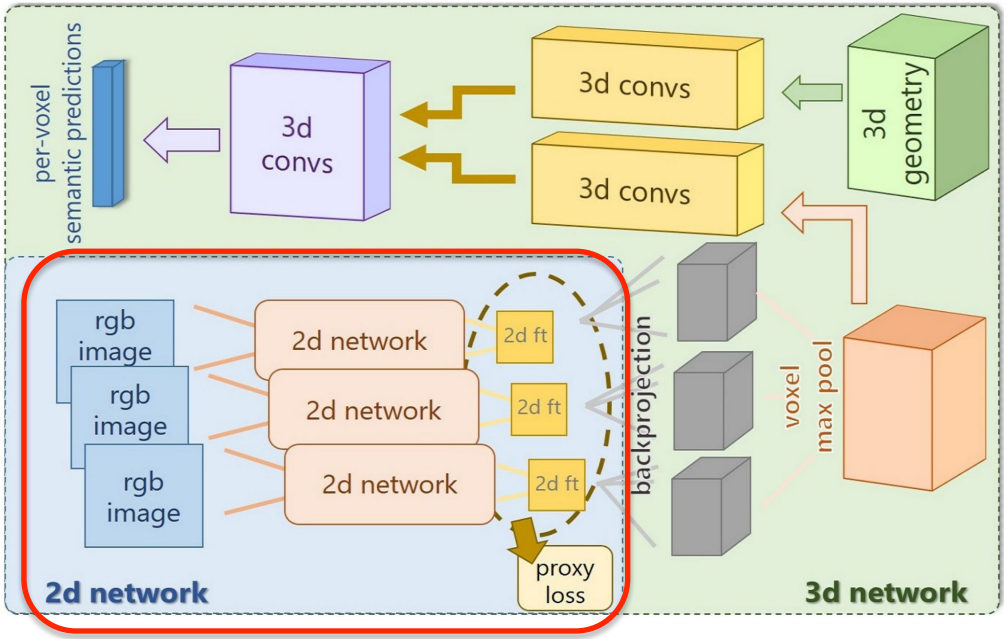
Lower resolution
Fewer points

Higher resolution
More points

Qi, Charles R., et al. **"PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space."**, NeurIPS 2017

Task: 3D Semantic Segmentation

- Multimodal



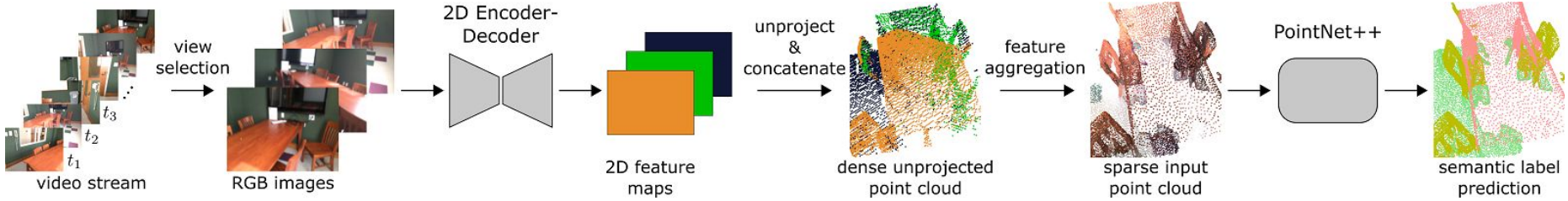
Backproject 2D features to 3D voxels

Apply voxel-wise max-pooling across multiple views

Fuse 2D and 3D features at the intermediate level

Dai, Angela, and Matthias Nießner. **"3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation."**, ECCV 2018

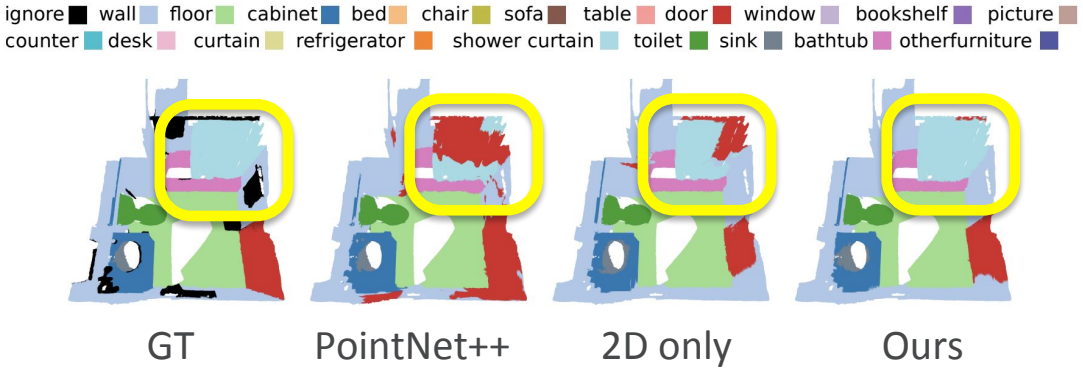
Task: 3D Semantic Segmentation



Backproject 2D features to 3D points

Apply PointNet to aggregate multi-view features

Fuse 2D and 3D features as input to 3D network

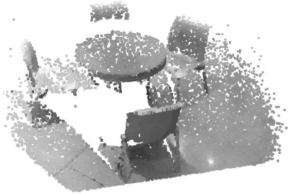


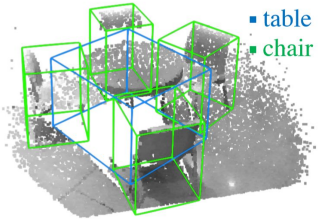
Task: Instance-level Understanding

Input

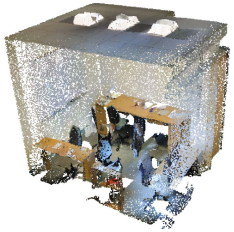
Output

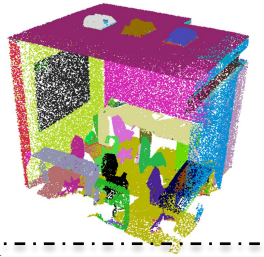
Object Detection





Object Segmentation





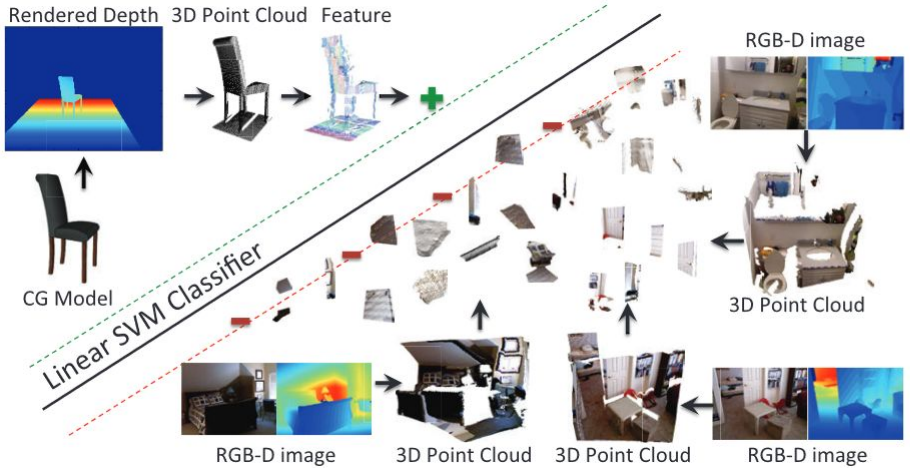
Part Segmentation

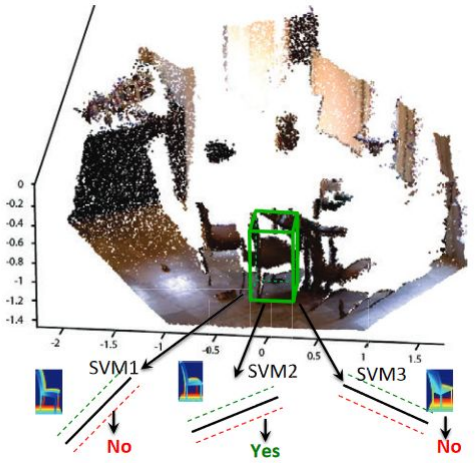
Task: 3D Detection & Instance Segmentation

- Top-down Methods
  - Does this point cloud contain an object?
  - How to select point clouds to classify objectness?

Task: 3D Detection & Instance Segmentation

- Sliding Shapes



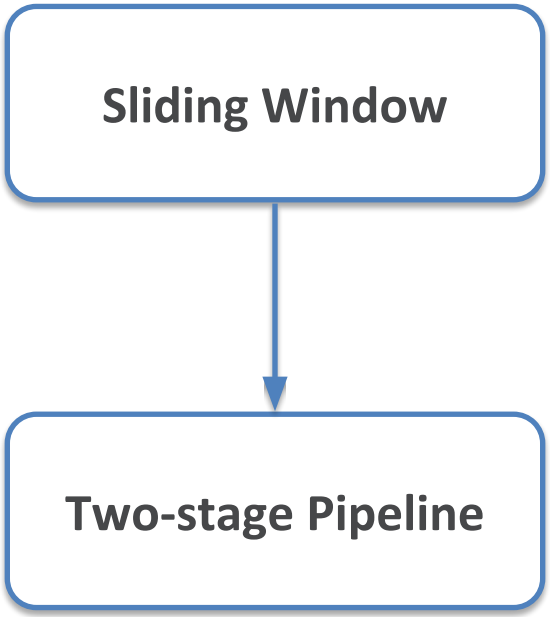


Sliding window to walk over the entire space

Expensive !

Song et al., "**Sliding Shapes for 3D Object Detection in Depth Images**", *ECCV 2014*

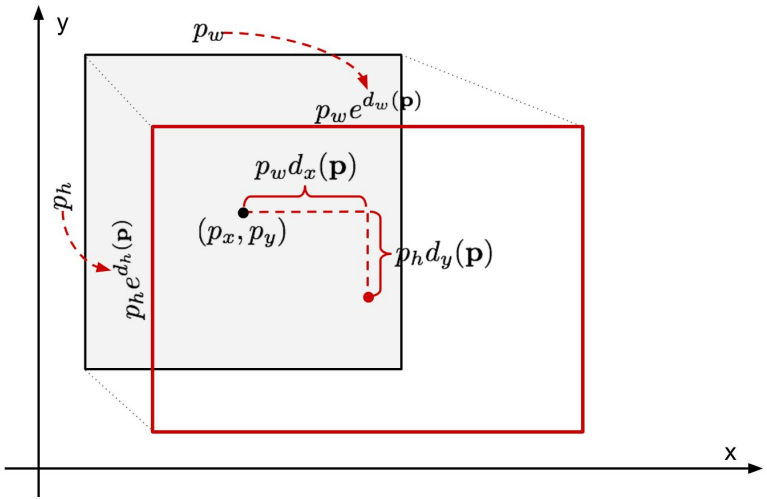
Task: 3D Detection & Instance Segmentation



First stage: Proposal
Second stage: Refinement

Localization

- 3D bounding box regression
- Predefined bounding boxes (anchors) at each sliding window
- Re-parameterize as relative offsets



$$\Delta_x = \frac{\mu - \mu_{anchor}}{\phi_{anchor}}$$

Center shift

$$\Delta_w = \ln(\frac{\phi}{\phi_{anchor}})$$

Size scale

Figure from https://lilianweng.github.io/lil-log/2017/12/31/object-recognition-for-dummies-part-3.html

From Box to Instance Segmentation

- Box includes background points or other instances
- foreground/background segmentation

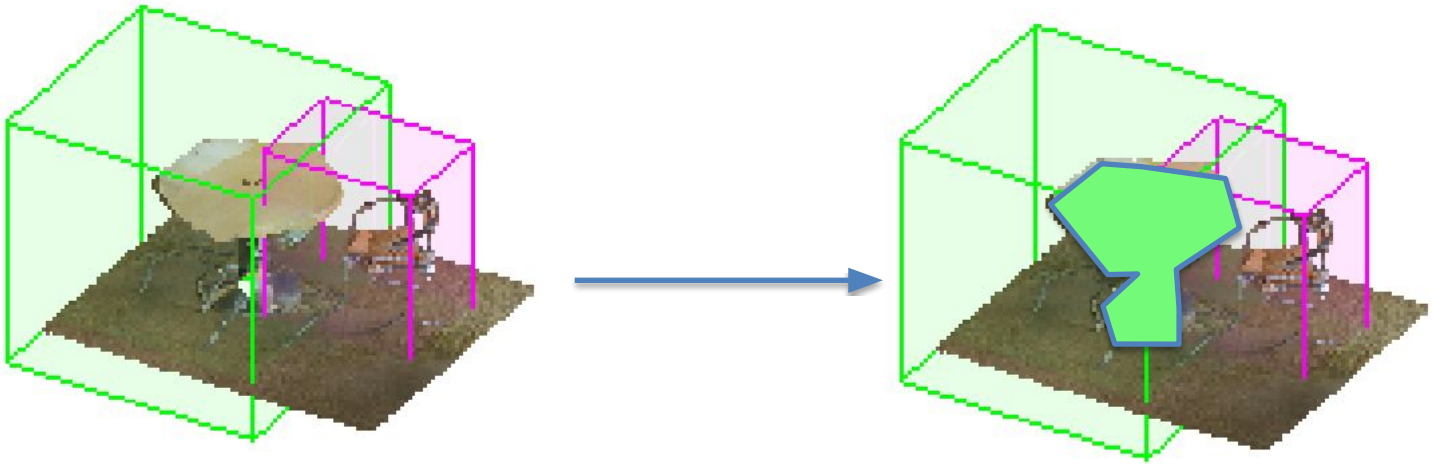
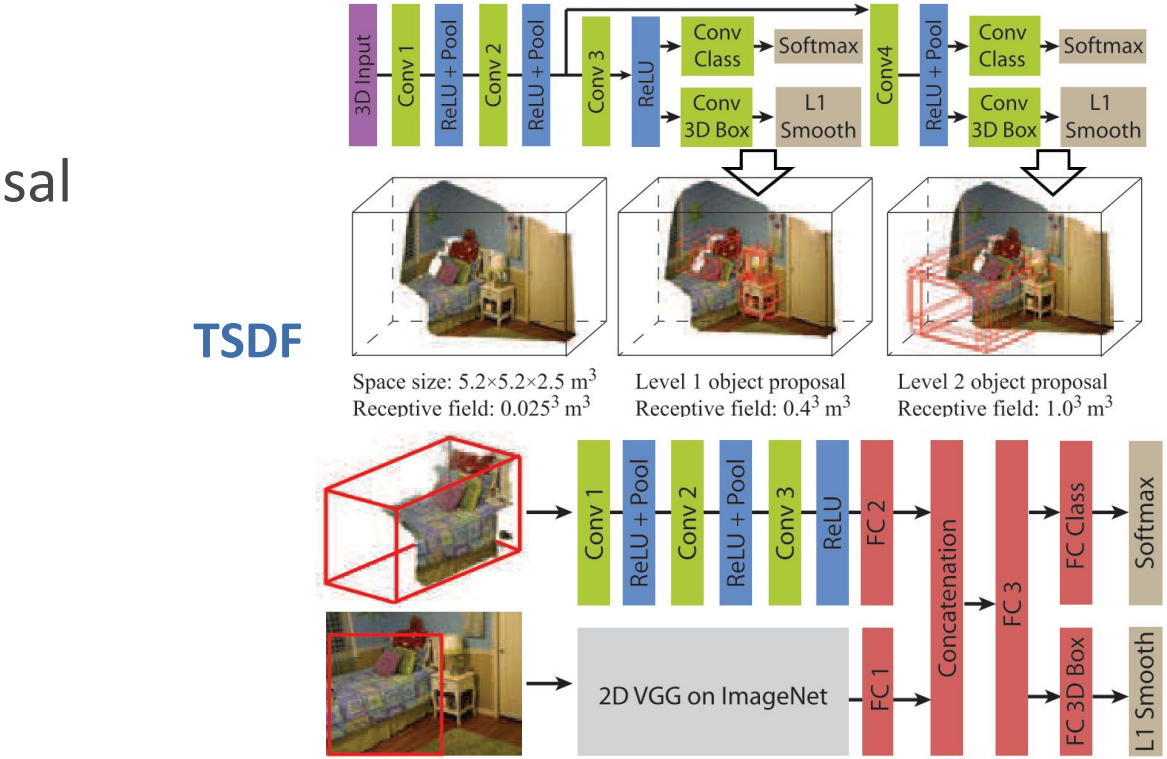


Figure from **“Learning Object Bounding Boxes for 3D Instance Segmentation on Point Clouds”**

Volumetric R-CNN

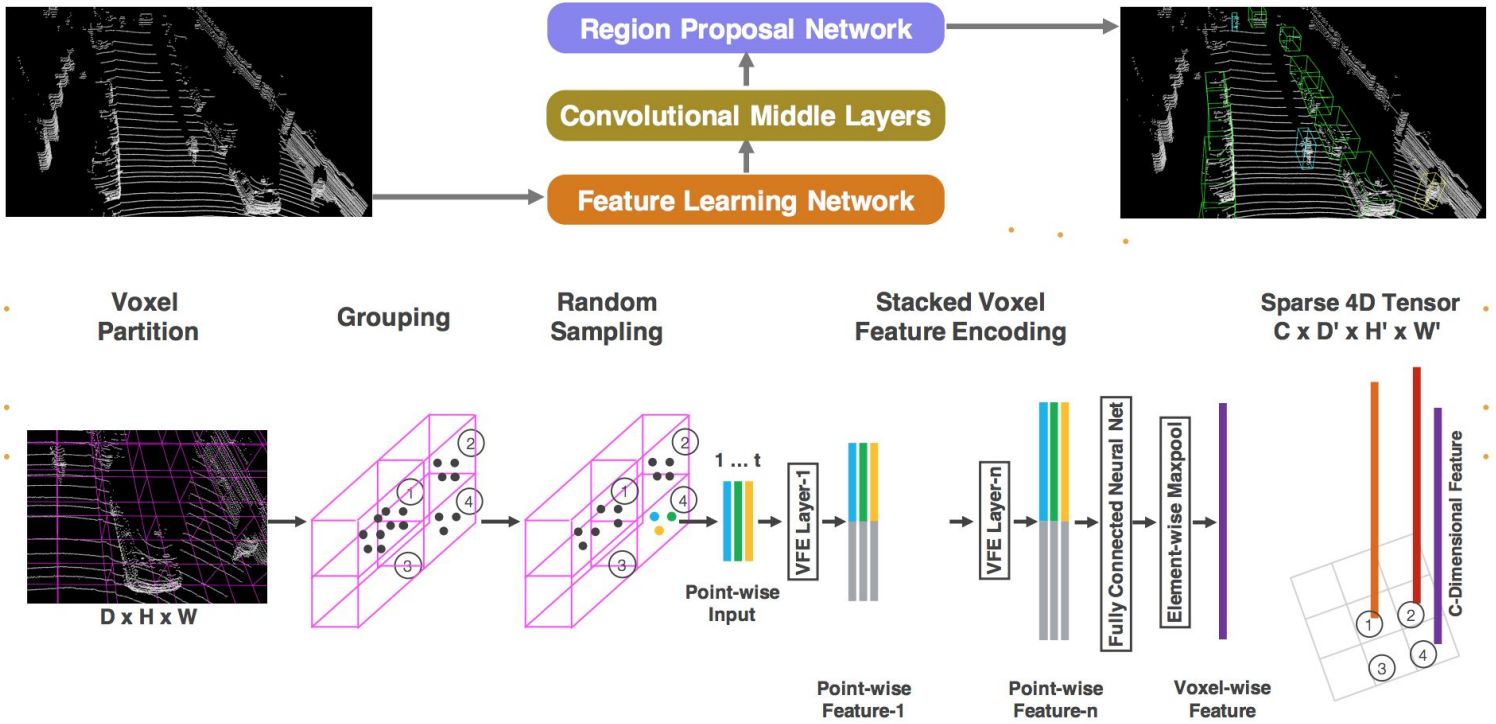
Stage1: 3D Region Proposal Network

Stage2: Joint Object Recognition Network



Song et al., "**Deep Sliding Shapes for Amodal 3D Object Detection in RGB-D Images**", *CVPR 2016*

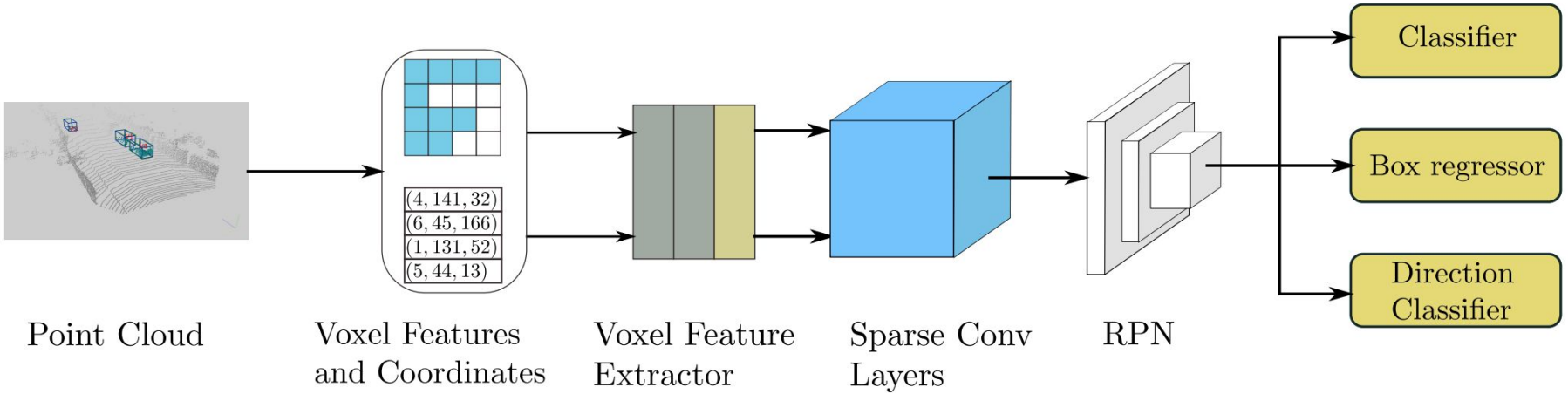
VoxelNet: 3D Convolution



Zhou et al., "**VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection**", *CVPR 2018*

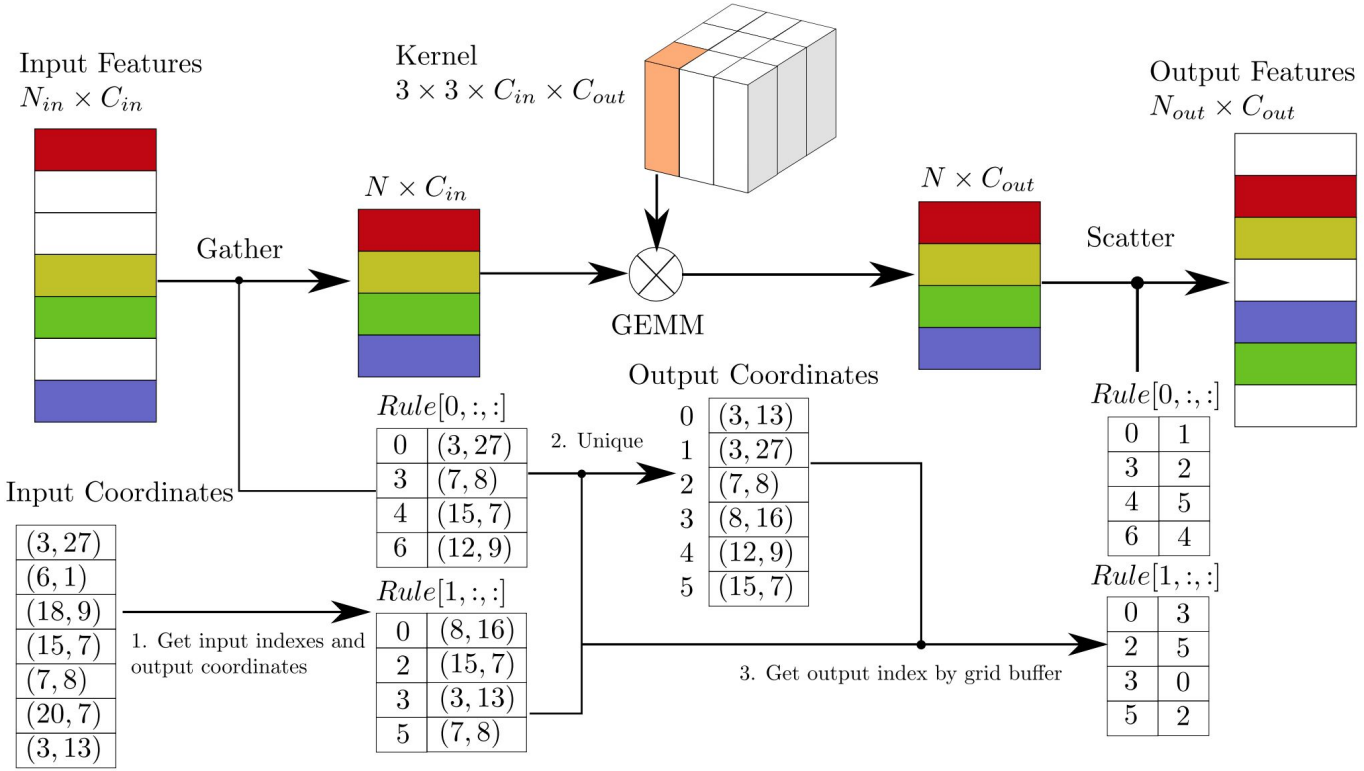
SECOND: Sparse 3D Convolution

- Sparse 3D Convolution:
  - Reduce complexity
  - Avoid computation on empty voxels



Yan, Yan, et al. **"Second: Sparsely Embedded Convolutional Detection."** Sensors 2018

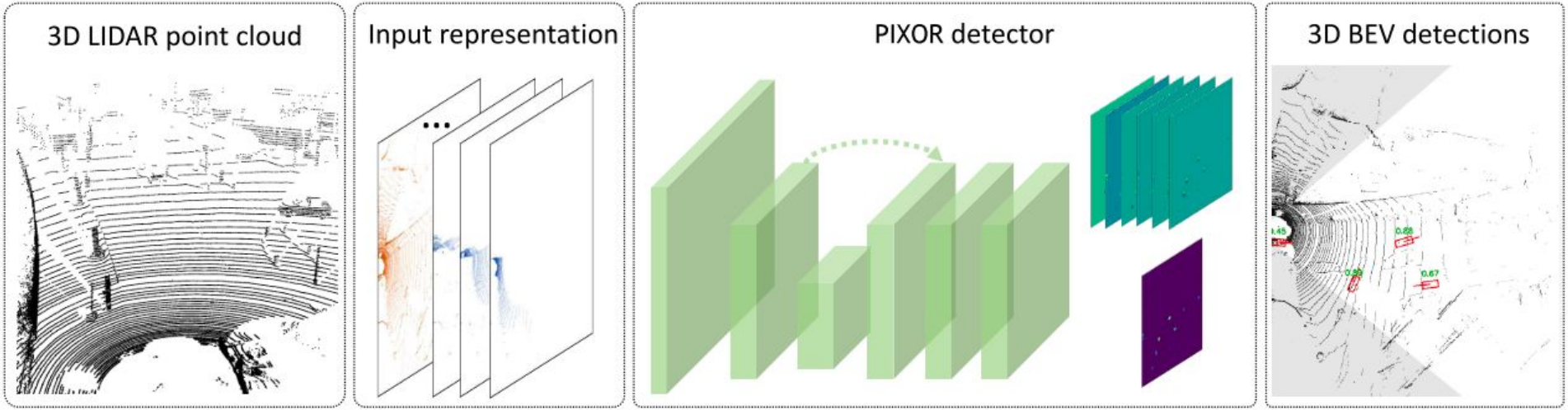
SECOND: 3D Sparse Convolution



Yan, Yan, et al. **"Second: Sparsely Embedded Convolutional Detection."** Sensors 2018

BEV (bird-eye view)

- Top View Representation (2D Projection)



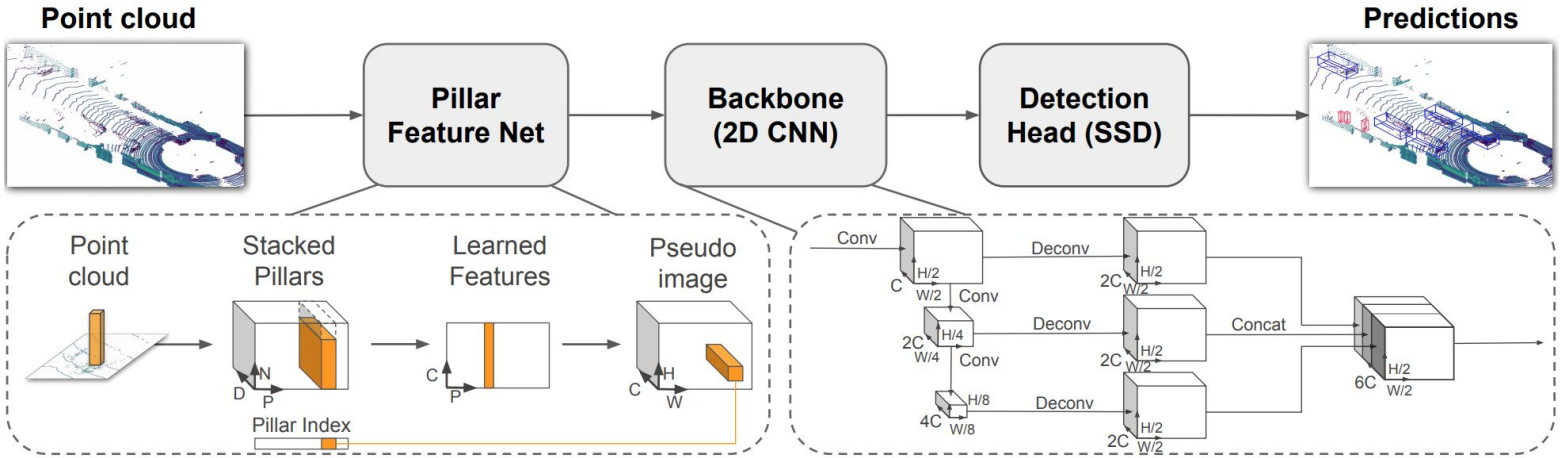
HxWxD voxels

HxW image with
D channels

Yang, Bin, Wenjie Luo, and Raquel Urtasun. **"Pixor: Real-time 3d object detection from point clouds."** CVPR 2018

PointPillars: Bird Eye View Convolution

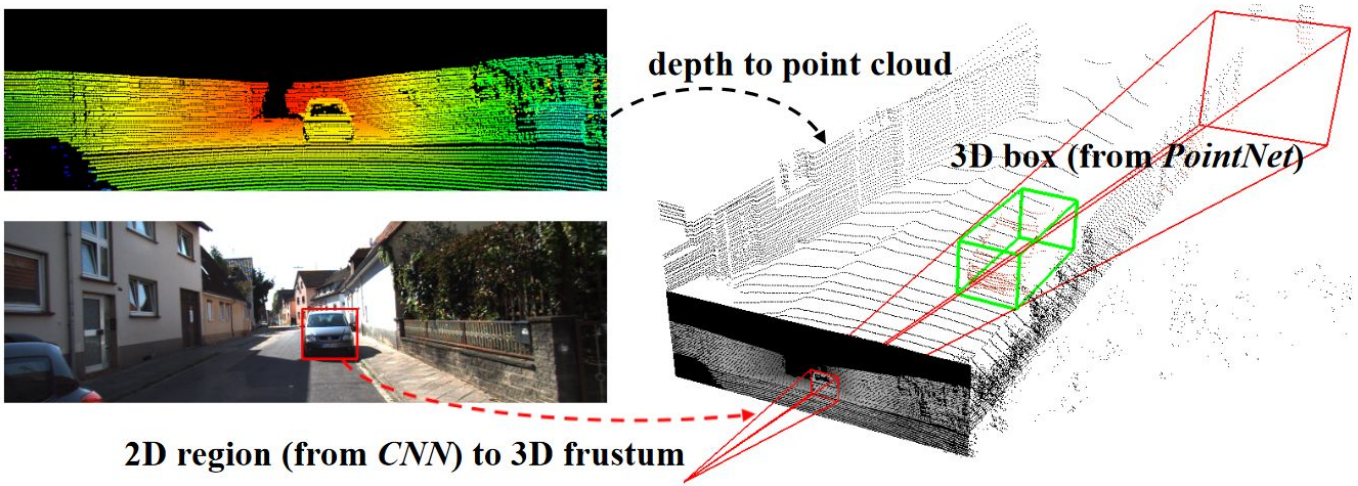
- 2D Convolution from top view:
  - Replace occupancy of each pixel with point cloud features of the pillar



Lang, Alex H., et al. **"Pointpillars: Fast Encoders for Object Detection from Point Clouds."** CVPR 2019

Frustum PointNets: View-based Proposal

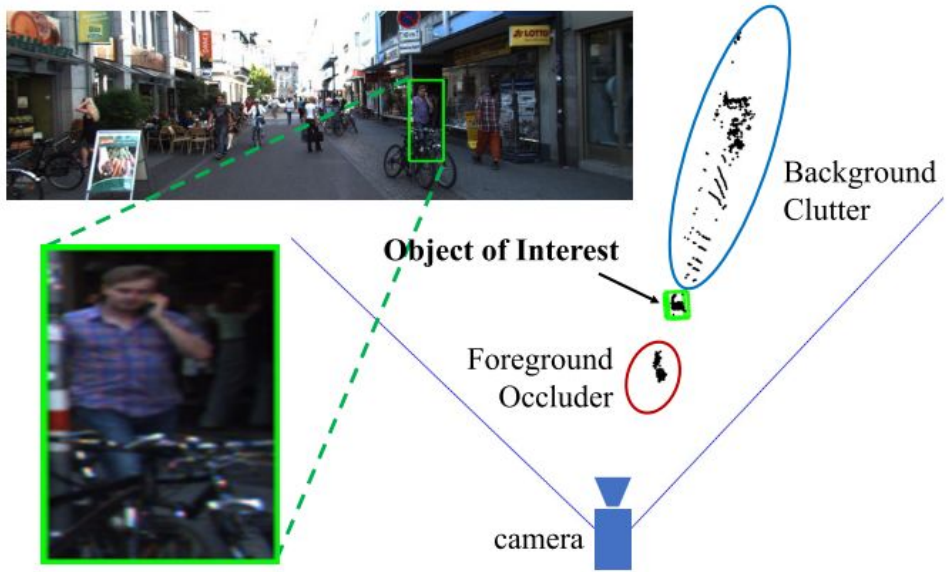
- Generate object proposals from a view (e.g., using SSD)



Qi et al., "**Frustum PointNets for 3D Object Detection from RGB-D Data**", *CVPR 2018*

Frustum PointNets: FG/BG Segmentation

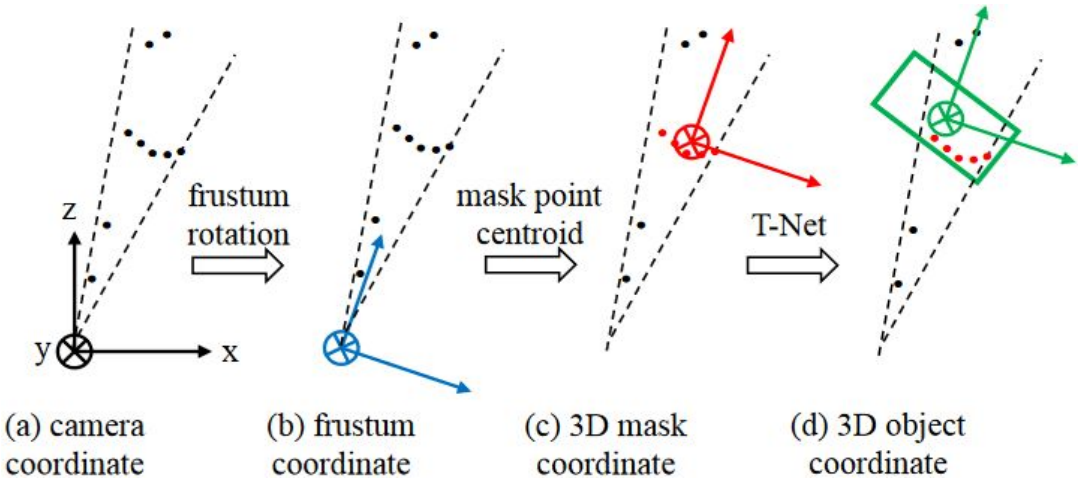
- Stage 2: FG/BG Segmentation
  - Instance segmentation to remove other foreground instances and background clutter



Qi et al., "**Frustum PointNets for 3D Object Detection from RGB-D Data**", *CVPR 2018*

Frustum PointNets: Coordinate Normalization

- Instance segmentation to remove other foreground instances and background clutter

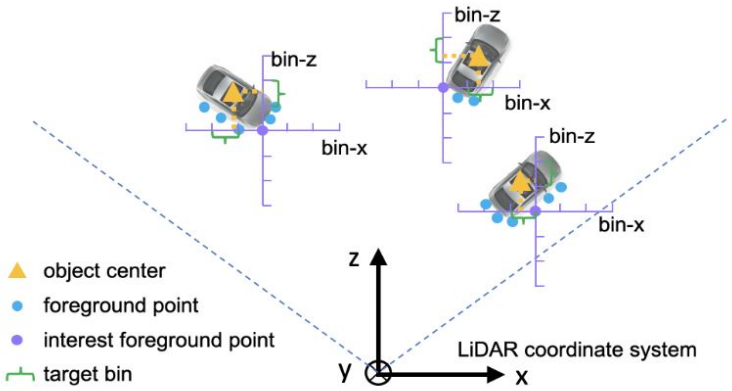


Qi et al., "**Frustum PointNets for 3D Object Detection from RGB-D Data**", *CVPR 2018*

PointRCNN: Region Proposal Network

Stage-1: Foreground/Background segmentation and generate 3D proposals for each foreground point

Stage-2: Refine proposals in the canonical coordinates

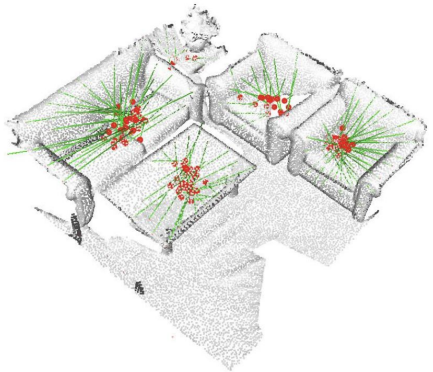


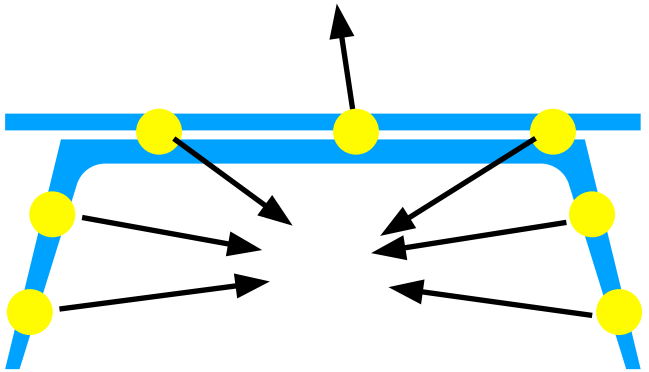
Shi et al., "**PointRCNN: 3D Object Proposal Generation and Detection from Point Cloud**", *CVPR 2019*

VoteNet: Proposal from Voting

- **Challenge:** 3D object centroid can be far from any surface point, thus hard to regress accurately
  - Sample a set of seed points and generate votes, targeting at object centers

Voting from input point cloud



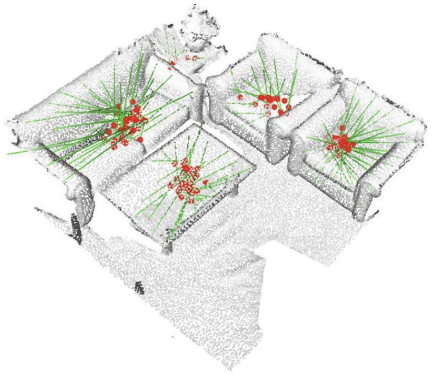


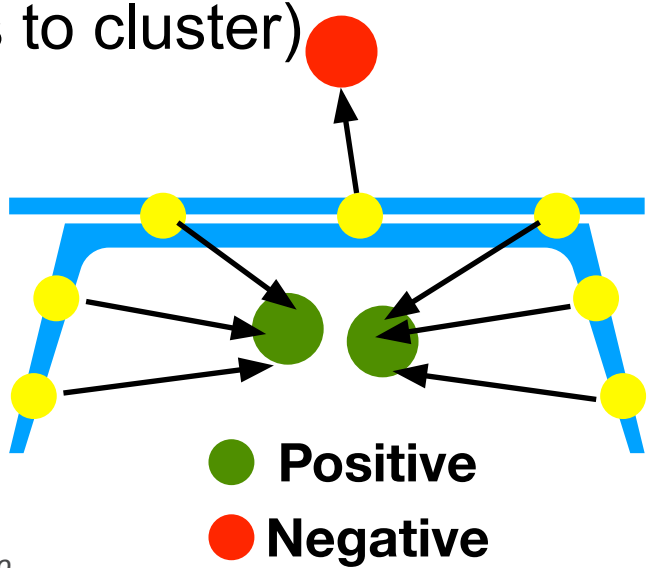
Qi et al., "**Deep Hough Voting for 3D Object Detection in Point Clouds**", *ICCV 2019*

VoteNet: Proposal from Voting

- **Challenge:** 3D object centroid can be far from any surface point, thus hard to regress accurately
  - Votes cluster near object centers (Farthest point sample votes to cluster)

Voting from input point cloud

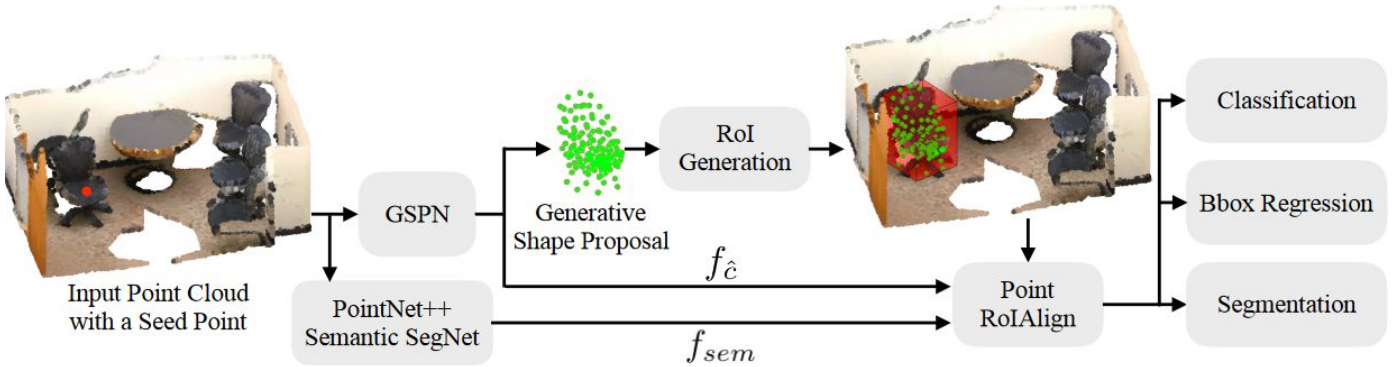




Qi et al., "**Deep Hough Voting for 3D Object Detection in Point Clouds**", *ICCV 2019*

GSPN : Proposal from Generative Network

- Randomly sample seeds points
- Take point cloud and a seed point as input, use conditional VAE to generate a point cloud as proposal
- Convert the proposal to an ROI box
- R-PointNet (mask RCNN) to segment the object



Yi et al., "**GSPN: Generative Shape Proposal Network for 3D Instance Segmentation in Point Cloud**", *CVPR 2019*

Task: 3D Detection & Instance Segmentation

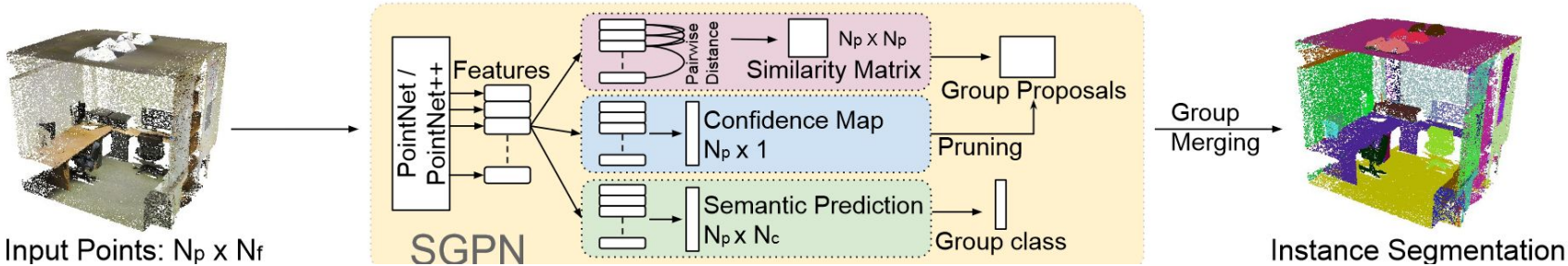
- Bottom-Up Methods
  - Do these points belong to the same instance?
  - How to measure the similarity between two points?

Associative Embedding

- Learn a per-point embedding, so that points from the same instance have similar embeddings

$$l(i,j) = \begin{cases} ||F_{SIM_i} - F_{SIM_j}||_2 & C_{ij} = 1 \\ \alpha \max(0, K_1 - ||F_{SIM_i} - F_{SIM_j}||_2) & C_{ij} = 2 \\ \max(0, K_2 - ||F_{SIM_i} - F_{SIM_j}||_2) & C_{ij} = 3 \end{cases}$$

- Clustering gives proposals

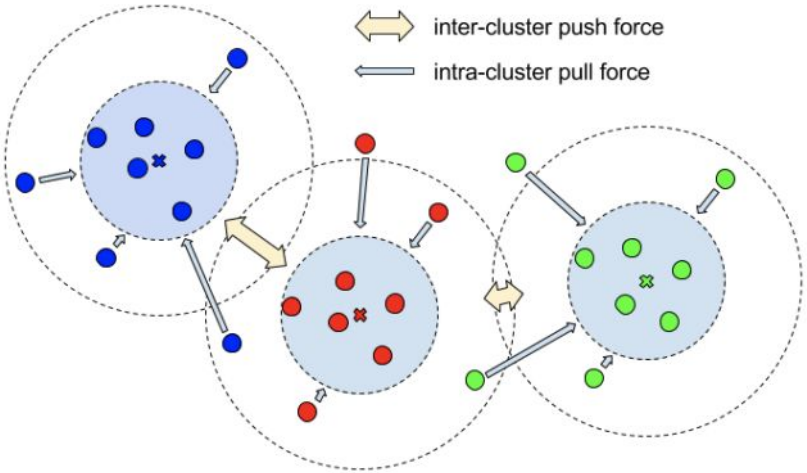


Wang et al., "**SGPN: Similarity Group Proposal Network for 3D Point Cloud Instance Segmentation**", *CVPR 2018*

JSIS3D: “push/pull” forces

- A discriminative function to present the embedding loss

$$\mathcal{L}_{embedding} = \alpha \cdot \mathcal{L}_{pull} + \beta \cdot \mathcal{L}_{push} + \gamma \cdot \mathcal{L}_{reg}$$



$$\mathcal{L}_{pull} = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{N_k}\sum_{j=1}^{N_k}\left[\|\boldsymbol{\mu}_k - \mathbf{e}_j\|_2 - \delta_v\right]_+^2$$

$$\mathcal{L}_{push} = \frac{1}{K(K-1)}\sum_{k=1}^{K}\sum_{m=1,m\neq k}^{K}\left[2\delta_d - \|\boldsymbol{\mu}_k - \boldsymbol{\mu}_m\|_2\right]_+^2$$

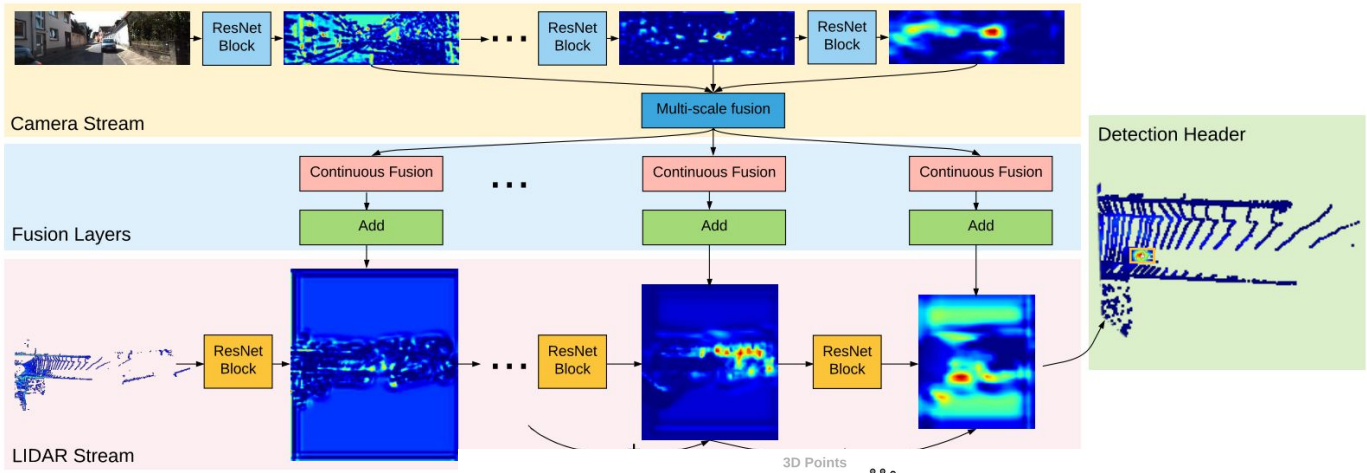
$$\mathcal{L}_{reg} = \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{\mu}_k\|_2$$

Pham, Quang Hieu, et al. **“JSIS3D: Joint Semantic-Instance Segmentation of 3D Point Clouds with Multi-Task Pointwise Networks and Multi-Value Conditional Random Fields.”**, CVPR 2019

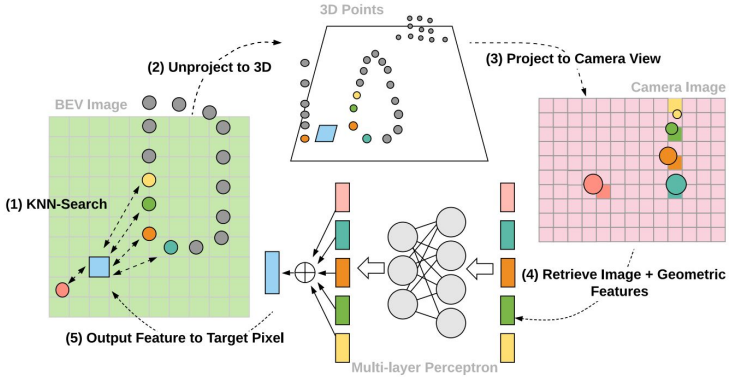
Multi-Modal Continuous Fusion

Image feature network: 2D image

BEV network: 3D voxel (HxWxC)



3D points as the intermediate media to transform from the camera to BEV



Liang, Ming, et al. **"Deep Continuous Fusion for Multi-Sensor 3D Object Detection."** ECCV 2018

Task: Few-shot/Zero-shot Learning

- Can be a better platform than images

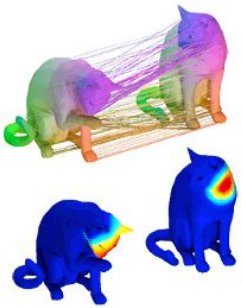
- Shapes are **pure,** with no contamination by projection distortion, illumination, …

If one image is more than a thousand words, then one shape is more than one thousand pictures

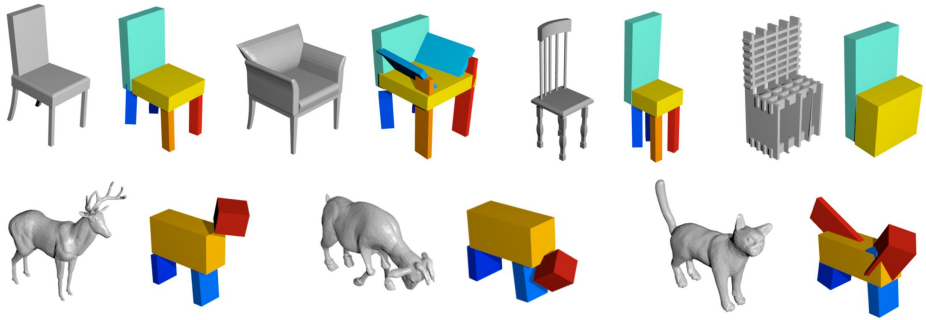
Task: Few-shot/Zero-shot Learning

Algorithmically, 3D shapes are:

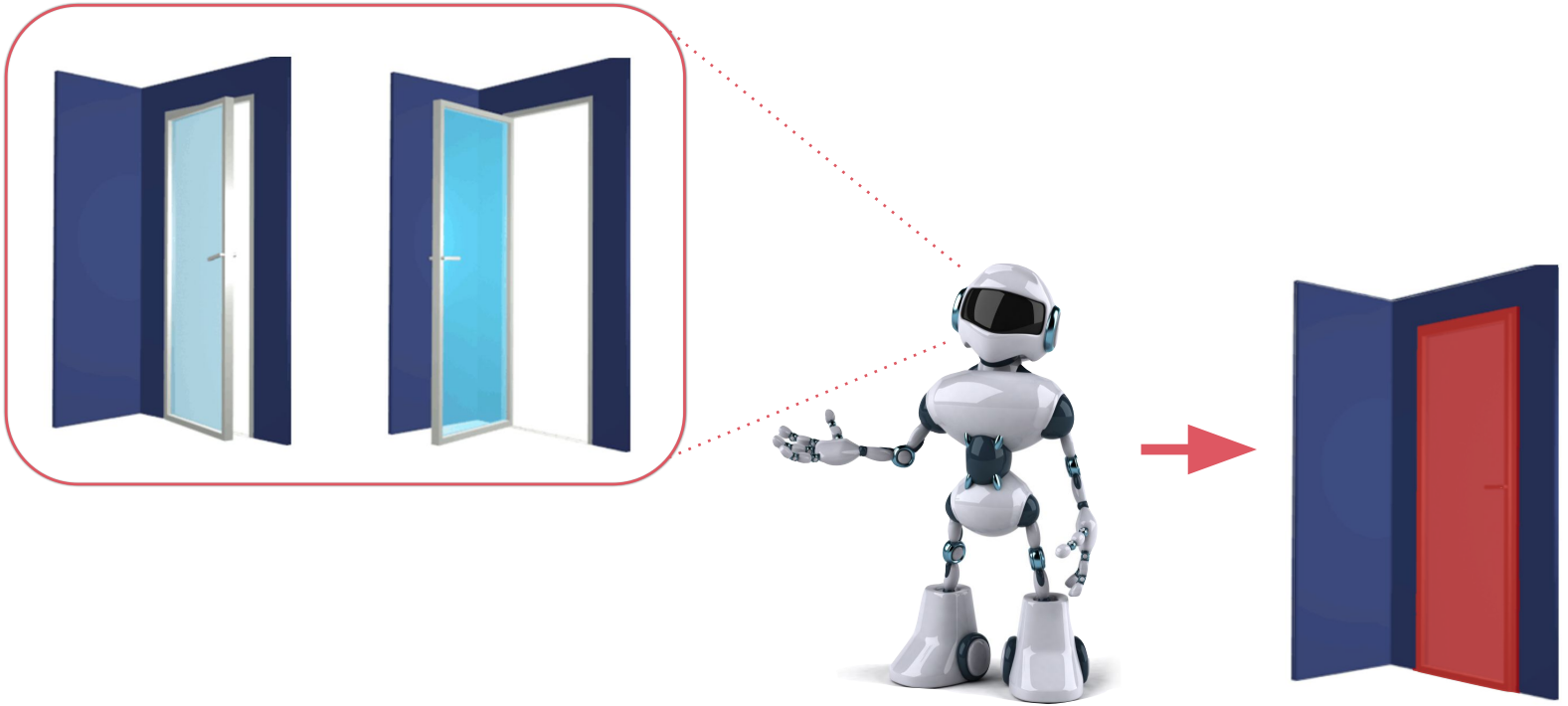
- easier to be **related (correspondence)**



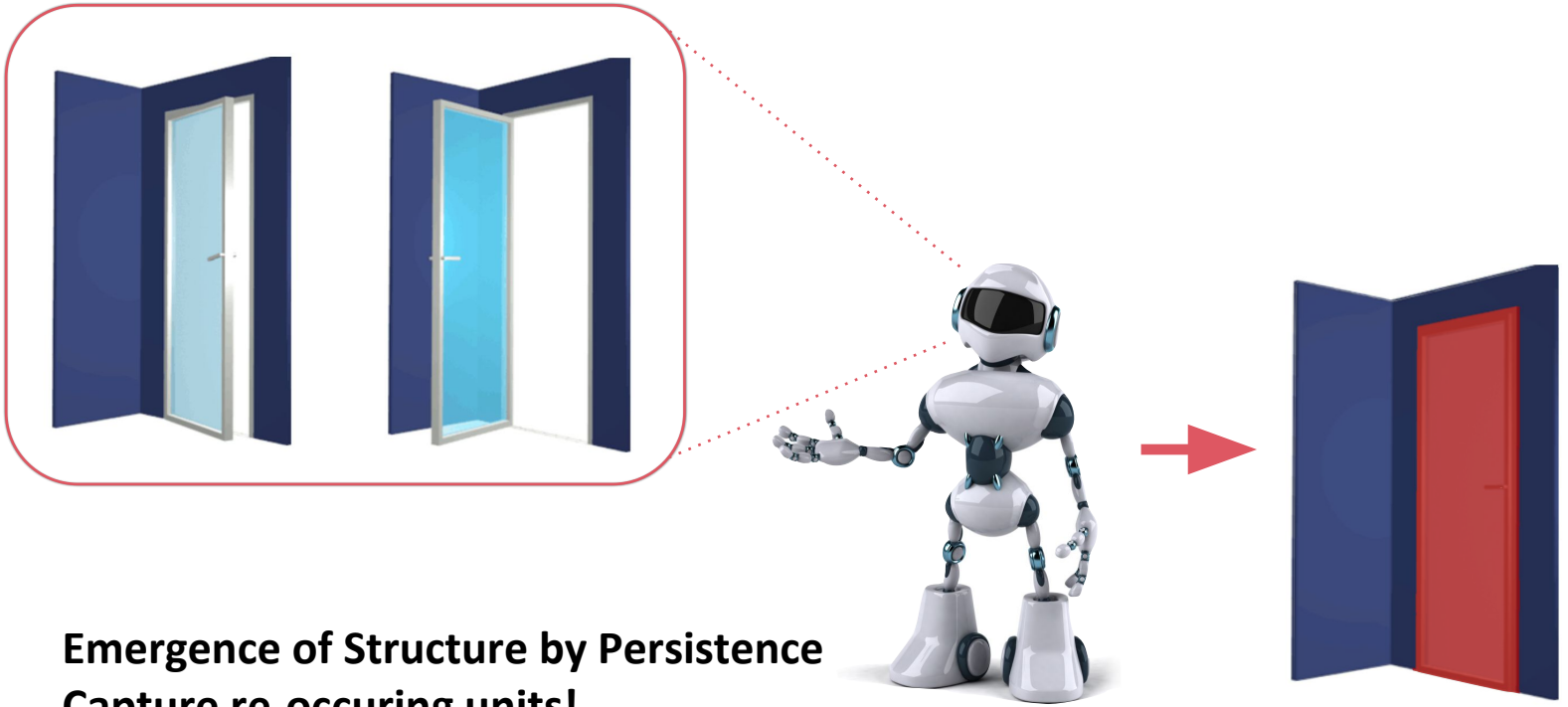
- easier to be **compared**
- easier to **abstracted**



Task: Few-shot Structure Induction

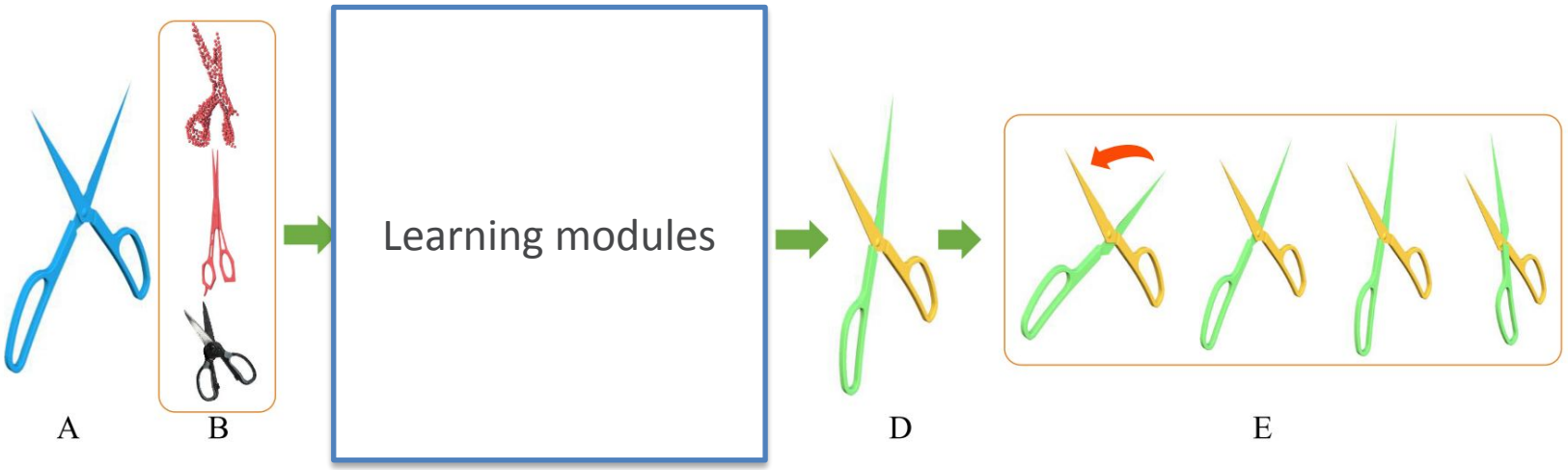


Task: Few-shot Structure Induction



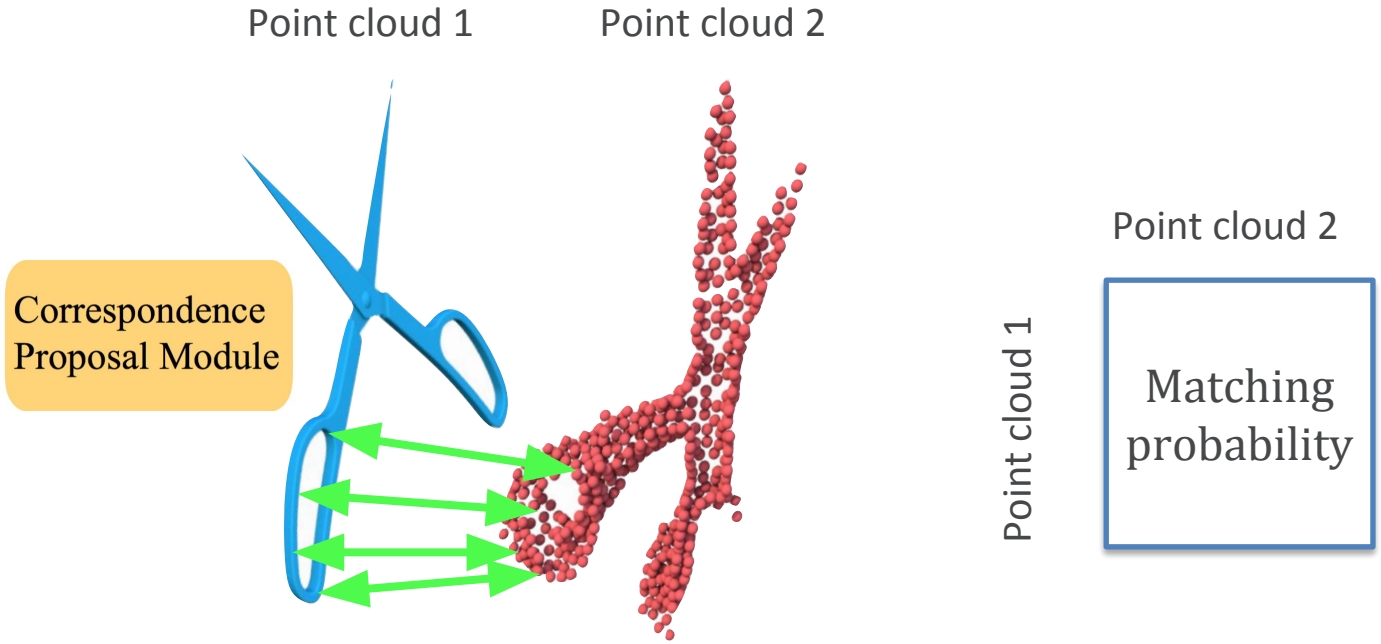
Emergence of Structure by Persistence
Capture re-occuring units!

Part Induction by Relating Shapes



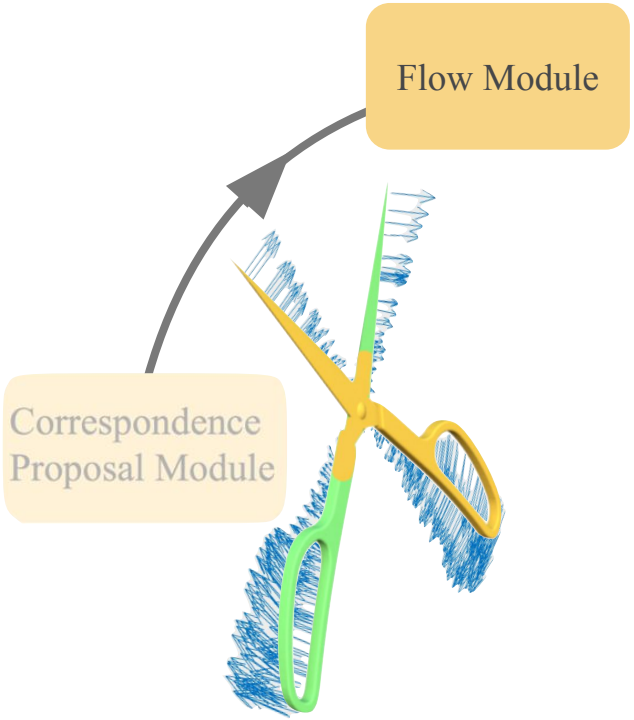
Yi et al., "**Deep Part Induction from Articulated Object Pairs**", *SIGGRAPH Asia 2018*

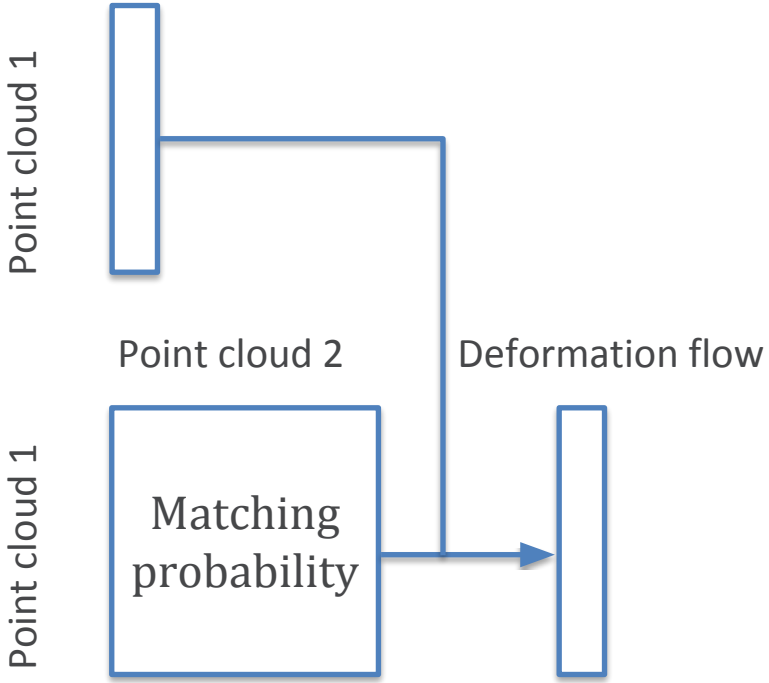
Part Induction by Relating Shapes



Yi et al., "**Deep Part Induction from Articulated Object Pairs**", *SIGGRAPH Asia 2018*

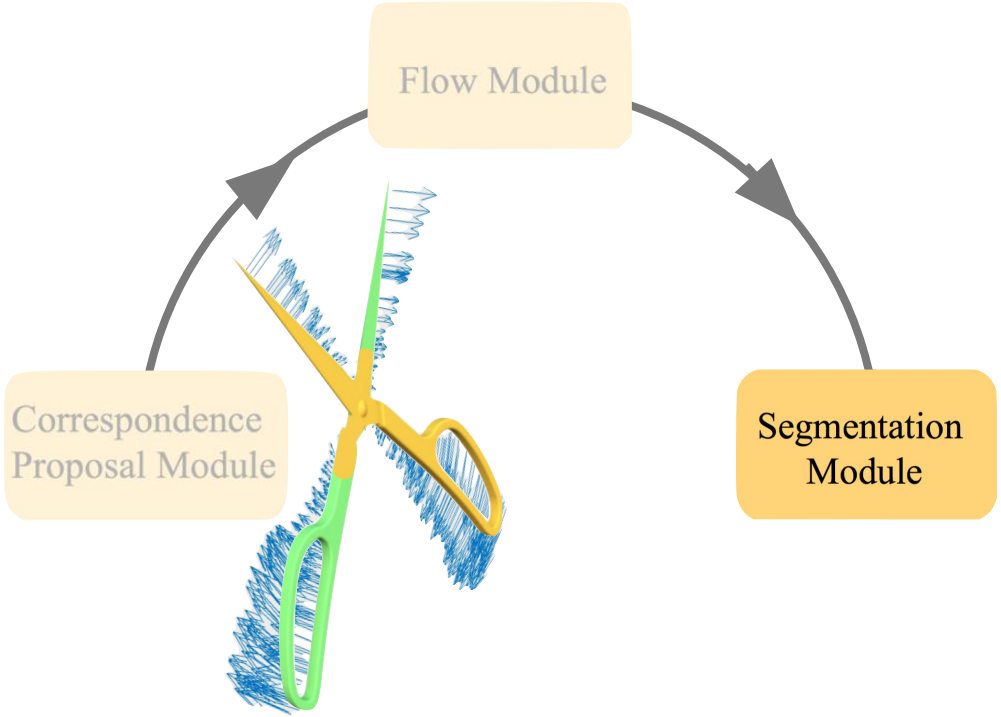
Part Induction by Relating Shapes





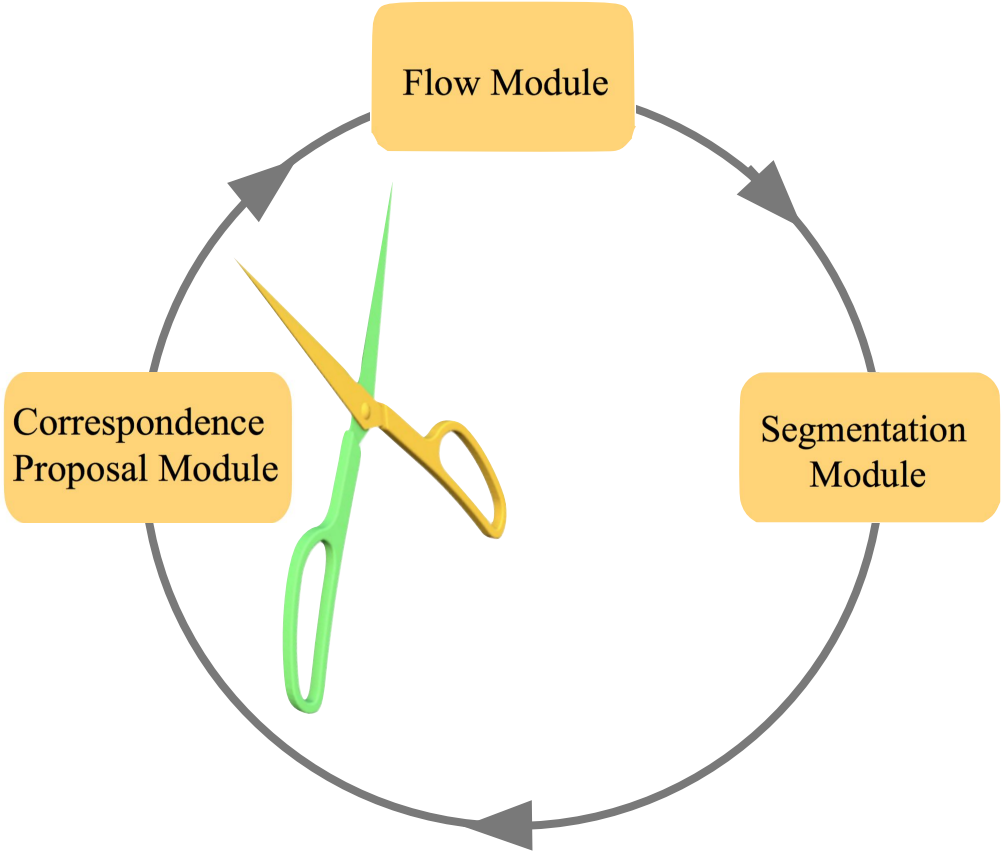
Yi et al., "**Deep Part Induction from Articulated Object Pairs**", *SIGGRAPH Asia 2018*

Part Induction by Relating Shapes



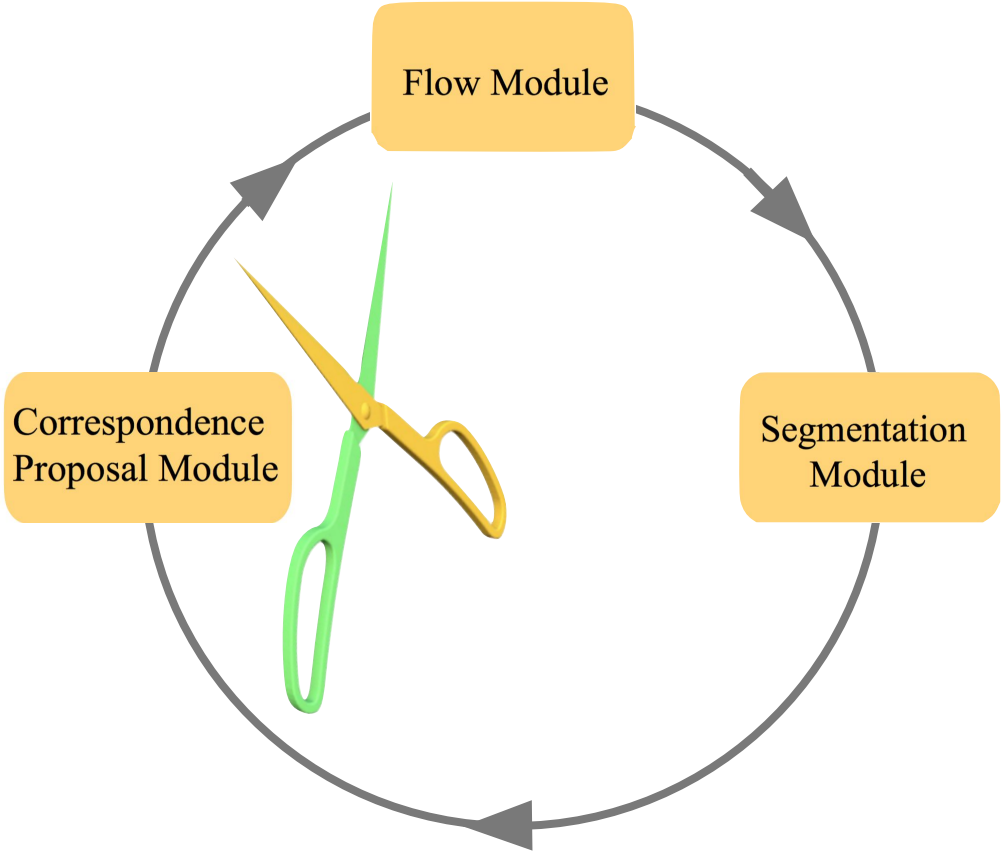
Yi et al., "**Deep Part Induction from Articulated Object Pairs**", *SIGGRAPH Asia 2018*

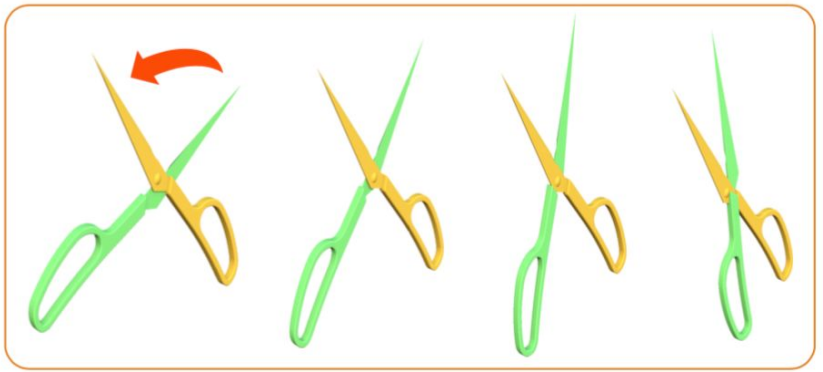
Part Induction by Relating Shapes



Yi et al., "**Deep Part Induction from Articulated Object Pairs**", *SIGGRAPH Asia 2018*

Part Induction by Relating Shapes



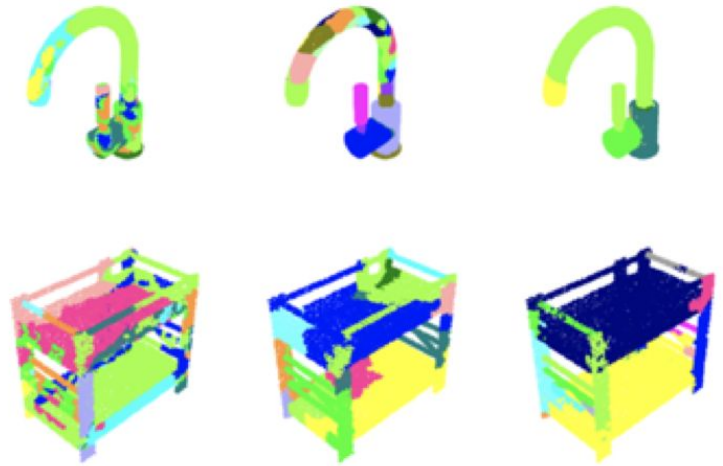


Yi et al., "**Deep Part Induction from Articulated Object Pairs**", *SIGGRAPH Asia 2018*

Task: Zero-shot Part Discovery

Train set

Test set



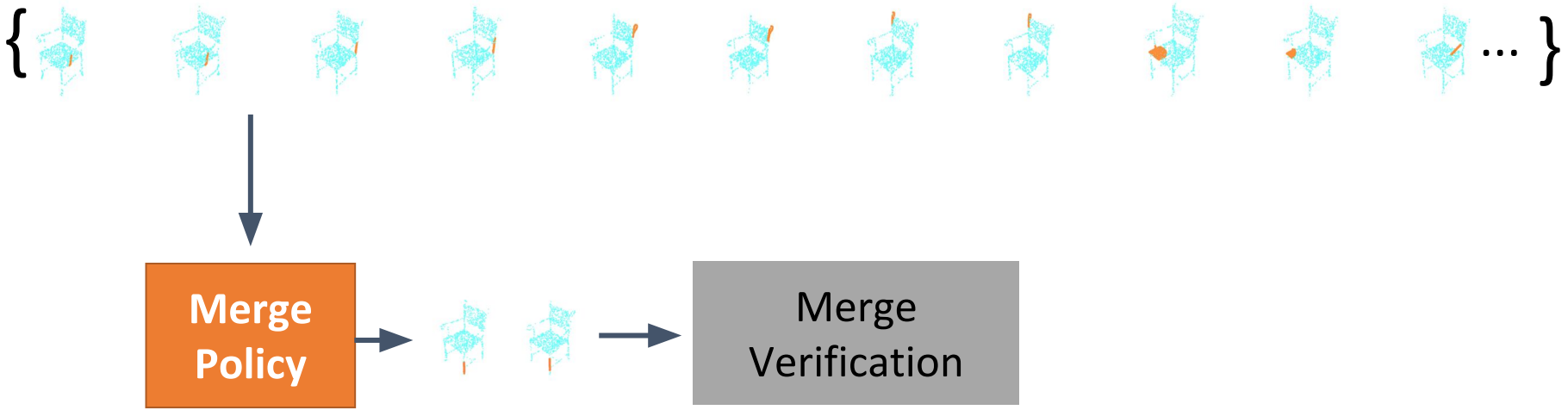
SOTA of Deep Learning (PartNet)

SOTA of Classics (WCSeg)

Ours

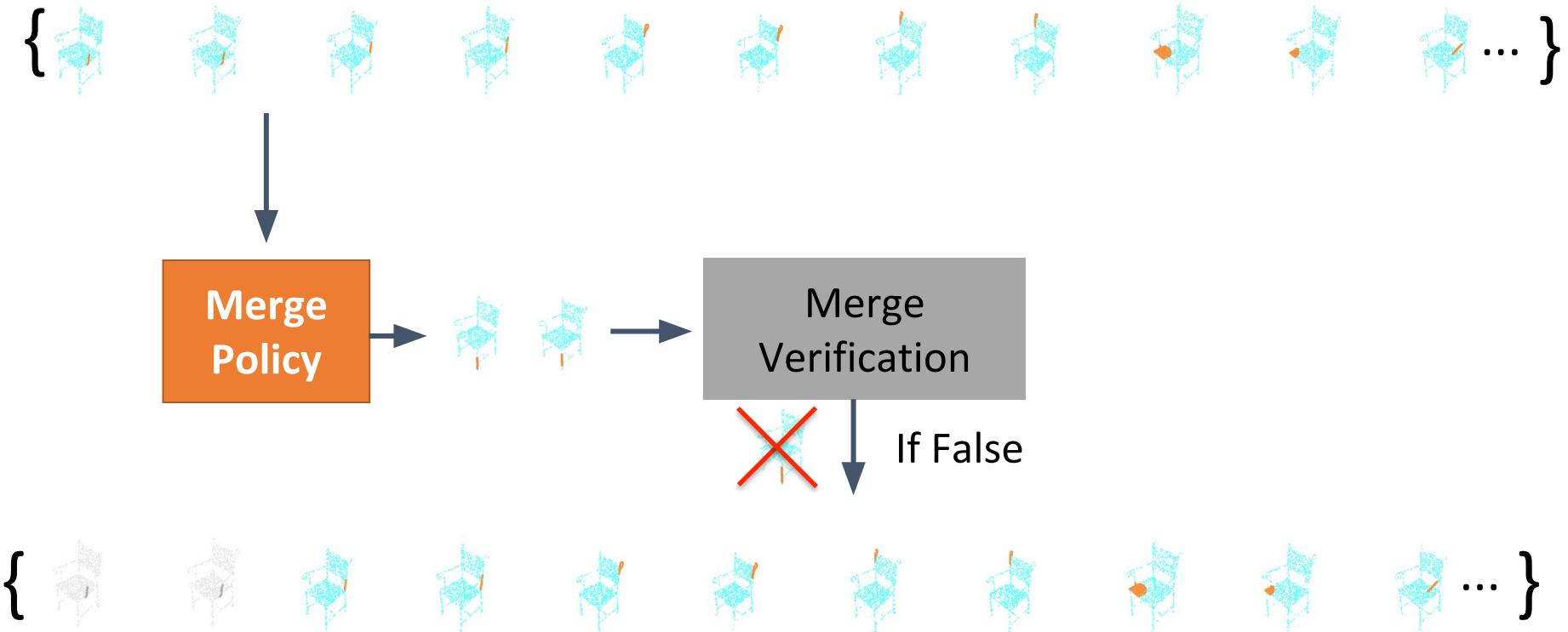
Luo, Tiange, et al. **"Learning to Group: A Bottom-Up Framework for 3D Part Discovery in Unseen Categories."**, ICLR 2020.

Learning to Group: Sub-Part Pool



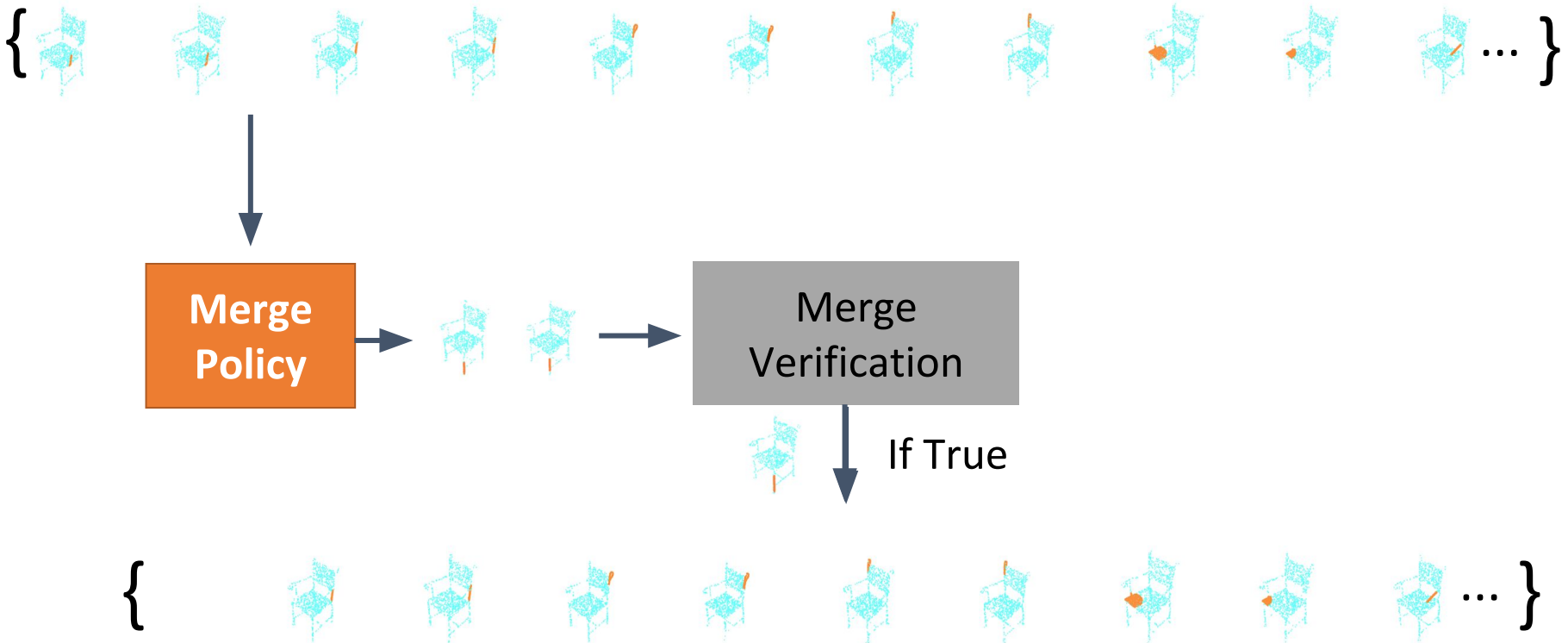
Luo, Tiange, et al. **"Learning to Group: A Bottom-Up Framework for 3D Part Discovery in Unseen Categories."**, ICLR 2020.

Learning to Group: Sub-Part Pool



Luo, Tiange, et al. **"Learning to Group: A Bottom-Up Framework for 3D Part Discovery in Unseen Categories."**, ICLR 2020.

Learning to Group: Sub-Part Pool



Luo, Tiange, et al. **"Learning to Group: A Bottom-Up Framework for 3D Part Discovery in Unseen Categories."**, ICLR 2020.

Learning to Group: Sub-Part Pool

Final

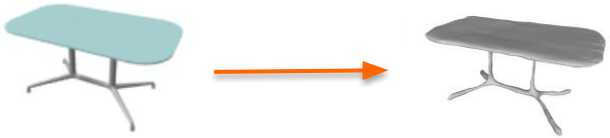
Luo, Tiange, et al. **"Learning to Group: A Bottom-Up Framework for 3D Part Discovery in Unseen Categories."**, ICLR 2020.

Outline

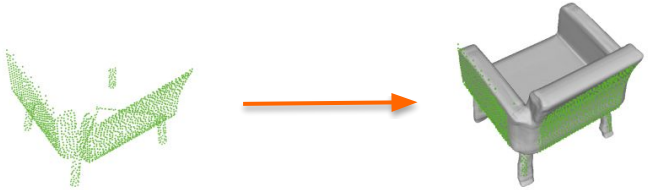
- 3D Data
- Classification
- Segmentation and Detection
- **3D Data synthesis**

Task: 3D Data Synthesis

- Conditional generation

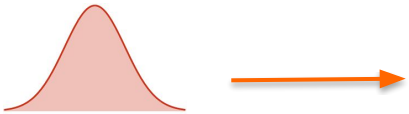


Single-image
3D reconstruction

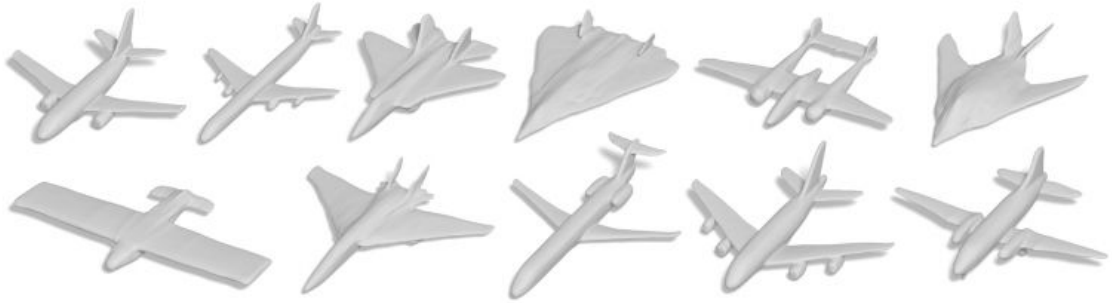


Shape Completion

- Free generation



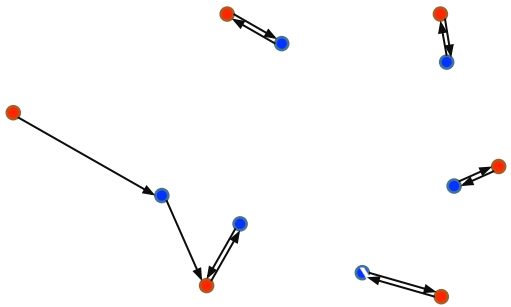
Gaussian Noise



Task: 3D Data Synthesis

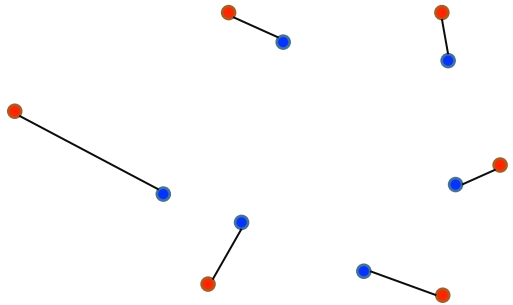
- Metrics to Compare Point Clouds

Chamfer Distance



$$d_{CD}(S_1, S_2) = \sum_{x \in S_1} \min_{y \in S_2} \|x - y\|_2^2 + \sum_{y \in S_2} \min_{x \in S_1} \|x - y\|_2^2$$

Earth Mover's Distance



$$d_{EMD}(S_1, S_2) = \min_{\phi: S_1 \to S_2} \sum_{x \in S_1} \|x - \phi(x)\|_2$$

where $\phi : S_1 \to S_2$ is a bijection.

Fan et al., "**A Point Set Generation Network for 3D Object Reconstruction from a Single Image**", *CVPR 2017*
Wang et al., "**Pixel2mesh: Generating 3d mesh models from single rgb images**", *ECCV 2018*
Mescheder et al., "**Occupancy Networks: Learning 3D Reconstruction in Function Space**", *CVPR 2019*

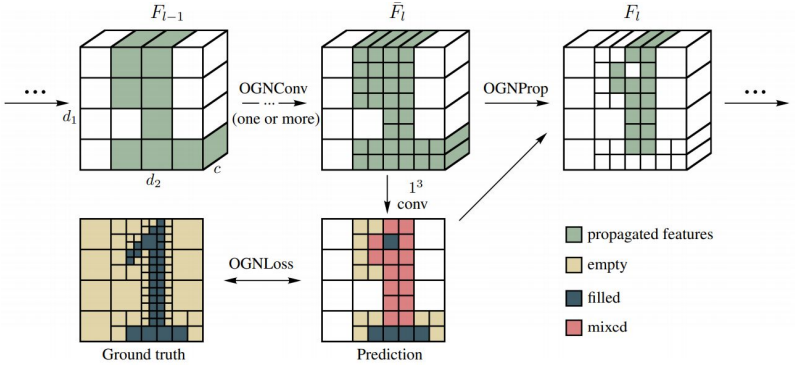
Task: 3D Data Synthesis

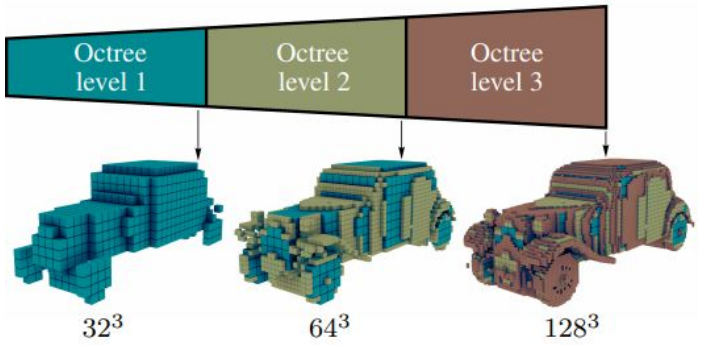
- Metrics to Compare Point Clouds
  - **F-score**: harmonic mean of precision/recall. Percentage of points in one point cloud that can find a neighbor from the other point cloud within a threshold.
  - **Normal Consistency**: dot product of the normals of each point and its nearest neighbor

Fan et al., "**A Point Set Generation Network for 3D Object Reconstruction from a Single Image**", *CVPR 2017*
Wang et al., "**Pixel2mesh: Generating 3d mesh models from single rgb images**", *ECCV 2018*
Mescheder et al., "**Occupancy Networks: Learning 3D Reconstruction in Function Space**", *CVPR 2019*

Task: Conditional Generation

- **From Single Image to Volume:**
  - Octree representation of shapes
  - Avoid $\mathcal{O}(n^3)$ reconstruction
  - Generate the octree layer by layer

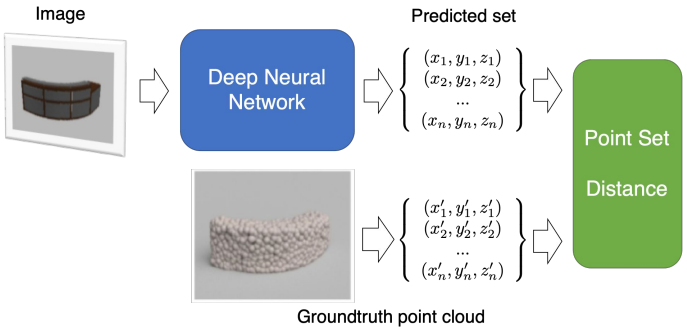


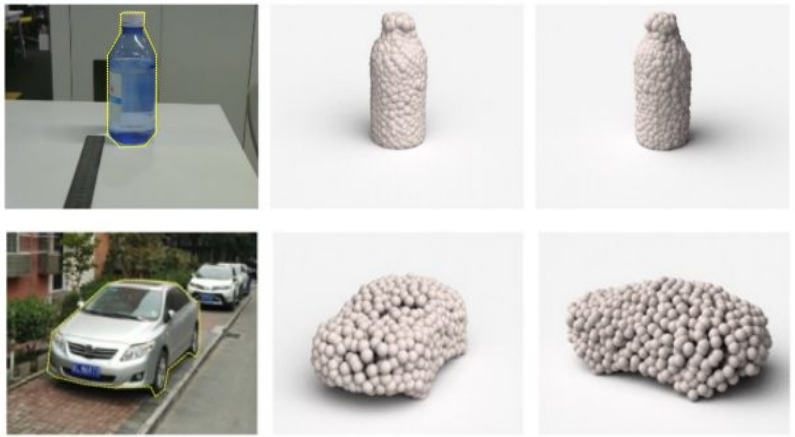


Tatarchenko et al., "**Octree Generating Networks: Efficient Convolutional Architectures for High-resolution 3D Outputs**", *ICCV 2017*

Task: Conditional Generation

- **From Single Image to Point Cloud:**
  - Point set generation

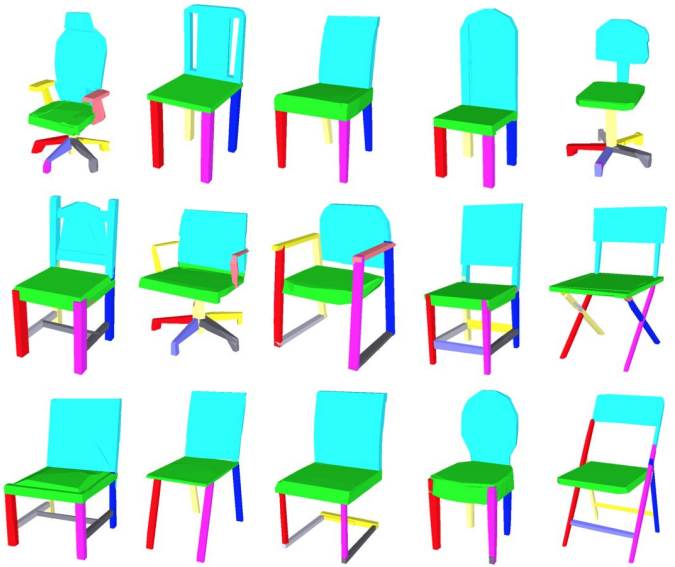




Fan et al., "**A Point Set Generation Network for 3D Object Reconstruction from a Single Image**", *CVPR 2017*

Task: Conditional Generation

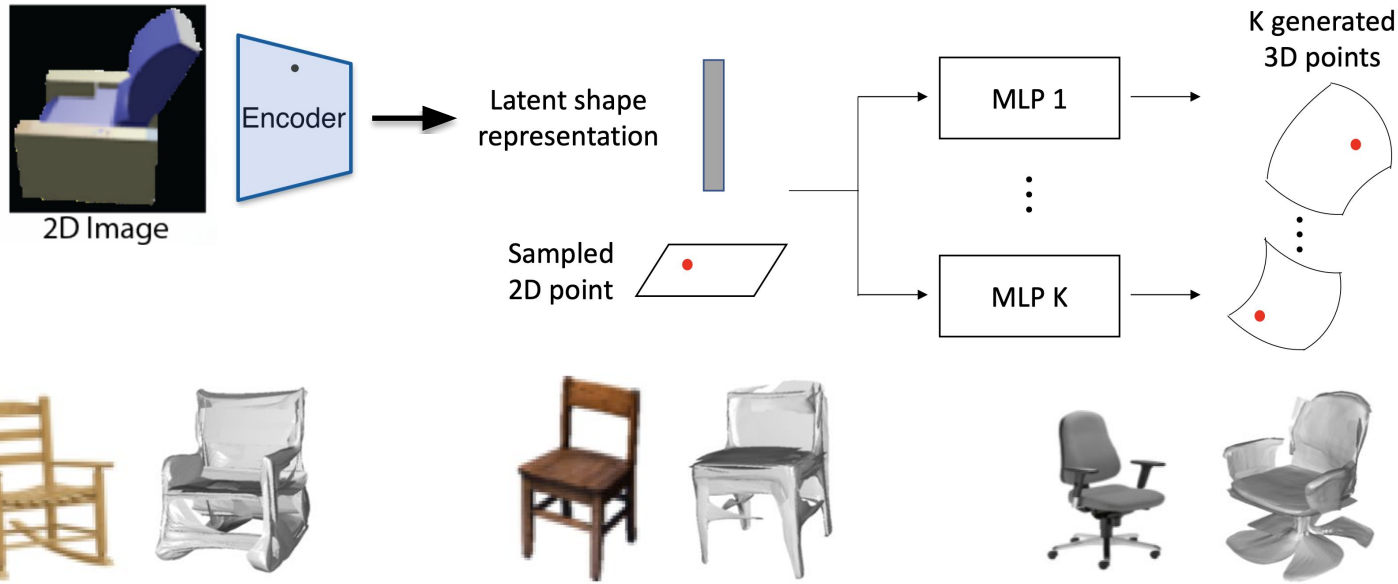
- **From Image to Shape:**
  - Planes -> convex polytopes -> shape



Chen et al., "**BSP-Net: Generating Compact Meshes via Binary Space Partitioning**", *CVPR 2020*

Task: Conditional Generation

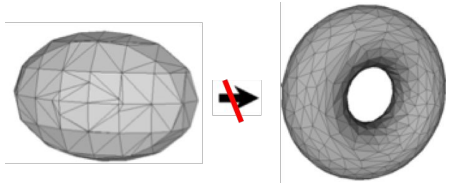
- **From Image to Surface:**
  - Learn to warp a plane to surface



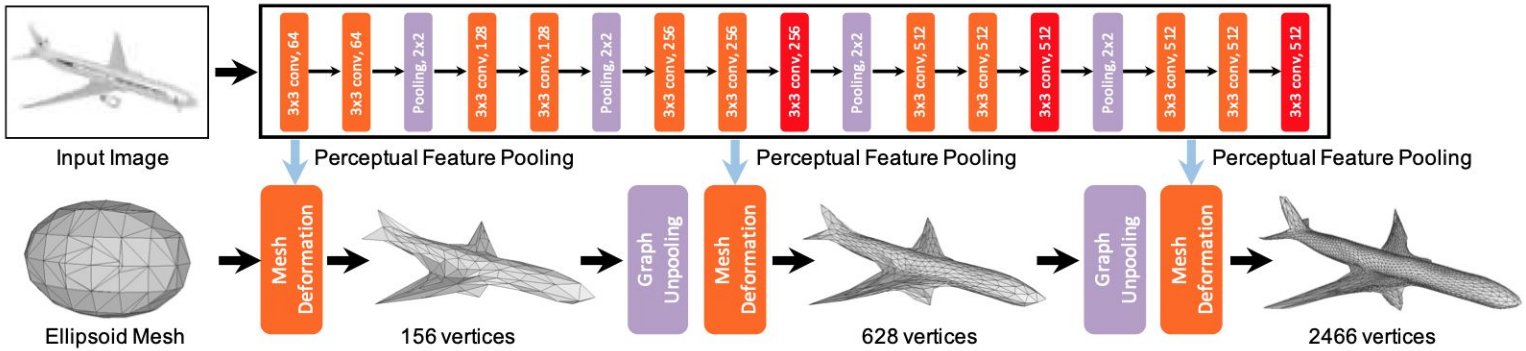
Groueix et al., "**AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation**", *CVPR 2018*
Yang et al., "**Foldingnet: Point cloud auto-encoder via deep grid deformation**", *CVPR 2018*

Task: Conditional Generation

- **From Image to Surface:**
  - Polygon meshes are irregular
  - Learn to deform a template mesh



cannot change topology of the template mesh

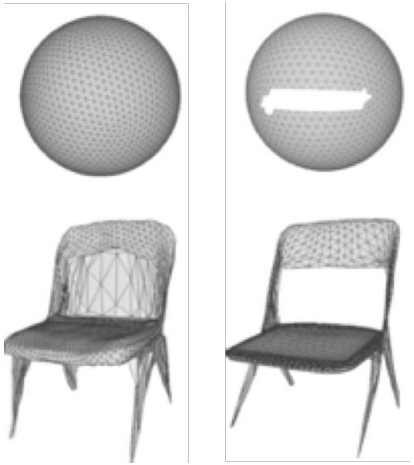


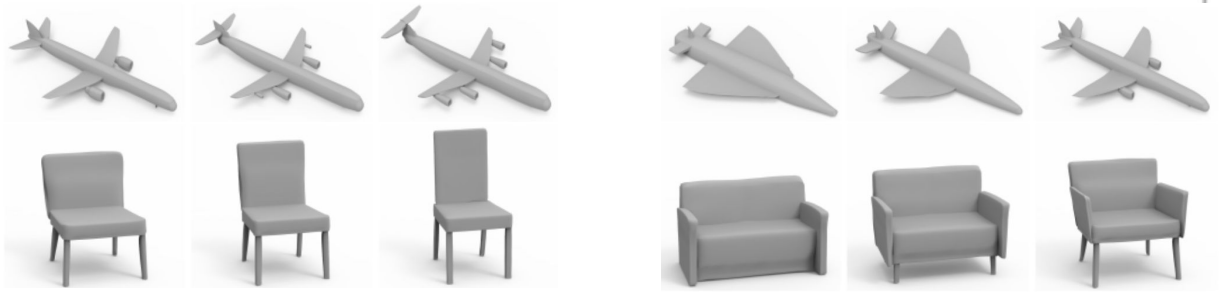
Groueix et al., "**AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation**", *CVPR 2018*
Yang et al., "**Foldingnet: Point cloud auto-encoder via deep grid deformation**", *CVPR 2018*

Task: Conditional Generation

- **From Image to Surface:**
  - modify the topology of the template mesh
  - part-level deformation

Pan et al., "**Deep Mesh Reconstruction from Single RGB Images via Topology Modification Networks.**", *ICCV 2019*

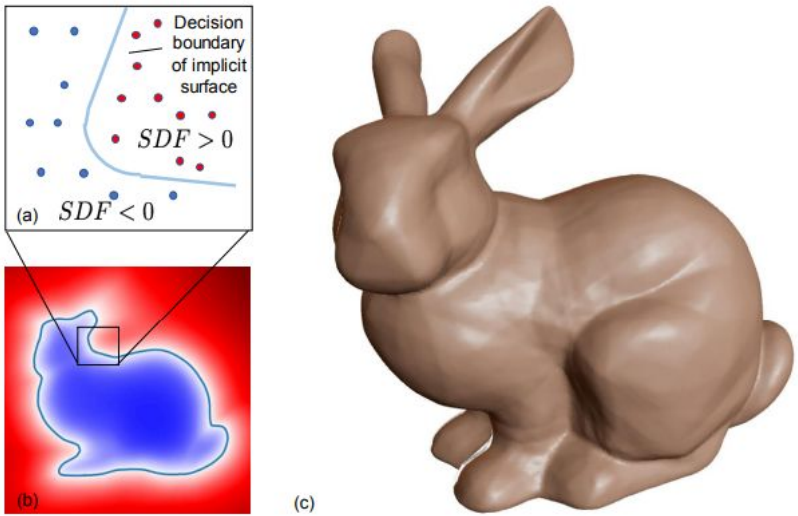


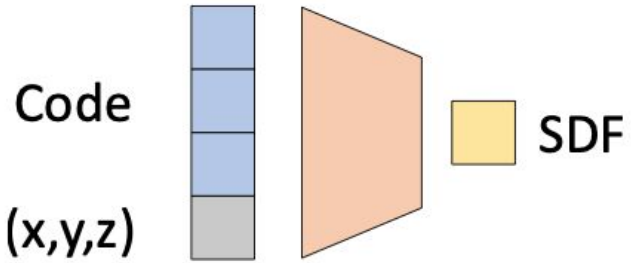


Gao et al., "**SDM-NET: Deep generative network for structured deformable mesh.**", *Siggraph Asia 2019*

Implicit Surface Reconstruction

- Implicit field function $F(x)$ (e.g., signed distance)
- Extract the iso-surface $F(x) = 0$

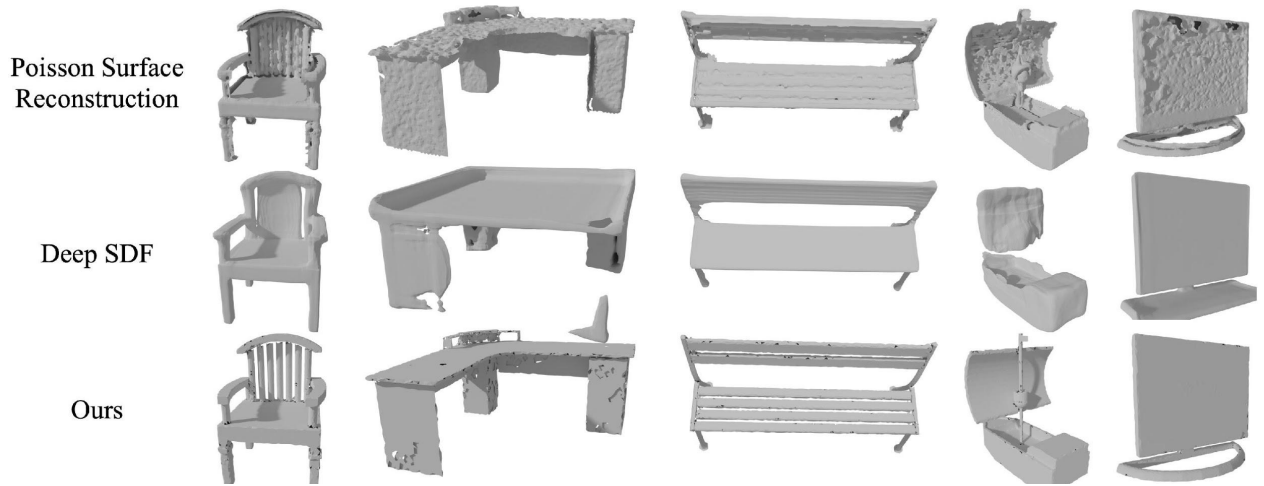




Park et al., "**DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation**", *CVPR 2019*
Mescheder et al., "**Occupancy Networks: Learning 3D Reconstruction in Function Space**", *CVPR 2019*
Chen et al., "**Learning Implicit Fields for Generative Shape Modeling**", *CVPR 2019*

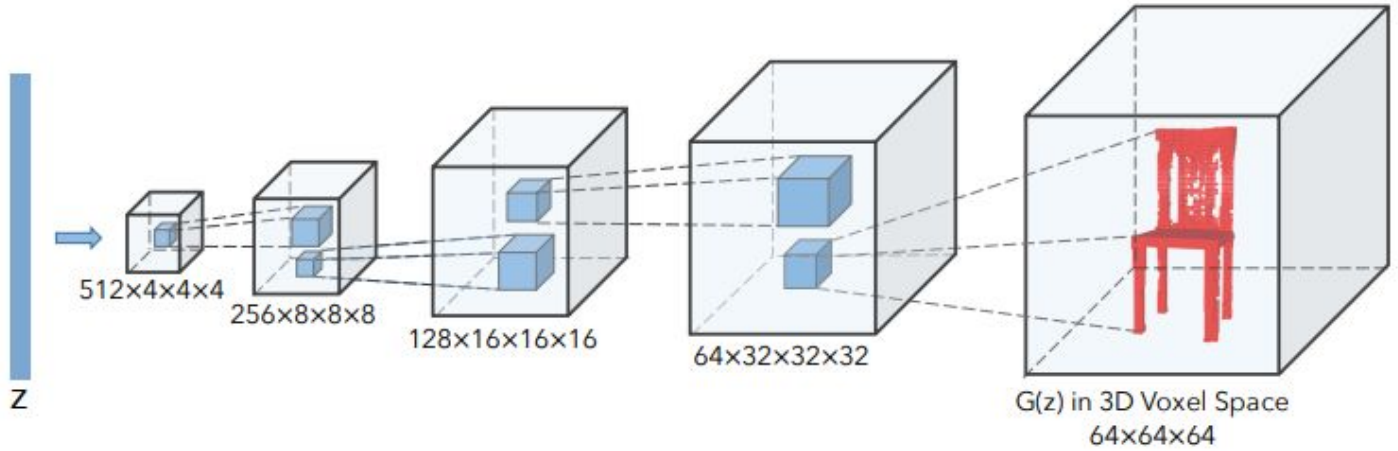
From Point Cloud to Mesh

- **Traditional methods**: cannot handle ambiguous structures (e.g., thin structures) when the resolution of point clouds is limited
- **Learning-based methods**: cannot generate fine-grained details or generalize to unseen objects



Task: Free Generation

- Volumetric Generation



$$L_{\text{3D-GAN}} = \log D(x) + \log(1 - D(G(z)))$$

Wu et al., "**Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling**", *NeurIPS 2016*

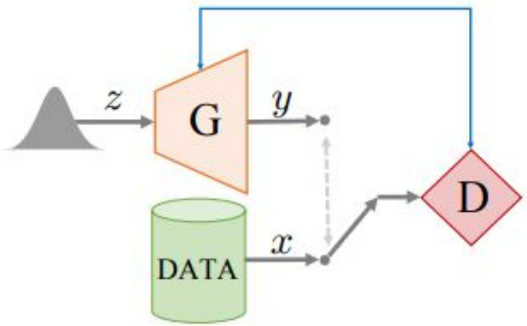
Task: Free Generation

- Volumetric Generation

Wu et al., “**Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling**”, *NeurIPS 2016*

Task: Free Generation

- Point Cloud Generation:
  - FC layer as generator
  - PointNet as discriminator
  - WGAN

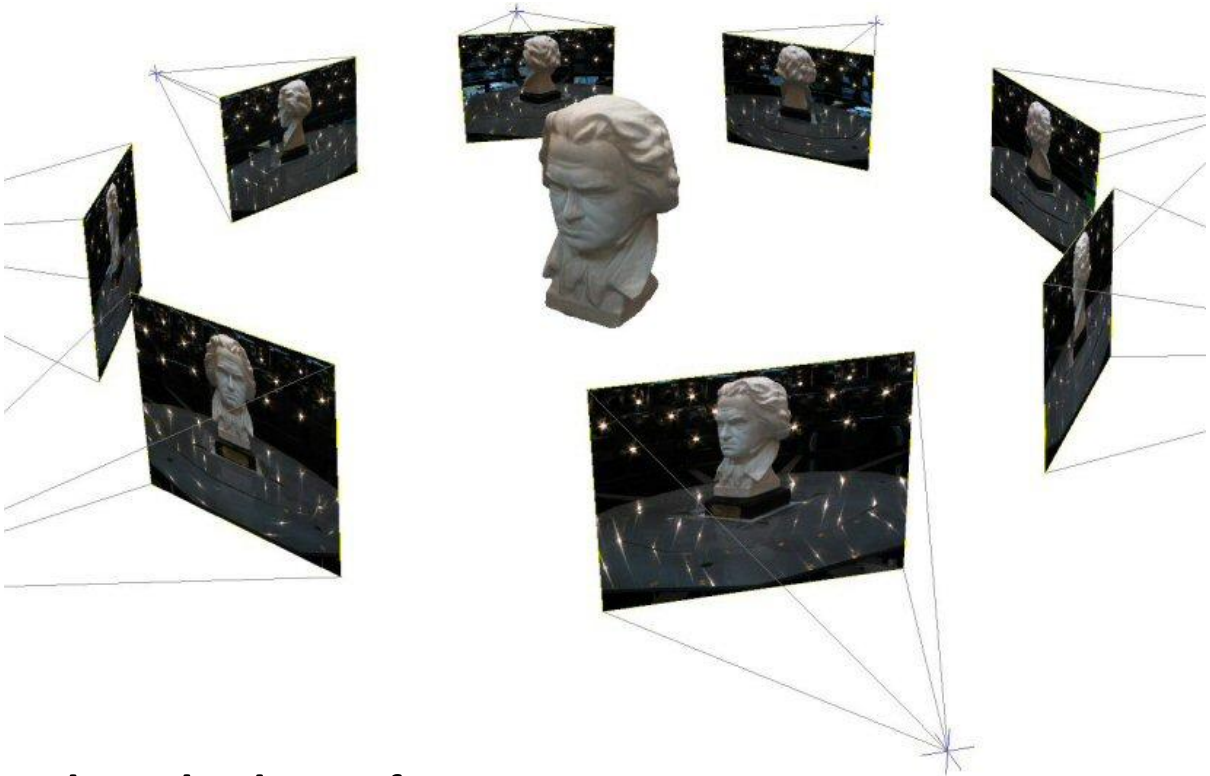


Achlioptas et al., "**Learning Representations and Generative Models for 3D Point Clouds**", *ICML 2018*

Task: Free Generation: Current challenges

- Still cannot generate high quality local details
- Still hard to generate complex structures
- For point clouds, using strong classifiers (than PointNet) as discriminator is highly tricky

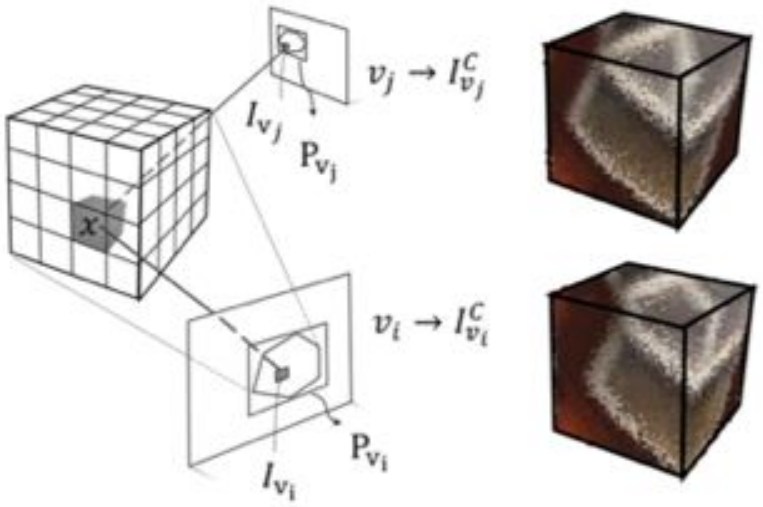
Task: Multi View Stereo Reconstruction



Covered methods: SurfaceNet, LSM, GC-Net, MVSNet, R-MVSNet, PointMVSNet, BA-Net

Task: Multi View Stereo Reconstruction

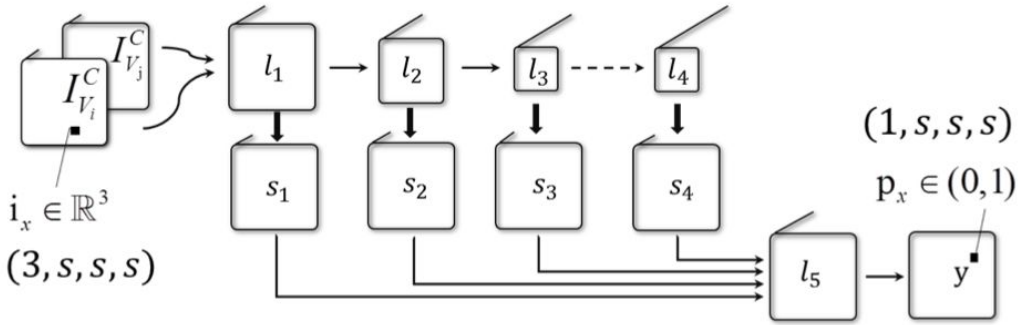
- Surface Reconstruction as Voxel Occupancy Prediction
  - project along viewing rays to build colored voxel cubes.



Ji et al., "**SurfaceNet: An End-to-End 3D Neural Network for Multiview Stereopsis**", *ICCV 2017*

Task: Multi View Stereo Reconstruction

- Surface Reconstruction as Voxel Occupancy Prediction
  - Predict the surface confidence for each voxel



$$L(I^C_{v_i}, I^C_{v_j}, \hat{S}^C) =$$

$$-\sum_{x \in C} \{\alpha \hat{s}_x \log p_x + (1-\alpha)(1-\hat{s}_x)\log(1-p_x)\}$$

Ji et al., "**SurfaceNet: An End-to-End 3D Neural Network for Multiview Stereopsis**", *ICCV 2017*

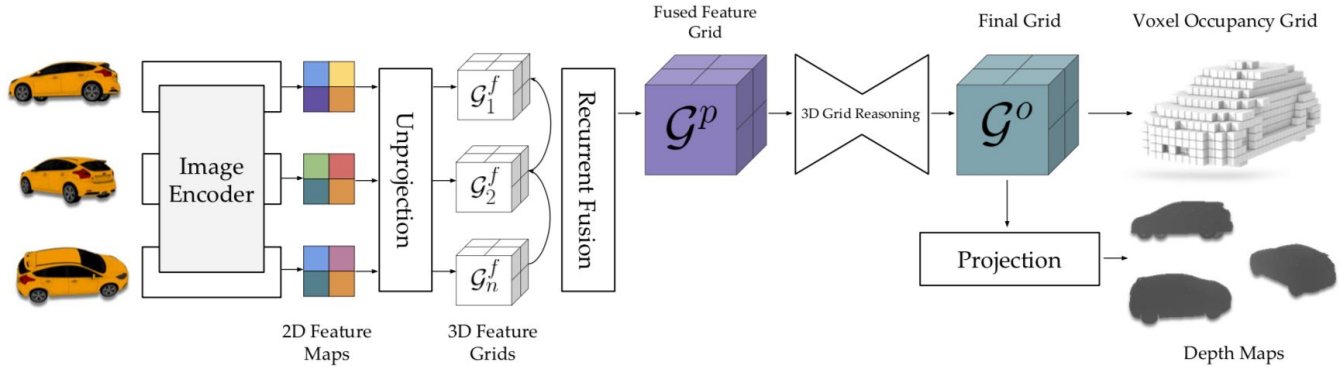
Task: Multi View Stereo Reconstruction

- Surface Reconstruction as Voxel Occupancy Prediction
  - Limitations:
    - Pre-computed voxel cubes can only take RGB colors at coarse resolution
    - Voxel binarization introduces quantization errors.

Ji et al., "**SurfaceNet: An End-to-End 3D Neural Network for Multiview Stereopsis**", *ICCV 2017*

Task: Multi View Stereo Reconstruction

- Learning-Based Stereopsis
  - End-to-end learning of deep features for each image pixel
  - Unproject image features into 3D grids



Kar et al., "**Learning a Multi-View Stereo Machine**", *NeurIPS 2017*

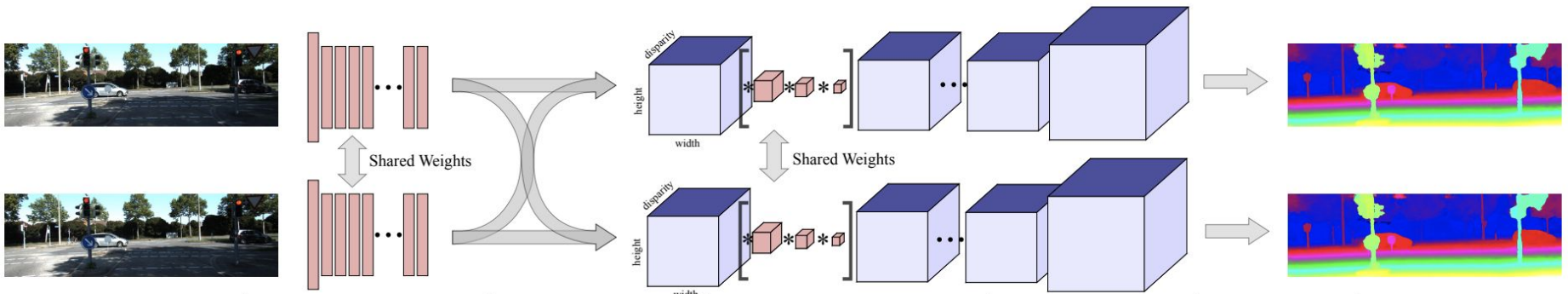
Task: Multi View Stereo Reconstruction

- Learning-Based Stereopsis
  - Still very coarse resolution (32x32x32) due to volumetric representation.

Kar et al., "**Learning a Multi-View Stereo Machine**", *NeurIPS 2017*

Task: Multi View Stereo Reconstruction

- View-aligned cost-volume construction
- Differentiable soft-argmin to achieve sub-pixel accuracy

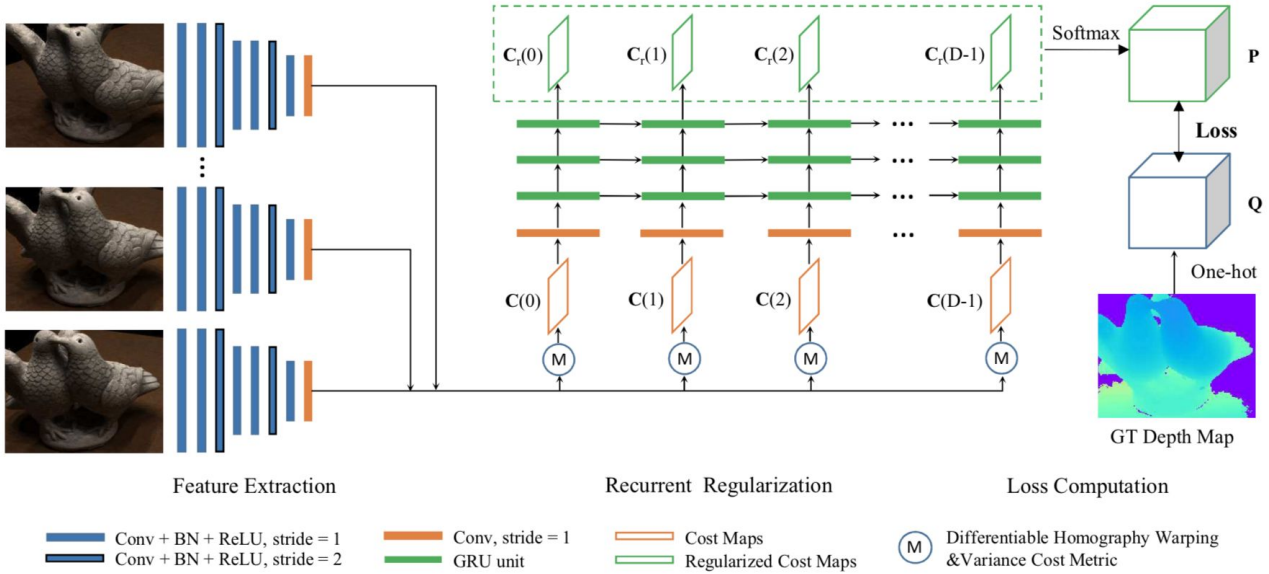


$$soft\ argmin := \sum_{d=0}^{D_{max}} d \times \sigma(-c_d)$$

Kendall et al., "**End-to-End Learning of Geometry and Context for Deep Stereo Regression**", *ICCV 2017*

Task: Multi View Stereo Reconstruction

- **Idea 1**: Slide-by-Slide Processing of Cost Volume by Recurrent Neural Network

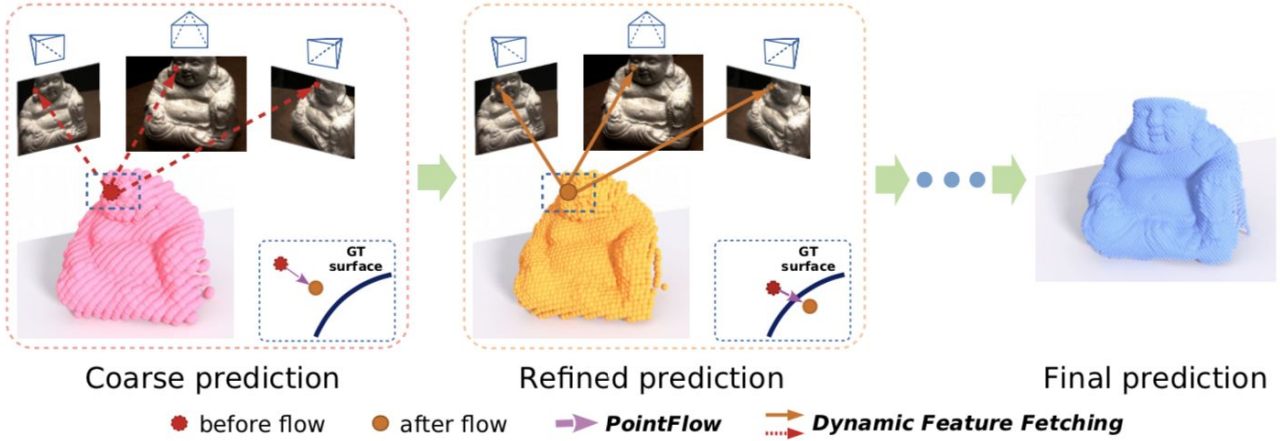


The cost volume is sequentially regularized along the depth direction.

Yao et al., "**MVSNet: Depth Inference for Unstructured Multi-view Stereo**", *ECCV 2018*
Yao et al., "**Recurrent MVSNet for High-resolution Multi-view Stereo Depth Inference**", *CVPR 2019*

Task: Multi View Stereo Reconstruction

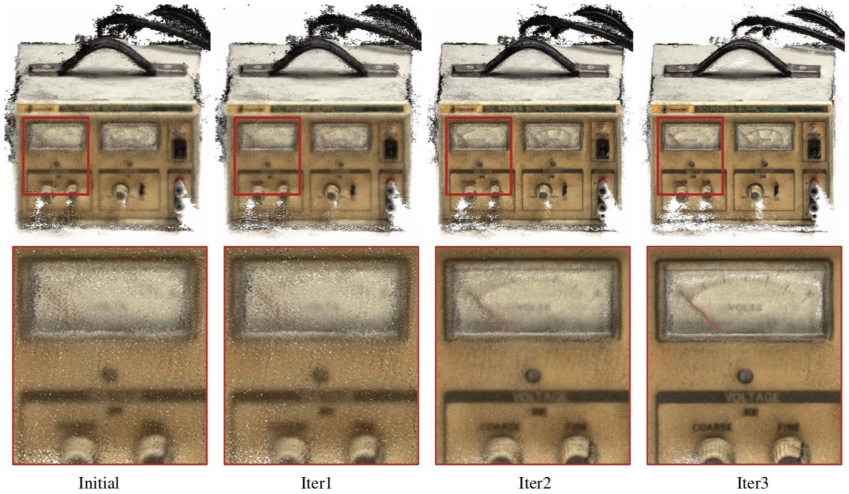
- **Idea 2**: Point-based MVS
  - Point-based representation for computational efficiency.
  - Iteratively update the location of points and spawn more points.
  - More flexible and accurate.



Chen et al., "**Point-Based Multi-View Stereo Network**", *ICCV 2019*

Task: Multi View Stereo Reconstruction

- **Idea 2**: Point-based MVS
  - Iterative refinement
  - Results on DTU benchmark



Iter.	Acc. (mm)	Comp. (mm)	Overall (mm)	0.5mm *f-score*	Depth Map Res.	Depth Interval (mm)	GPU Mem. (MB)	Runtime (s)
-	0.693	0.758	0.726	47.95	160×120	5.30	**7219**	**0.34**
1	0.674	0.750	0.712	48.63	160×120	5.30	7221	0.61
2	0.448	0.487	0.468	76.08	320×240	4.00	7235	1.14
3	**0.361**	**0.421**	**0.391**	**84.27**	**640×480**	**0.80**	8731	3.35
MVSNet[29]	0.456	0.646	0.551	71.60	288×216	2.65	10805	1.05

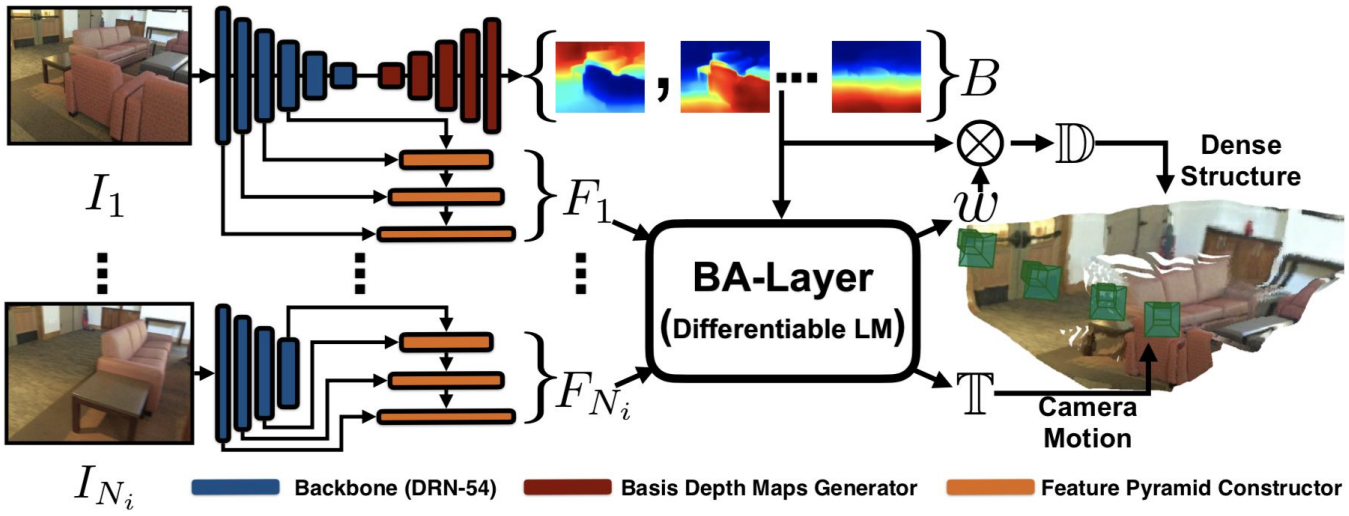
Chen et al., "**Point-Based Multi-View Stereo Network**", *ICCV 2019*

Task: Learning for SfM

- Above learning-based MVS methods all **assume camera poses are available**
- What if not?
  - Classic 3D: Bundle Adjustment
- **Learning-based bundle adjustment**

BA-Net: Dense Bundle Adjustment Network

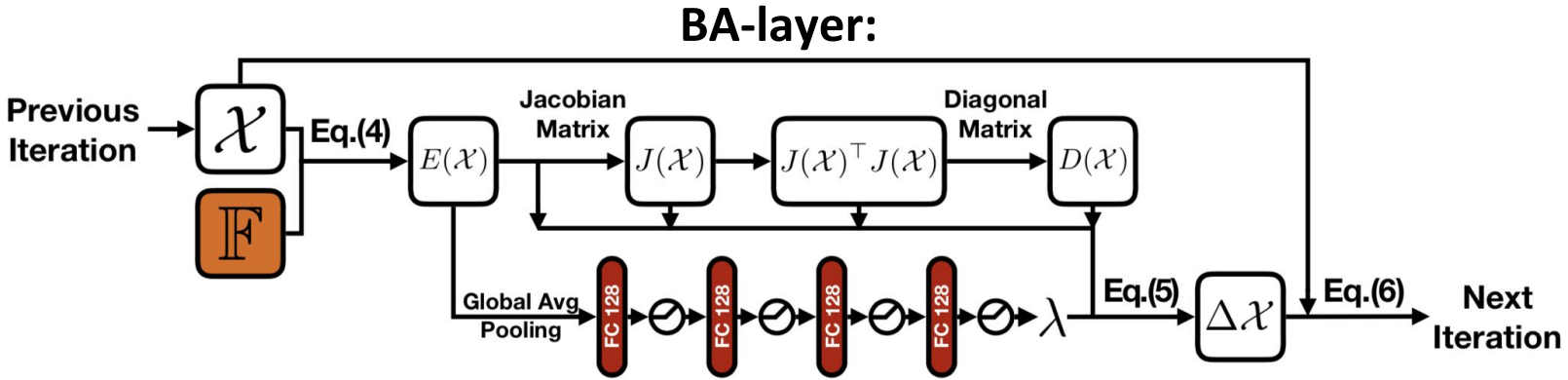
- End-to-end pipeline for SfM with differentiable bundle adjustment.



Tang et al., "**BA-Net: Dense Bundle Adjustment Network**", *ICLR 2019*

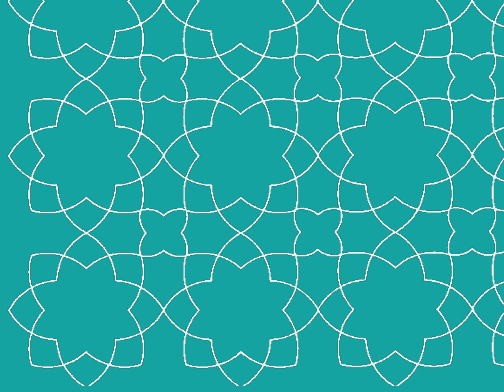
BA-Net: Dense Bundle Adjustment Network

- Differentiable LM algorithm:
  - Iterative update as rollout of network layers
  - Use network to predict the damping factor lambda.



$$\Delta\mathcal{X} = (J(\mathcal{X})^\top J(\mathcal{X}) + \lambda D(\mathcal{X}))^{-1} J(\mathcal{X})^\top E(\mathcal{X}).$$

Tang et al., "**BA-Net: Dense Bundle Adjustment Network**", *ICLR 2019*



Prof. Bernard Ghanem

3D DEEP LEARNING