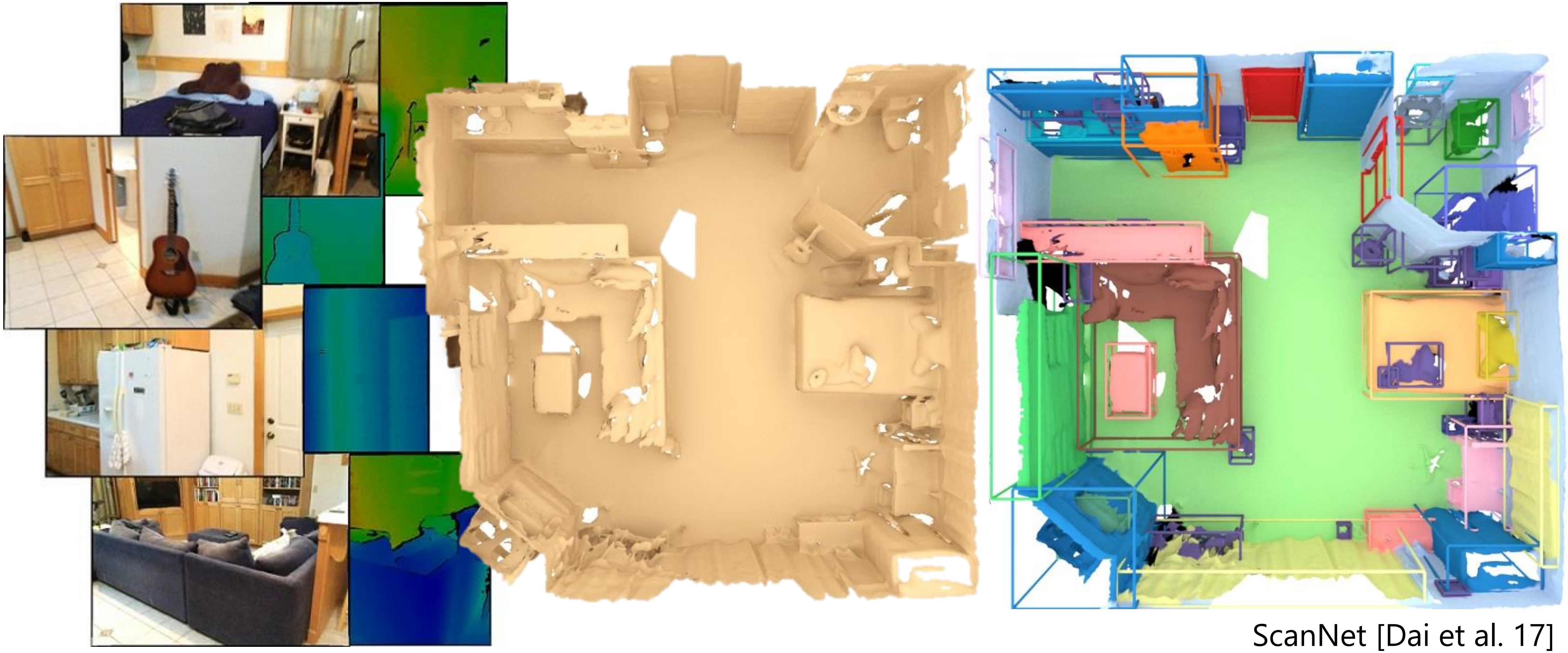
Shape Segmentation and Labeling

Prof. Angela Dai

Brief Recap

Machine Perception of Real-World Environments



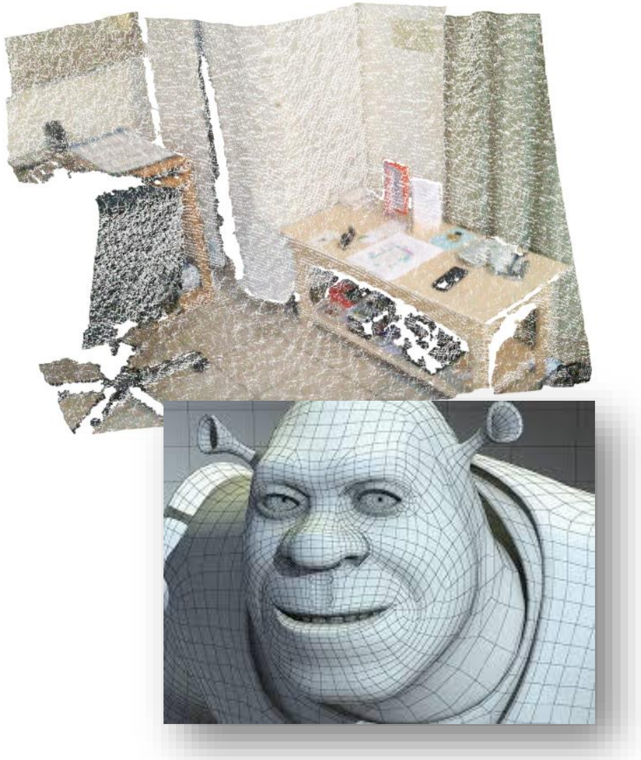
ScanNet [Dai et al. 17]

We *perceive* and *interact* with a 3D world

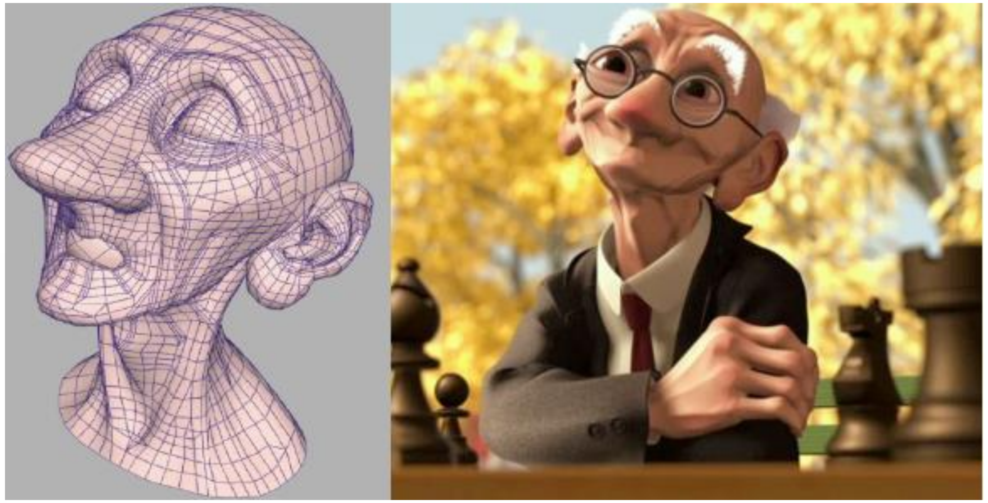
ASIMO, Honda


Star Trek TNG (Phantasms)

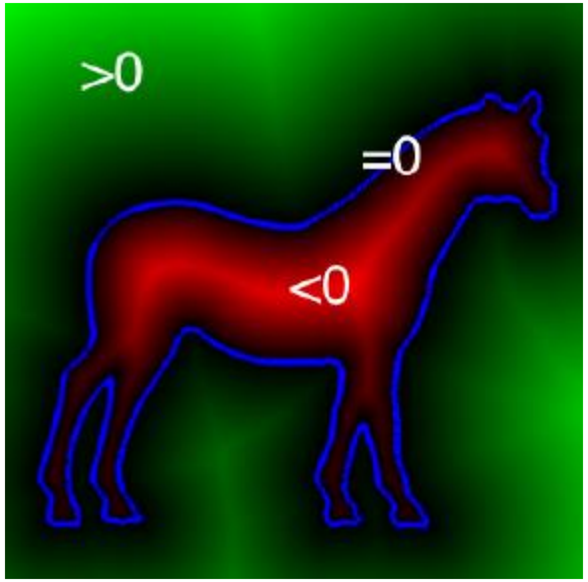
How to represent 3D?



Discrete:
Meshes,
Point Samples



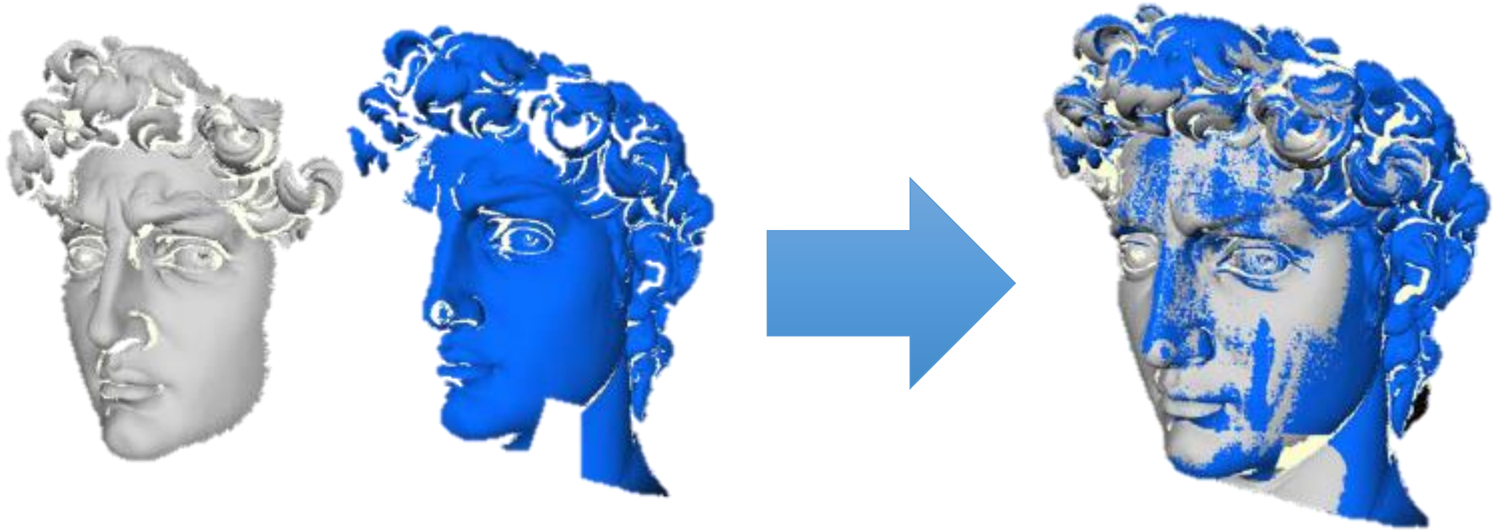
Parametric



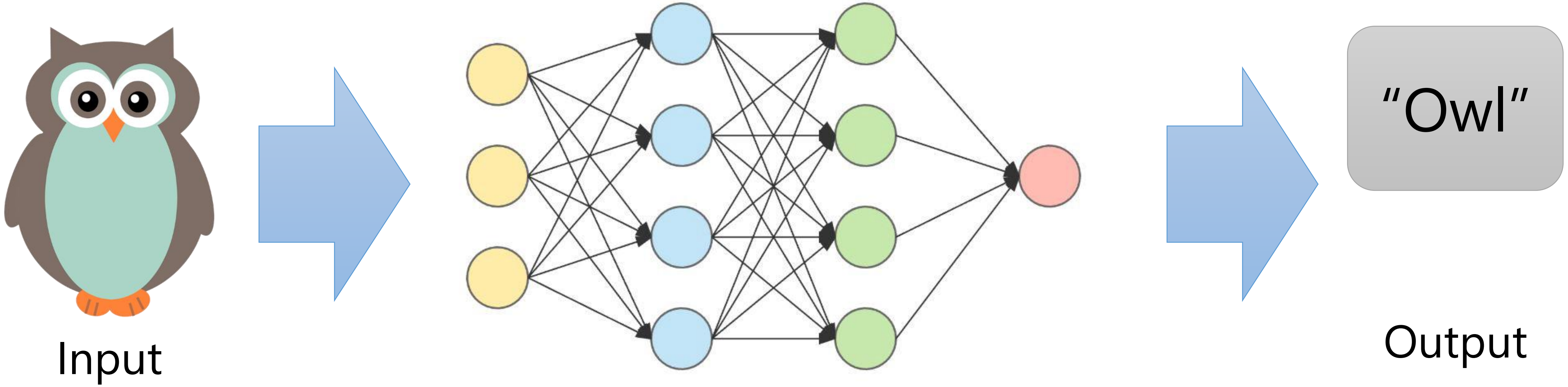
Implicit:
Distance Fields

Classical Geometry Processing

- Meshing
- Rigid Alignment
- Non-Rigid Alignment



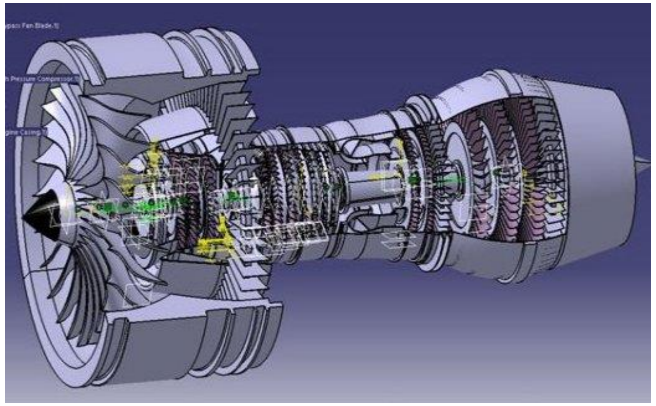
Deep Learning



Learned Feature Extraction and Classification

Want to automatically learn good feature representations for the task

Semantic understanding of 3D shapes

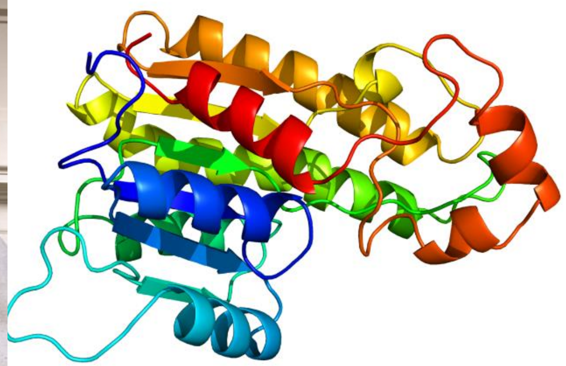


Mechanical CAD Models

Artist-Modeled Shapes

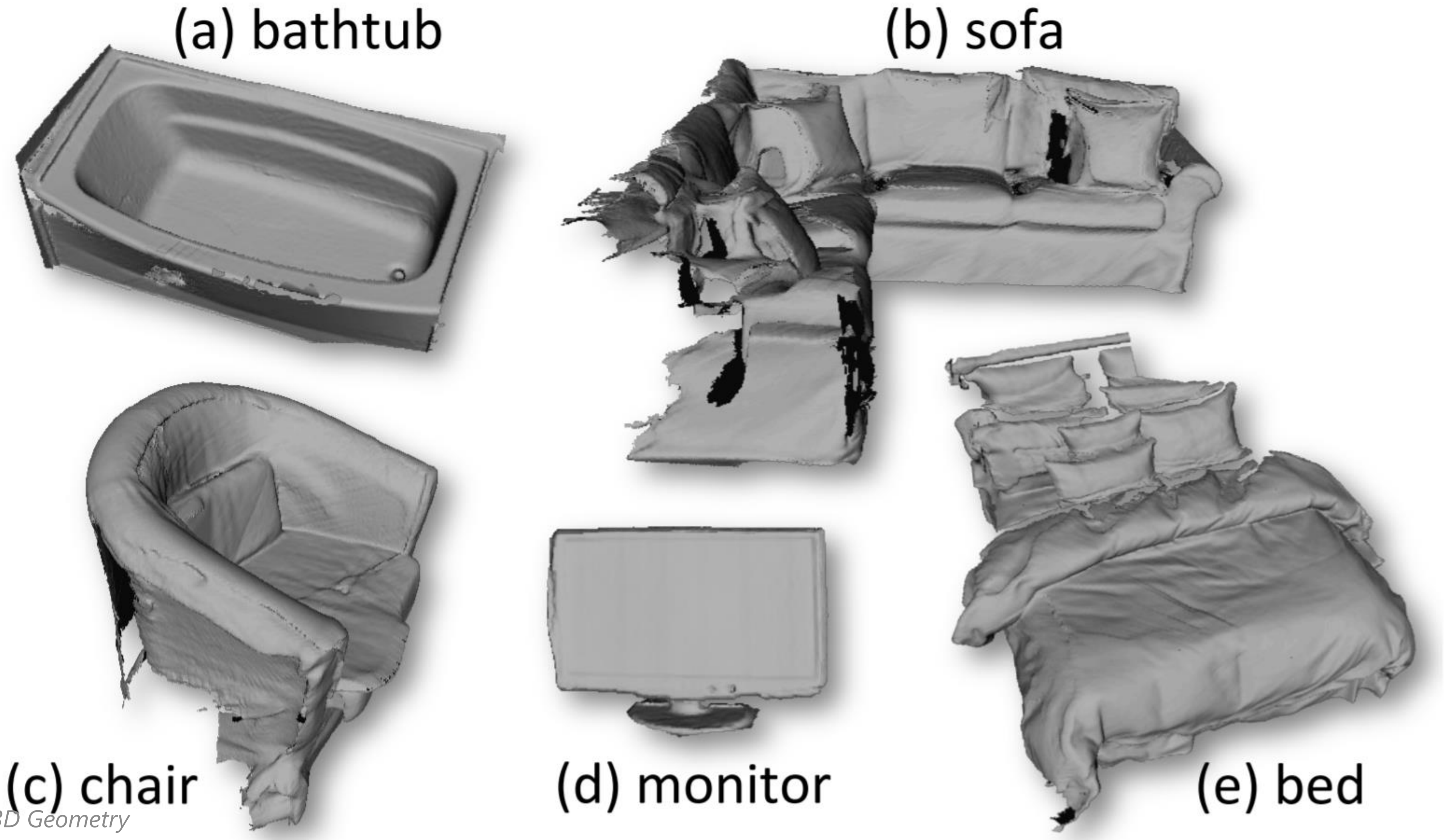
Interior Modeling

Cultural Heritage



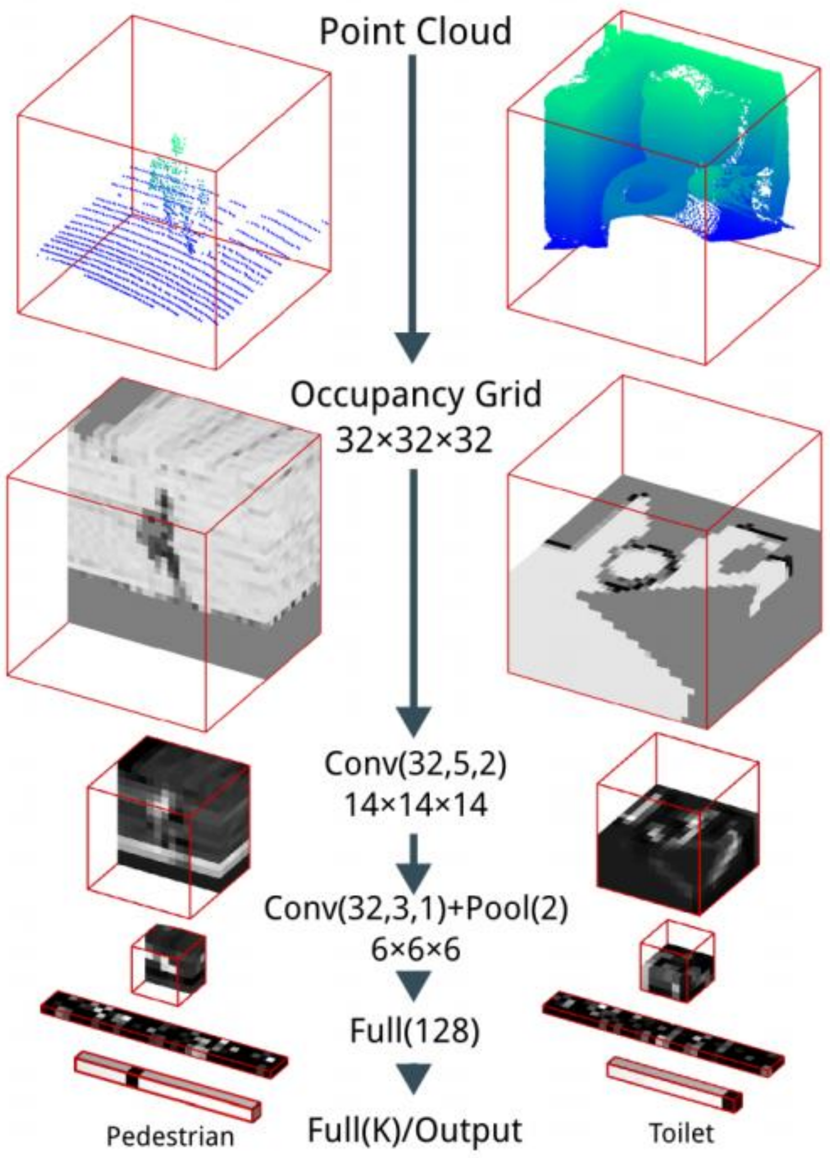
3D Medical Imaging

Shape Classification



Interpret Shapes as Voxels

- VoxNet: One of the first 3D CNNs
- Object classification
- Volumetric occupancy grid input
- Rotation augmentation to classify arbitrary poses:
  - During training
  - Voting at test time

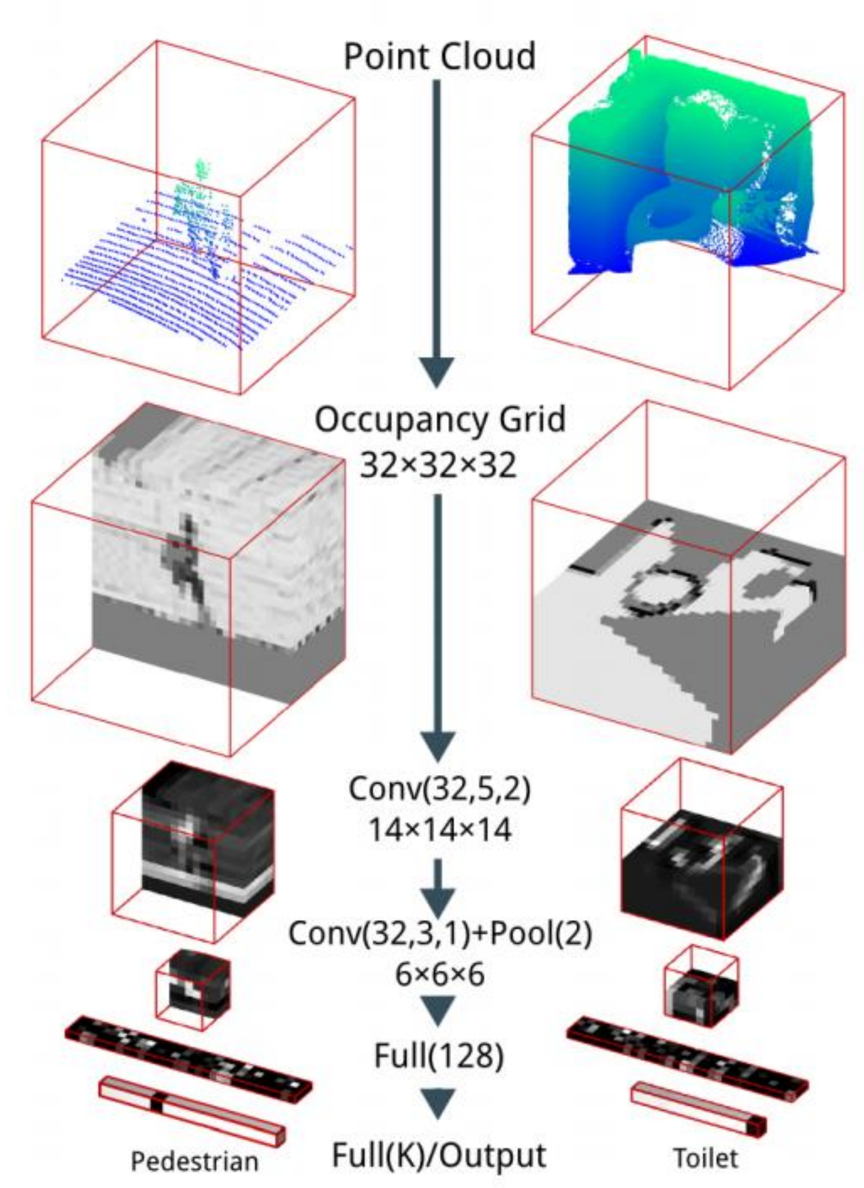


[Maturana and Scherer ’15]

VoxNet

- Small network
  - 1 million parameters
  - fast inference (~6ms)

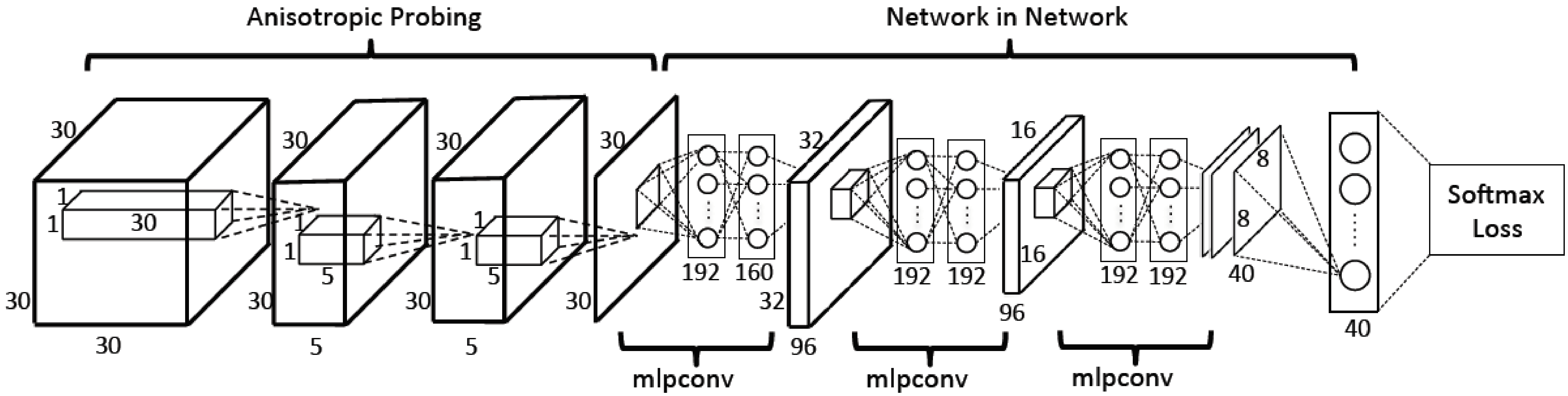
- Good classification performance on synthetic shape benchmarks



[Maturana and Scherer ’15]

Additional Voxel Grid Operators

- 3DCNN: Object classification with anisotropic kernels and network-in-network
- Volumetric occupancy grid

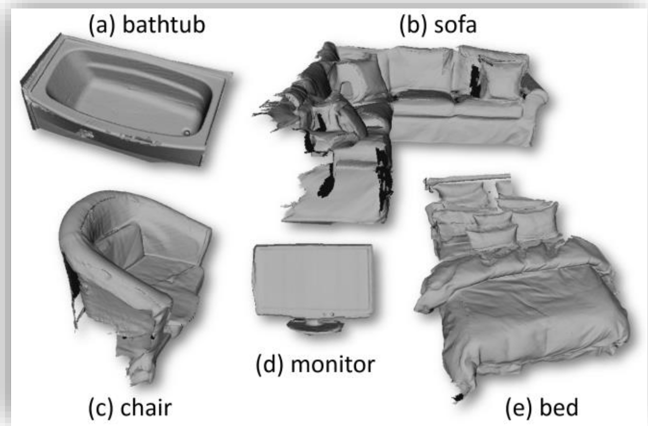


[Qi et al. ’16]

3DCNN

- Object classification with anisotropic kernels and network-in-network
- Volumetric occupancy grid
- Real-world object classification – without real data, performance drops!

Synthetic objects (40 classes): 89.9% instance accuracy

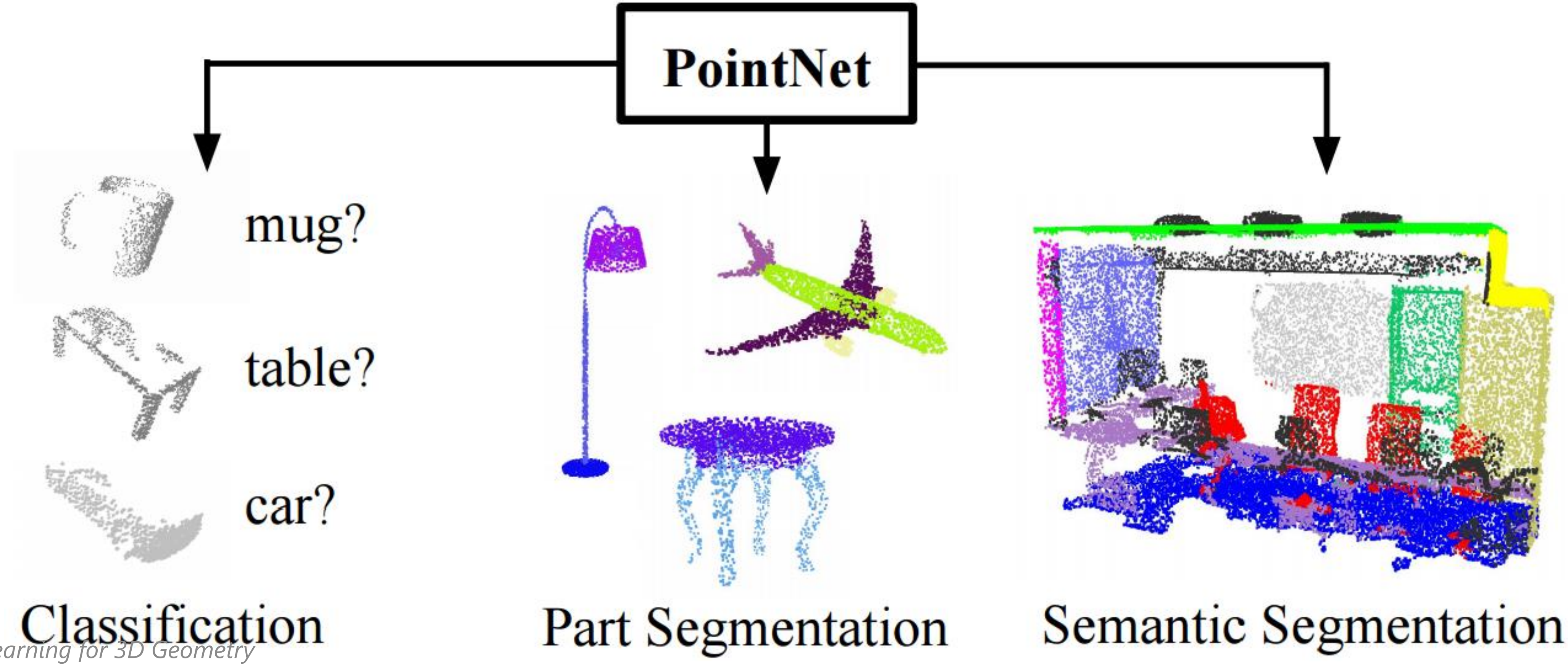


Real objects: 74.5% instance accuracy

[Qi et al. ’16]

Interpret Shapes as Points: PointNet

- Deep network architecture to operate on unordered point sets (point clouds)

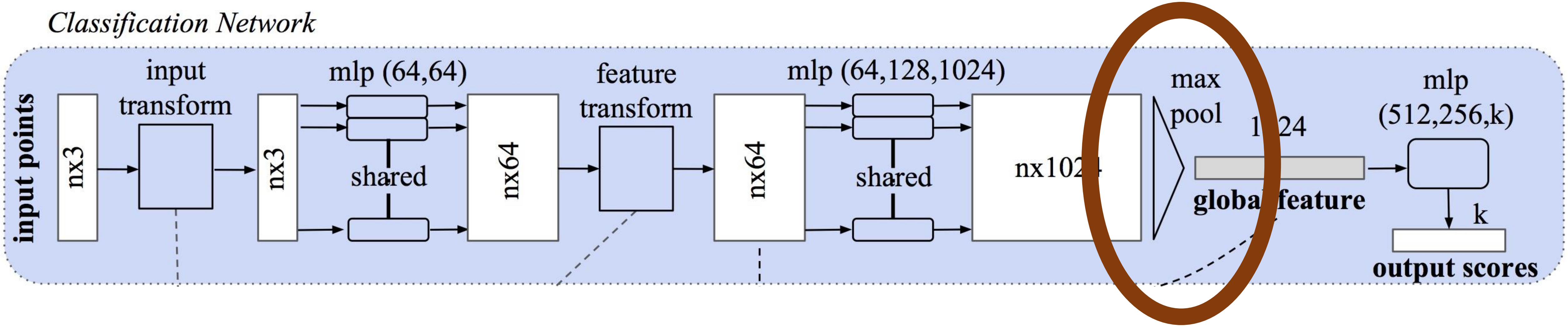


[Qi et al. '17]

PointNet

- Deep network architecture to operate on unordered point sets (point clouds)

Classification Network

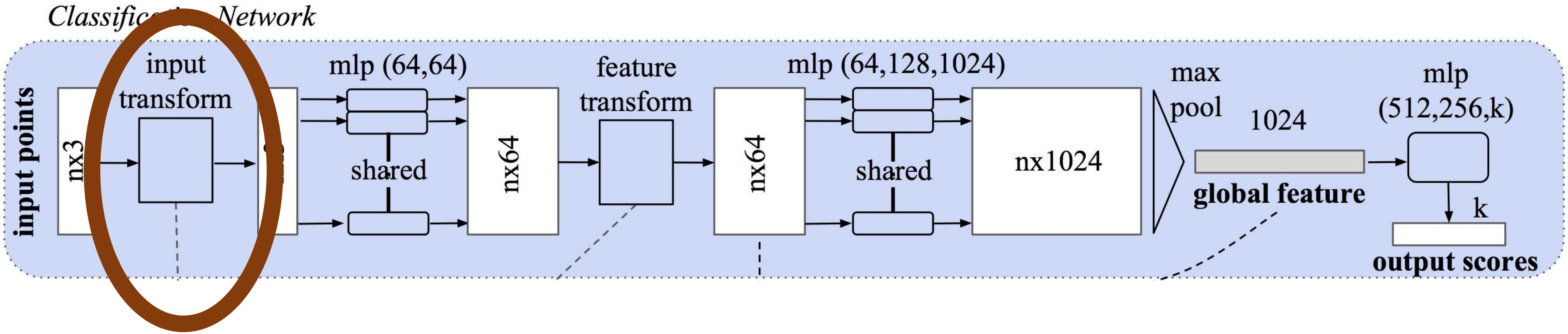


Aggregation by a symmetric function: max pooling

[Qi et al. ’17]

PointNet

- Deep network architecture to operate on unordered point sets (point clouds)



Learn data-dependent transformation into an aligned coordinate system

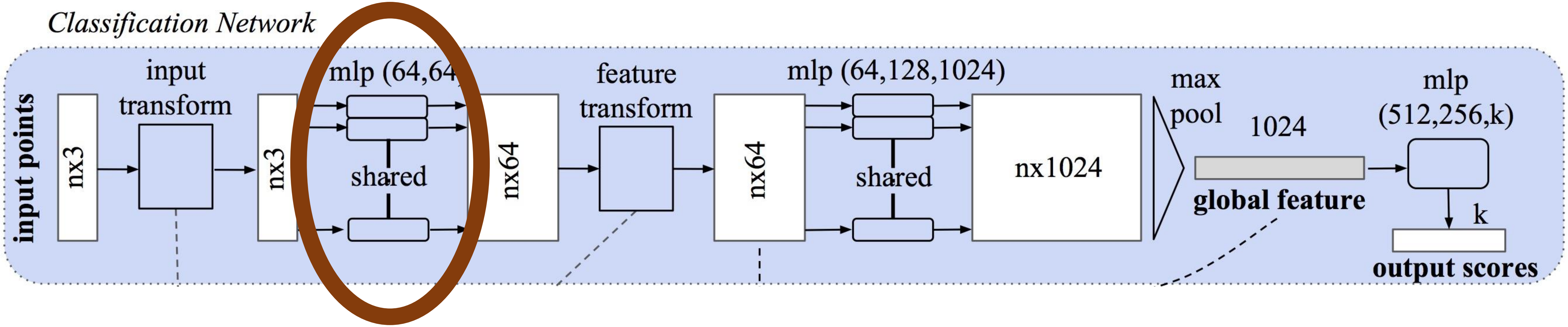
Learned transform: 3×3 matrix; represent affine transform

➢ matrix multiply on point data

[Qi et al. '17]

PointNet

- Deep network architecture to operate on unordered point sets (point clouds)

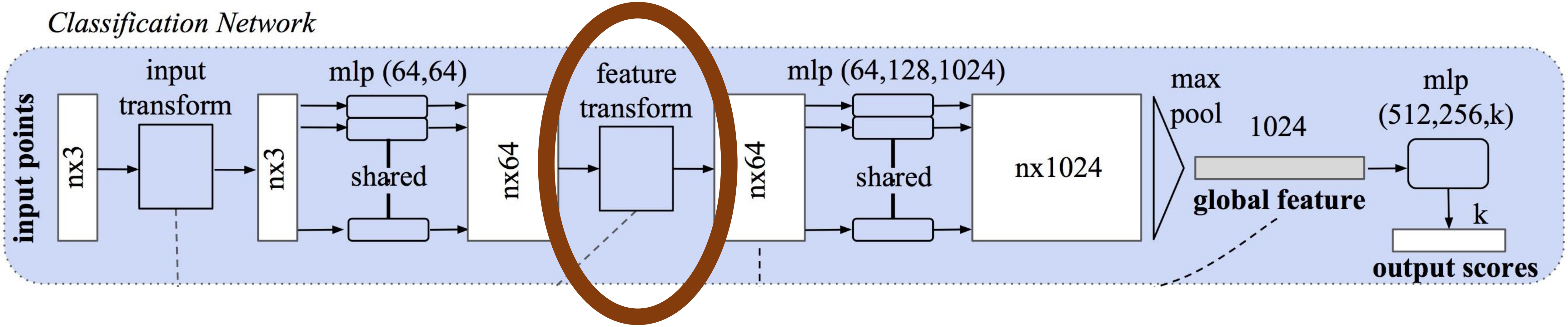


Shared MLP to process aligned points

[Qi et al. '17]

PointNet

- Deep network architecture to operate on unordered point sets (point clouds)

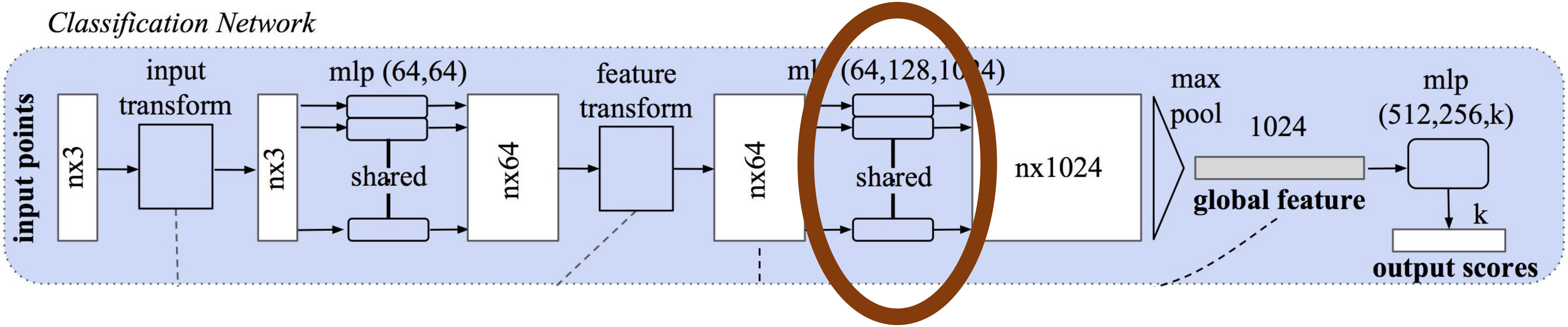


Learn data-dependent feature alignment transformation (64×64)
Regularization to be close to orthogonal

[Qi et al. '17]

PointNet

- Deep network architecture to operate on unordered point sets (point clouds)

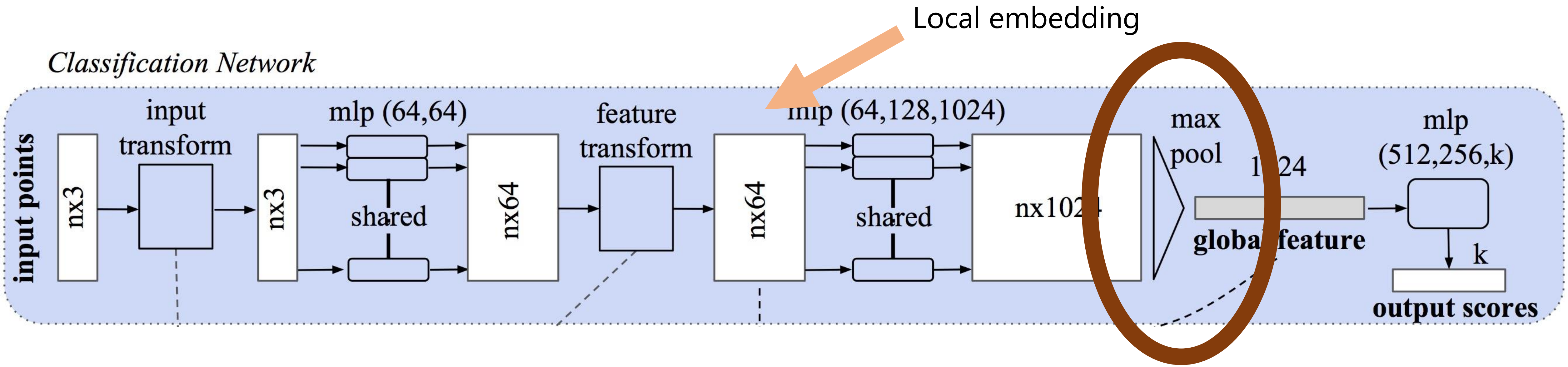


Shared MLP to process aligned features

[Qi et al. ’17]

PointNet

- Deep network architecture to operate on unordered point sets (point clouds)



Aggregation by a symmetric function: max pooling
➢ global feature

[Qi et al. ’17]

PointNet

- Object Classification on ModelNet40

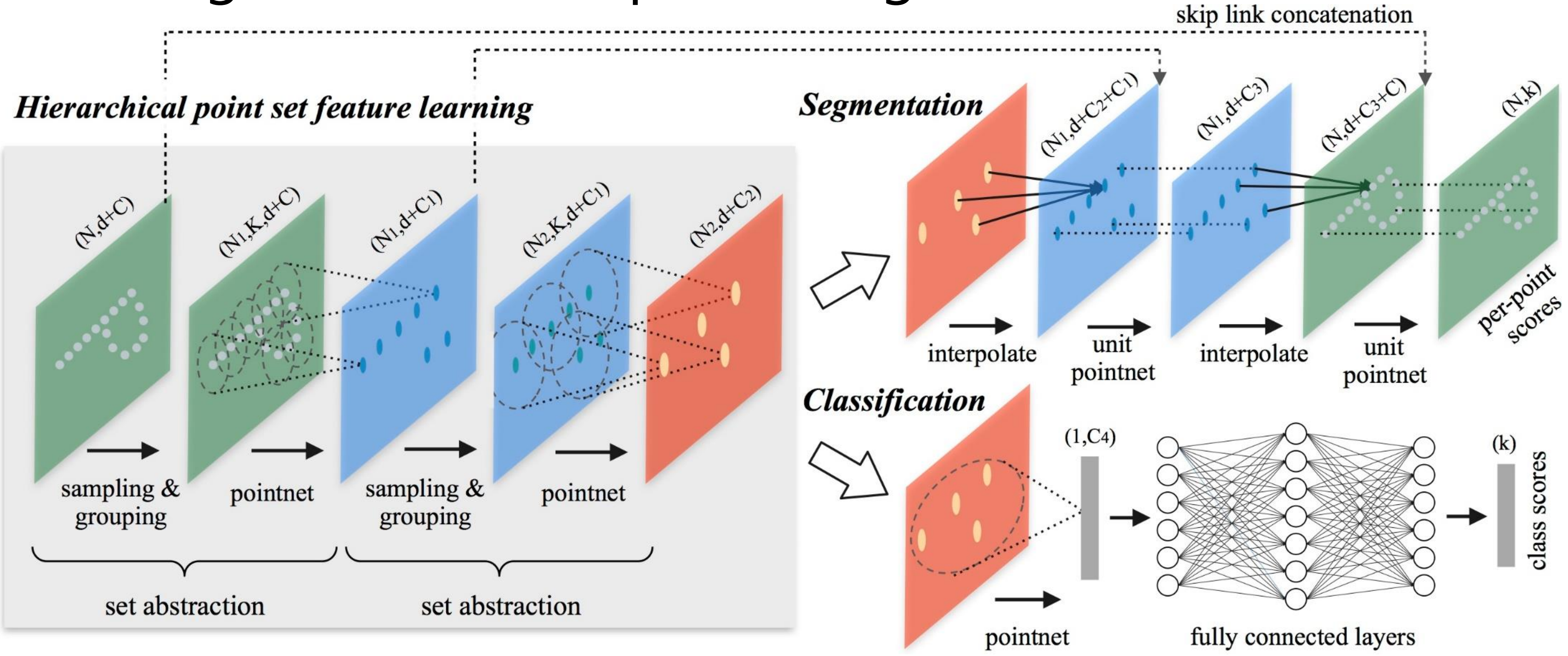
3D CNNs

	input	#views	accuracy avg. class	accuracy overall
SPH [12]	mesh	-	68.2	
3DShapeNets [29]	volume	1	77.3	84.7
VoxNet [18]	volume	12	83.0	85.9
Subvolume [19]	volume	20	86.0	**89.2**
LFD [29]	image	10	75.5	-
MVCNN [24]	image	80	**90.1**	-
Ours baseline	point	-	72.6	77.4
Ours PointNet	point	1	86.2	**89.2**

[Qi et al. ’17]

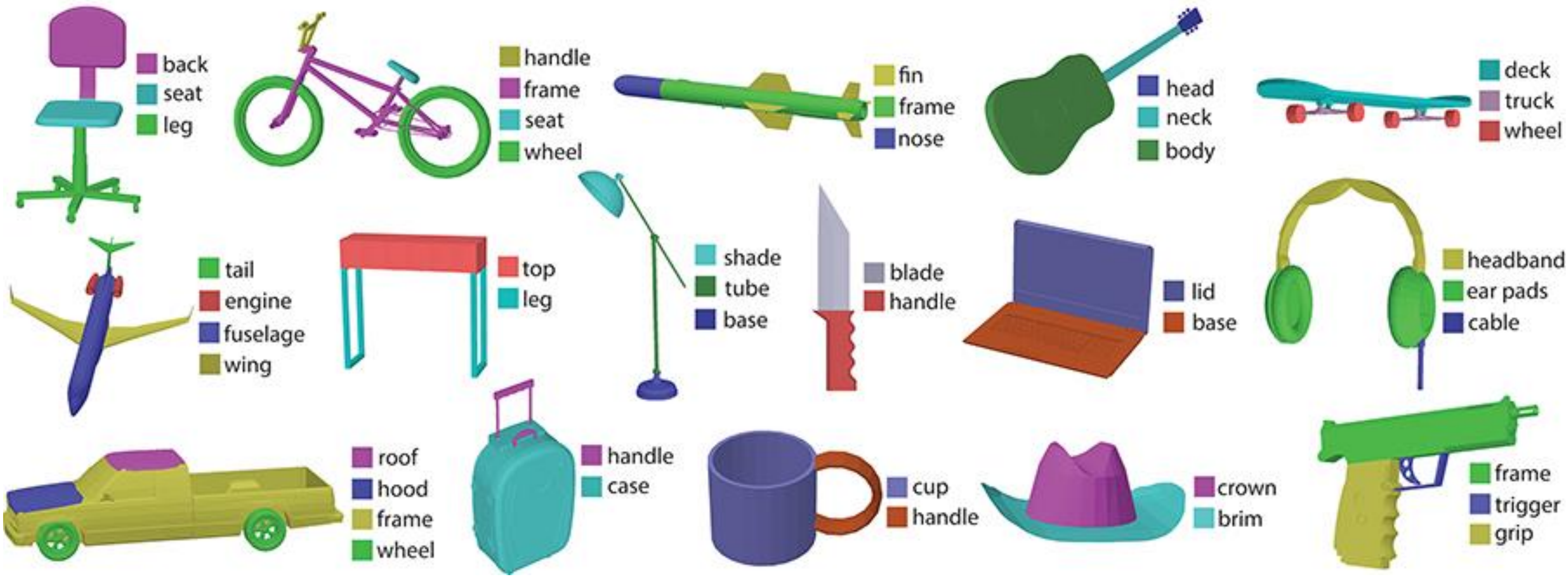
Understand Local Point Structures

- PointNet++: Capture local structures of point clouds by introducing a hierarchical processing



[Qi et al. '17]

Beyond Object Classification



How to decompose a 3D shape into parts?

Why decompose a 3D shape into parts?

Psychological understanding of shapes by their part decomposition
[Hoffmann et al. '84,'97]

Guidance for other tasks, e.g., shape matching or shape recognition, part-based modeling

3D Shape Segmentation: Labeling

- For each face/point on the surface of a shape, assign a part label

$f(p_i) \in$ { head, neck, torso, leg, tail, ear }

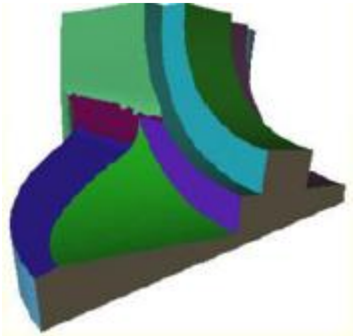
$f(p_i) \in$ { back, seat, leg, arm }

Unsupervised 3D Shape Segmentation

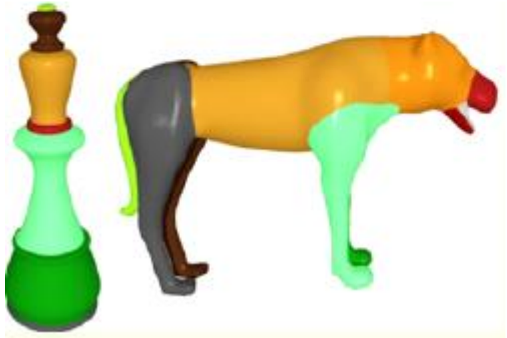
- Using only geometric features from a single shape

[Shalfman et al. '02]
K-Means

[Golovinskiy and Funkhouser '08]
Normalized Cuts



[Attene et al. '06]
Primitive Fitting



[Lai et al. '08]
Random Walks

Unsupervised 3D Shape Segmentation

- Can we leverage knowledge from a collection of shapes?

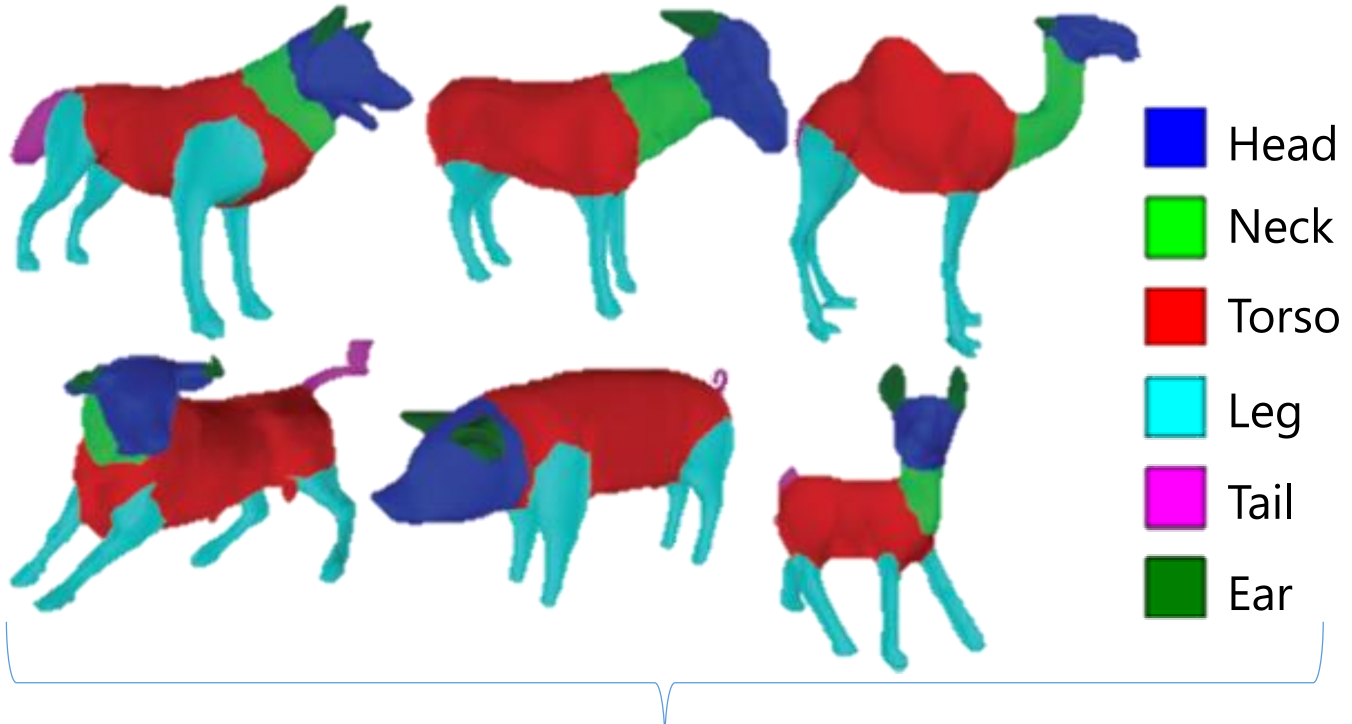
➢Consistent segmentations in shape collections

[Golovinskiy and Funkhouser '09]

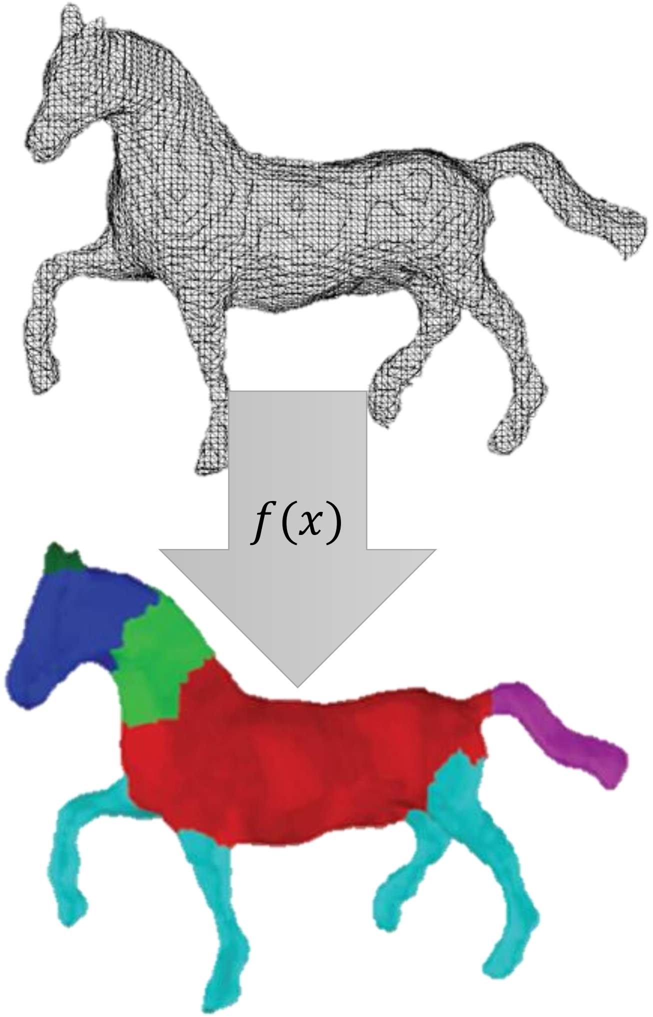
Requires collection of shapes in correspondence

Supervised Shape Segmentation

- Learn from examples

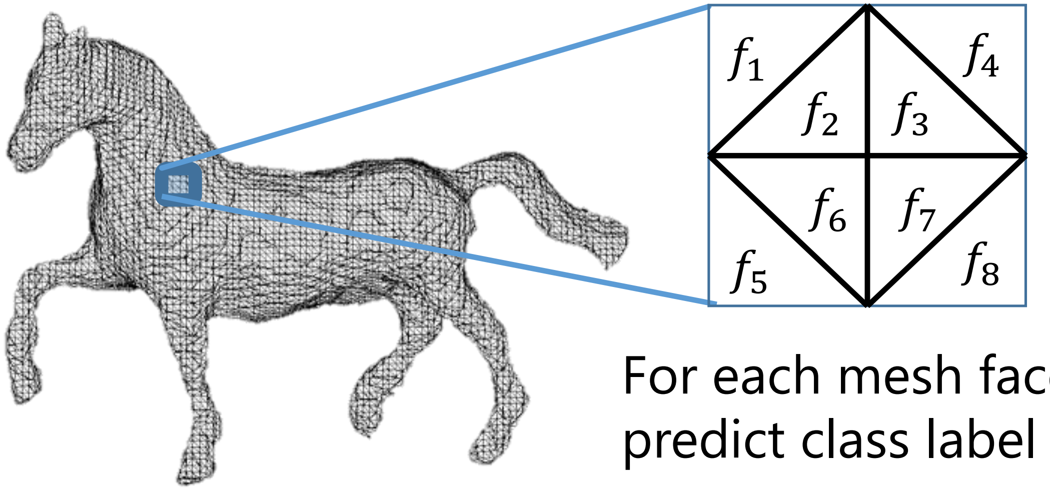


Annotated Train Set

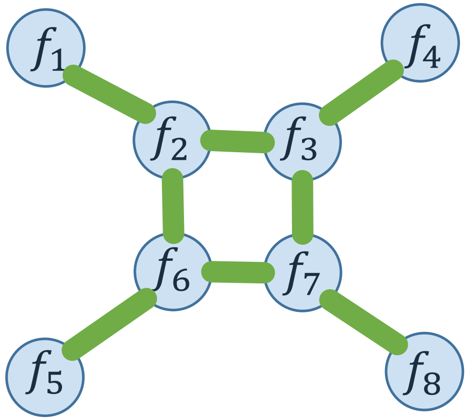


Conditional Random Fields for Labeling

- Conditional Random Fields (CRFs): probabilistic graphical model



For each mesh face, predict class label



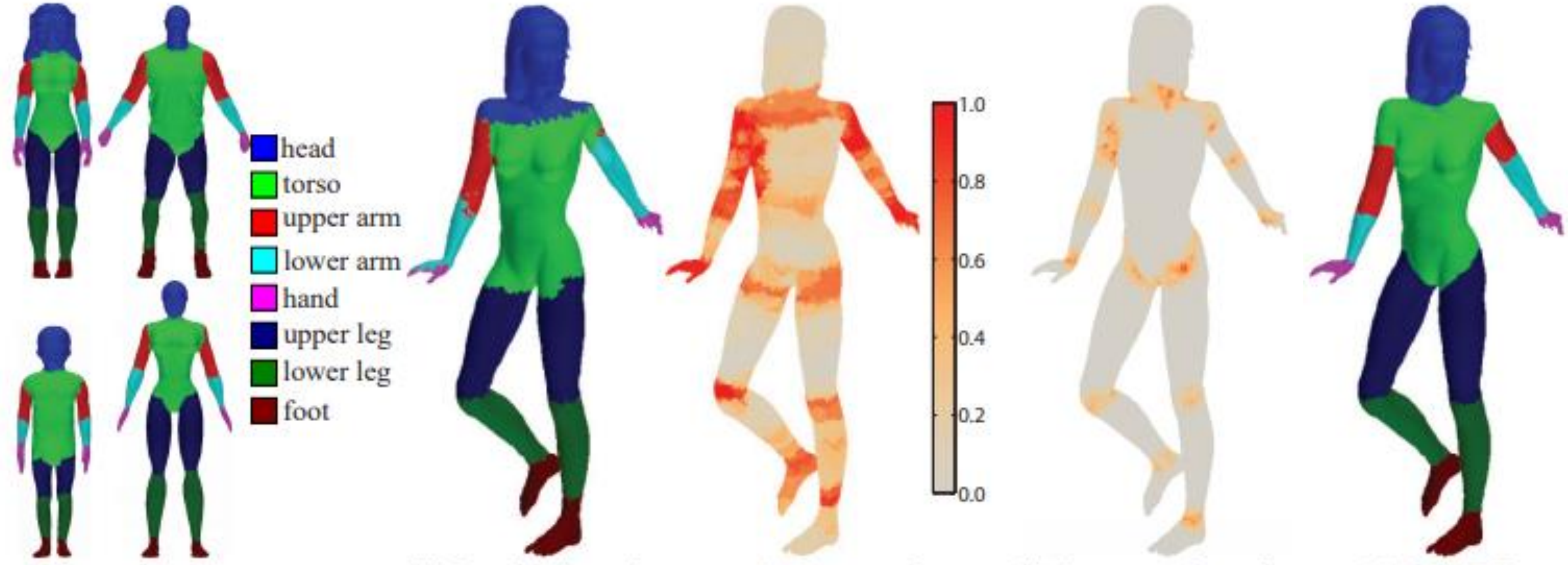
Observations: X
Random Variables: Y
Training: model $p(Y|X;\theta)$

$$y^* = argmin_y \left\{ \sum_i \alpha_i E_{unary}(y_i; x_i, \theta) + \sum_{i,j} l_{ij} E_{pair}(y_i, y_j; x_{ij}, \theta) \right\}$$

depends on face features, e.g., area, normal, curvature, spin image, etc.

face-face relationships

Conditional Random Fields for Labeling



(a) Training meshes (b) Result using only the unary classifier (c) Entropy of the unary classifier (d) Geometry-dependent pairwise term (g) Full CRF result

[Kalogerakis et al. ’10]

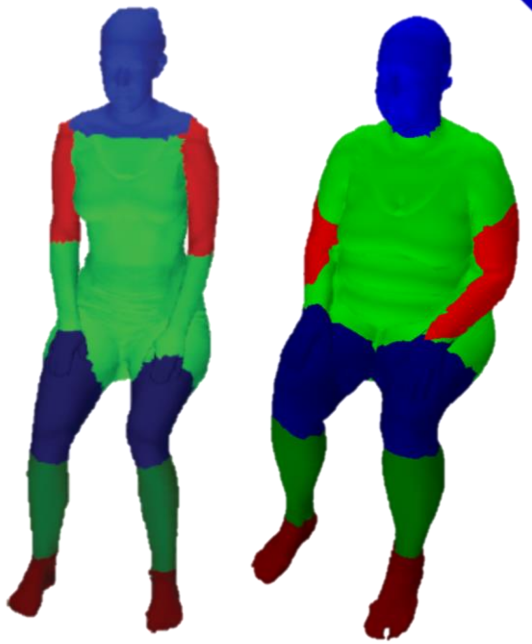
Conditional Random Fields for Labeling

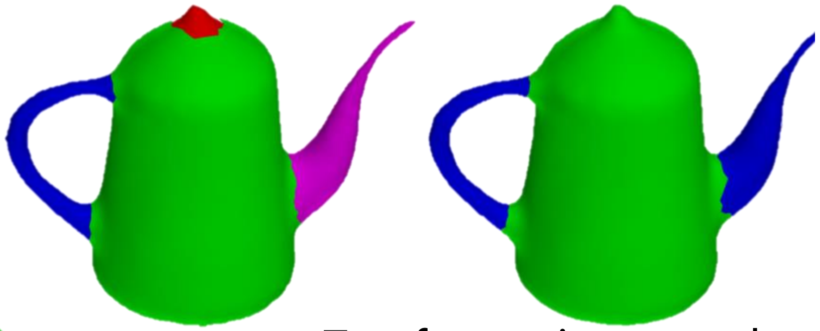
- Limitations

➢Labeling cannot distinguish instances (e.g., legs are labeled legs without notion of individual legs)

➢Need sufficient training data

➢Sensitivity to topology

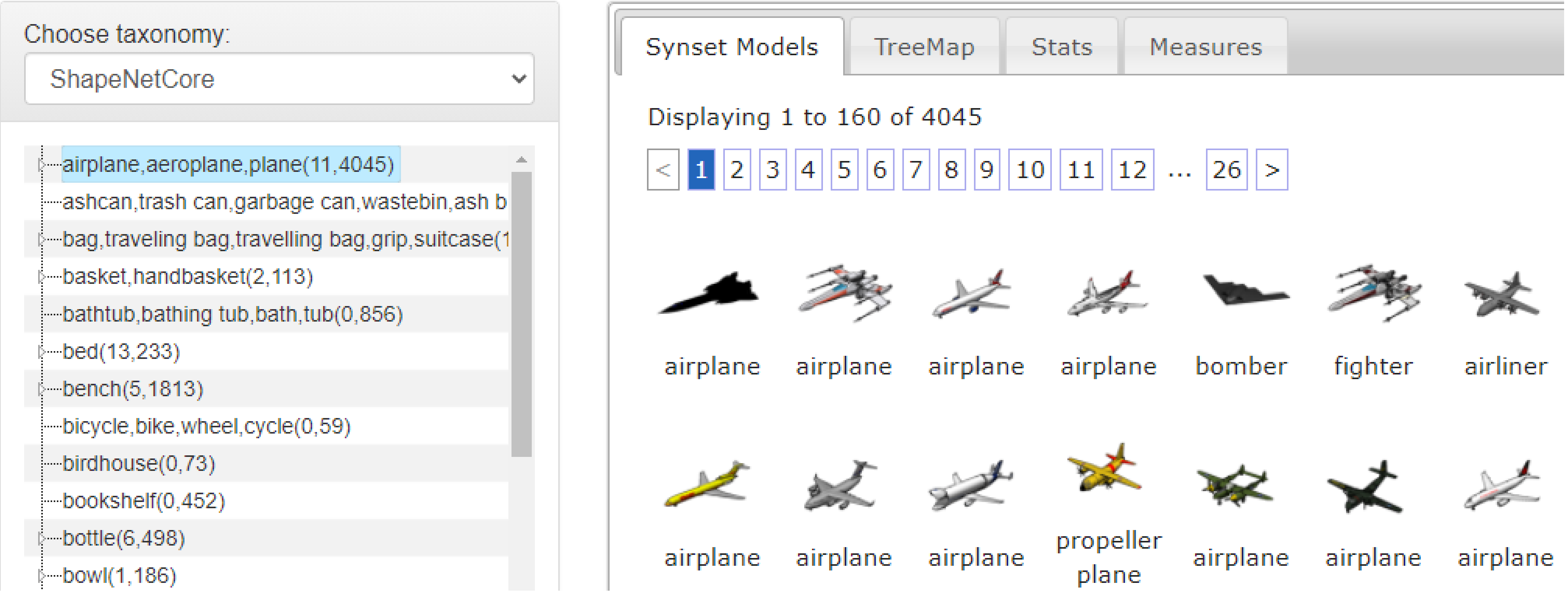




Too few train examples

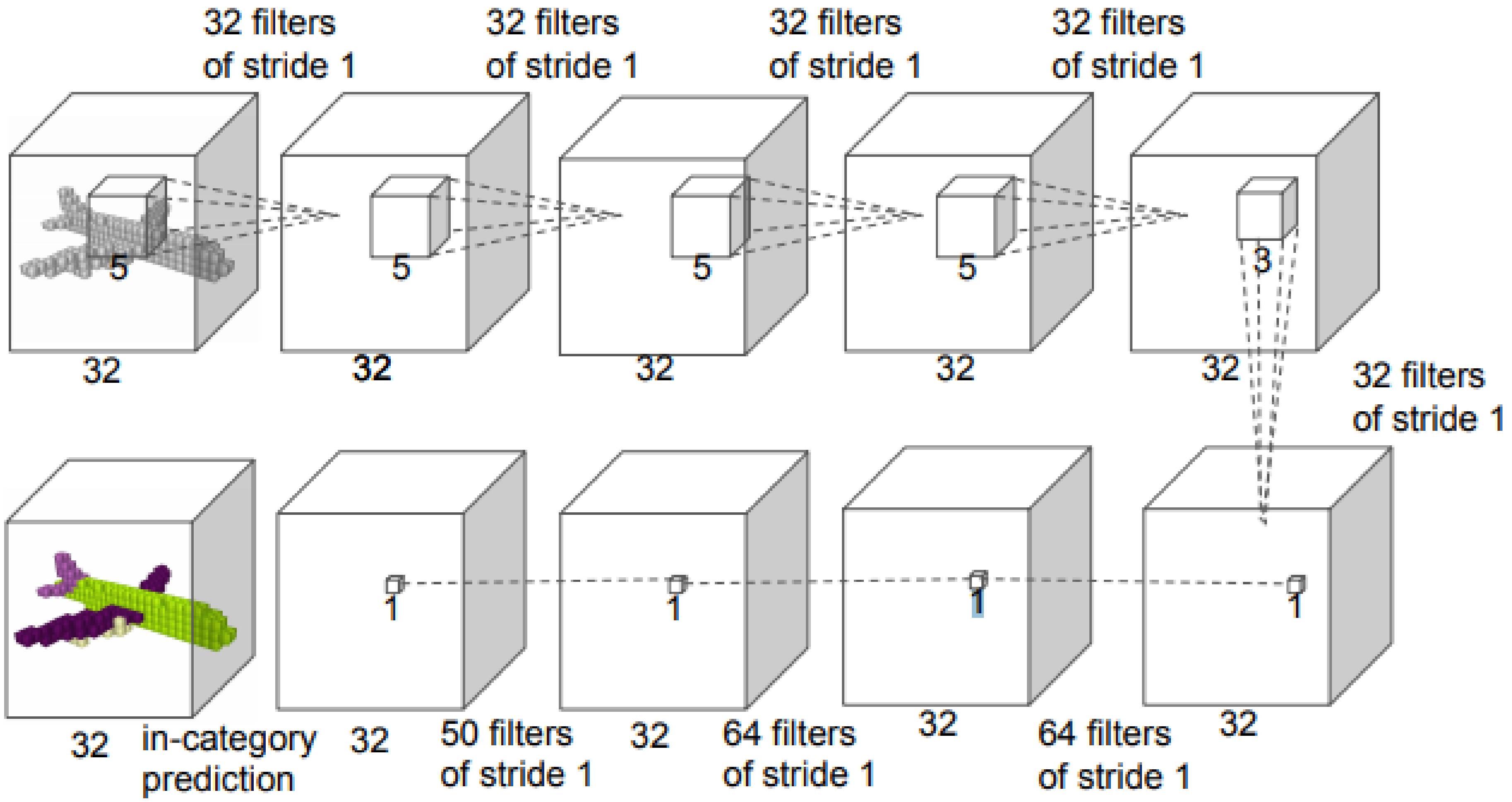
Deep Learning-based Part Segmentation

- Accelerated by the ShapeNet dataset (~51k models in ShapeNetCore)



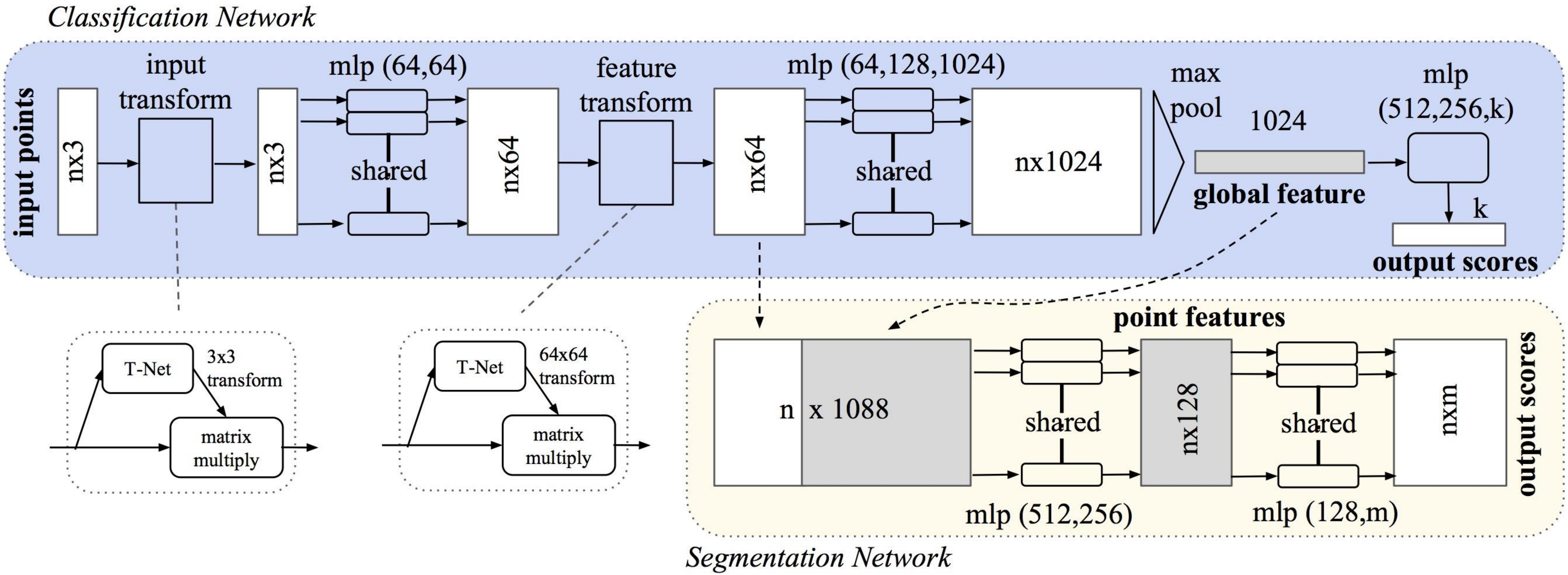
https://shapenet.org/

3D CNNs for Part Segmentation



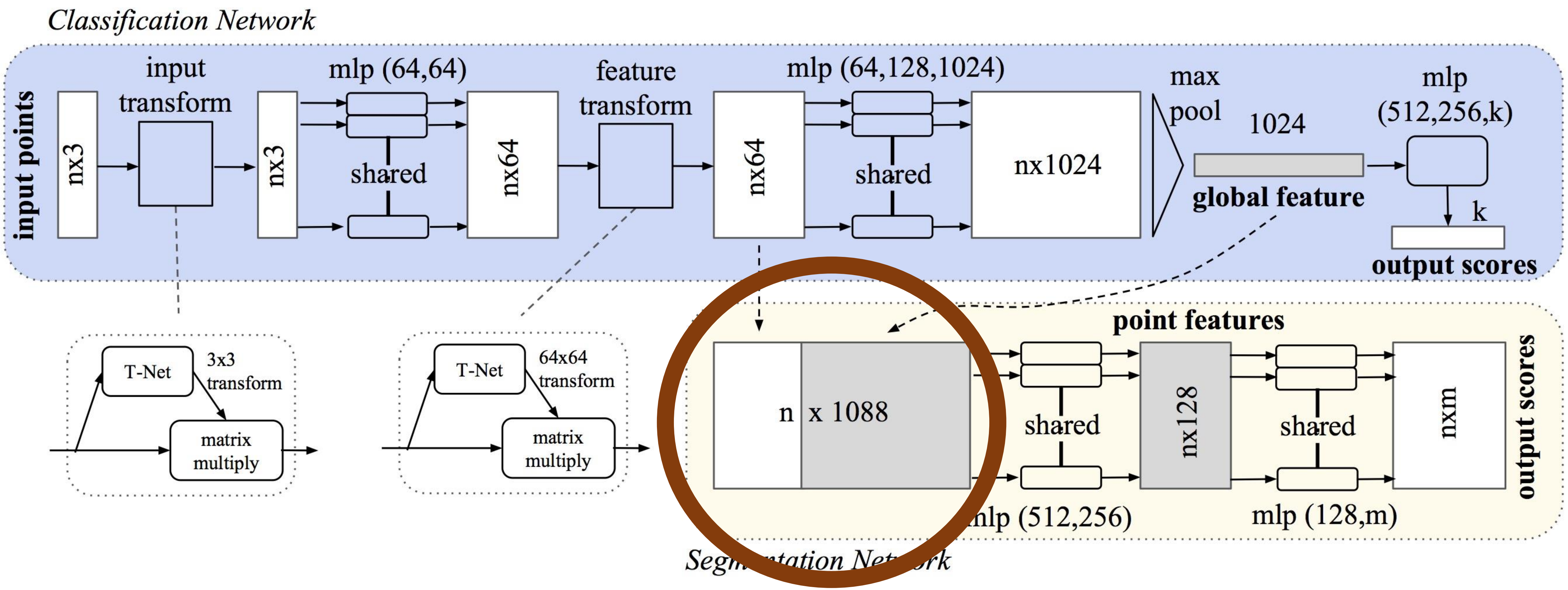
Example simple network architecture

Part Segmentation with Points: PointNet



[Qi et al. ’17]

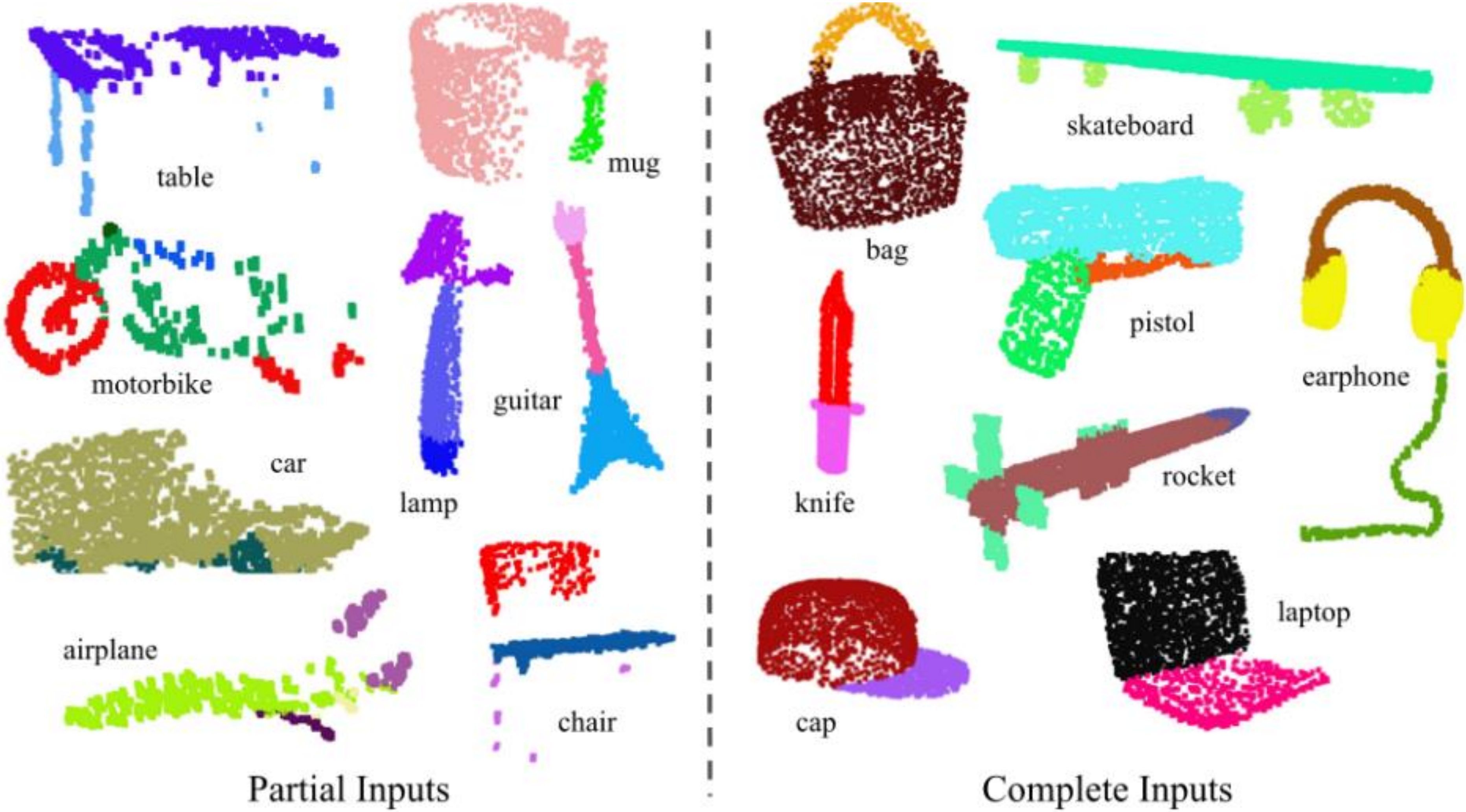
Part Segmentation with Points: PointNet



Concatenate local and global features

[Qi et al. '17]

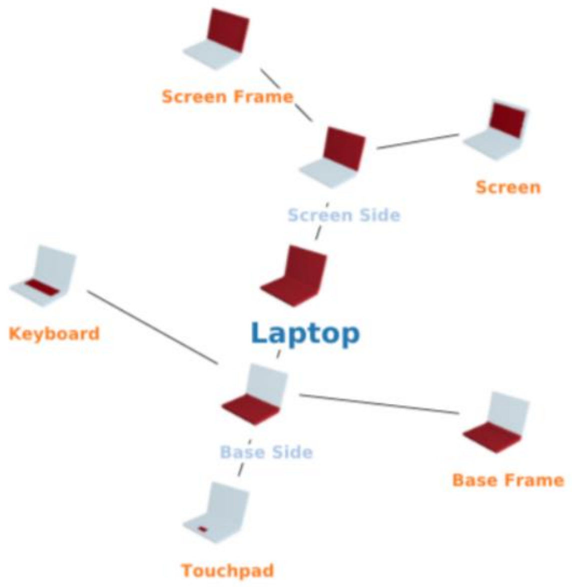
Part Segmentation with Points: PointNet

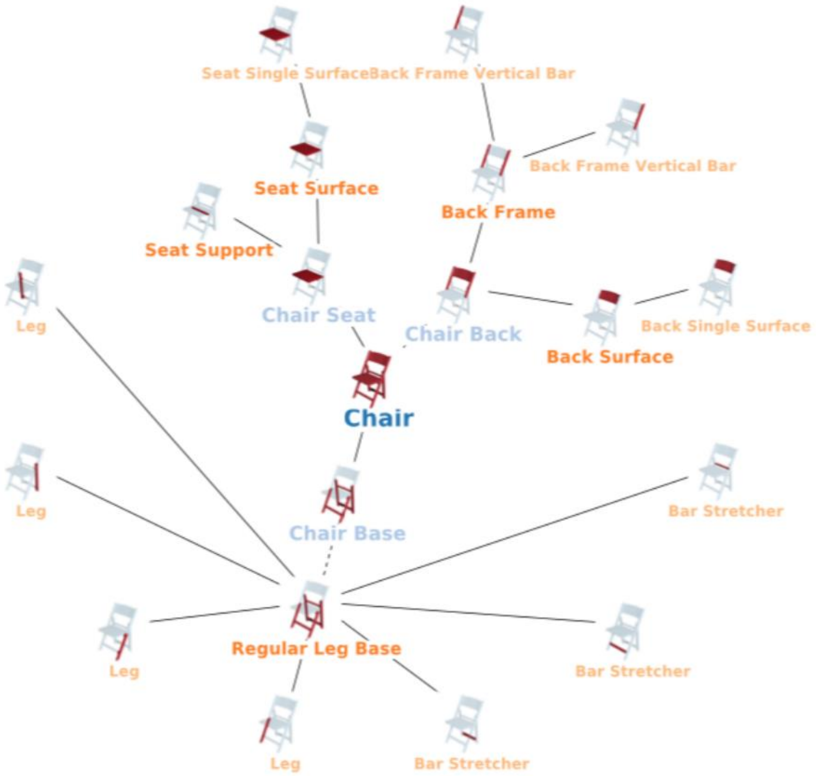


[Qi et al. ’17]

Part Structures are Hierarchical

- Don't have to label parts as just a set of categories

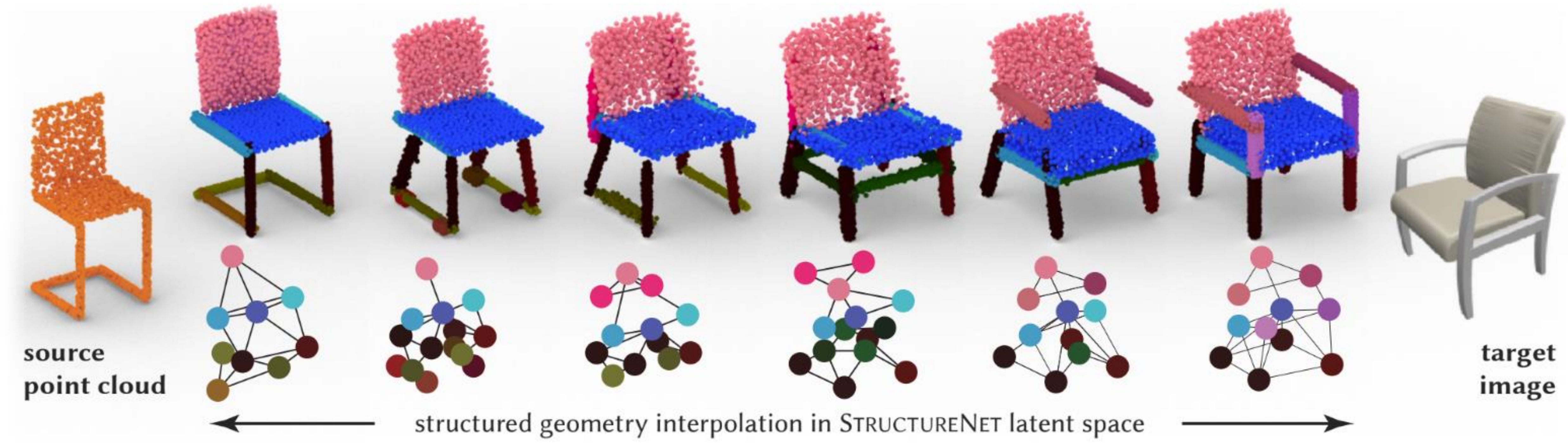




- Leverage structure of part labels

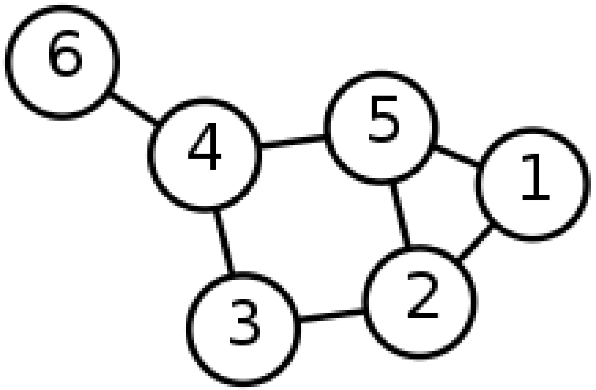
Encoding Object Part Hierarchies

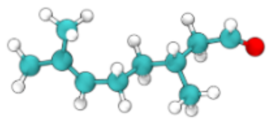
- Structurenet: Encode object parts as a hierarchical graph

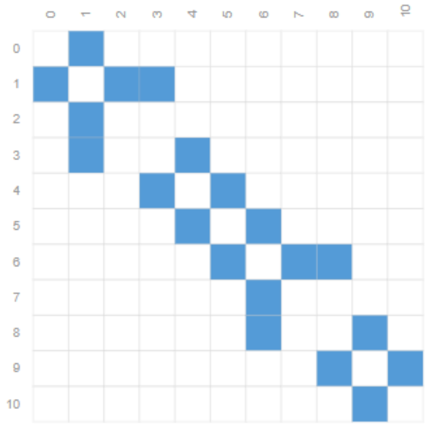


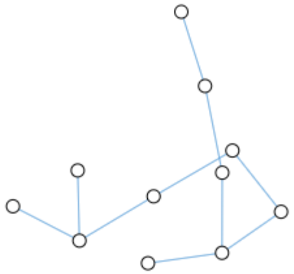
[Mo et al. '19]

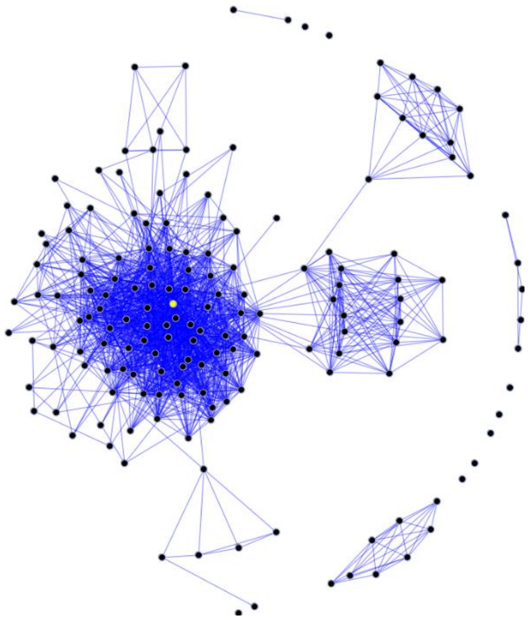
How to Interpret Graphs with Deep Learning?





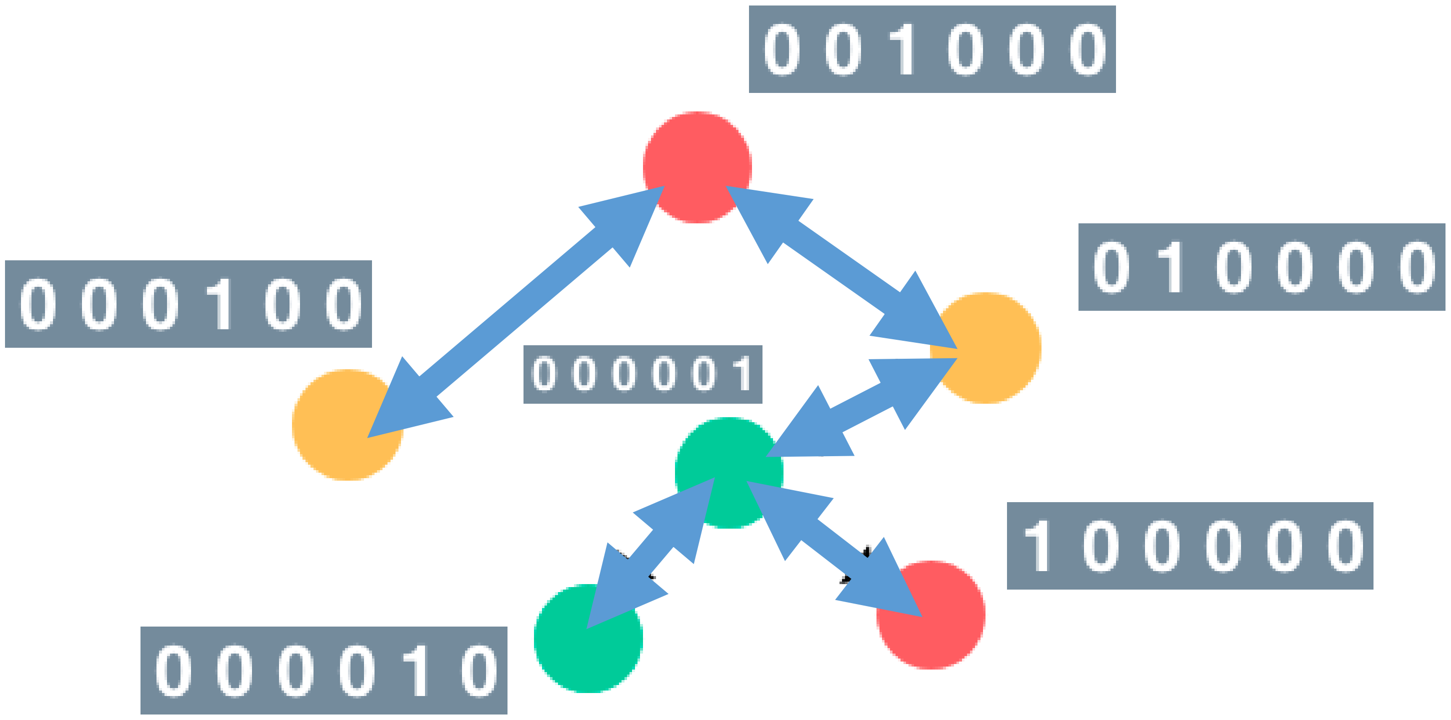






Message-Passing Graph Neural Networks

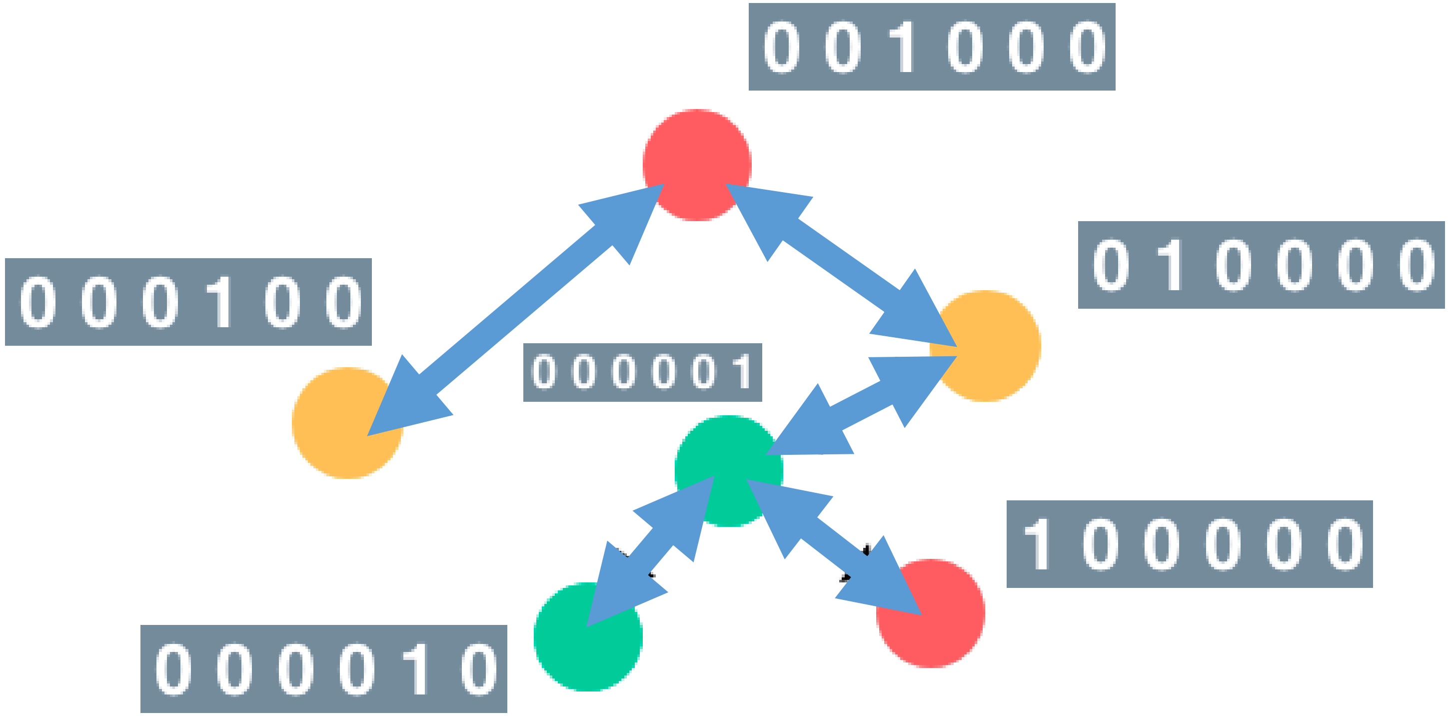
- Message-Passing



https://medium.com/dair-ai/an-illustrated-guide-to-graph-neural-networks-d5564a551783

Message-Passing Graph Neural Networks

- Message-Passing
- Each node has a feature
- Each edge has a feature
- (Optional) global graph feature



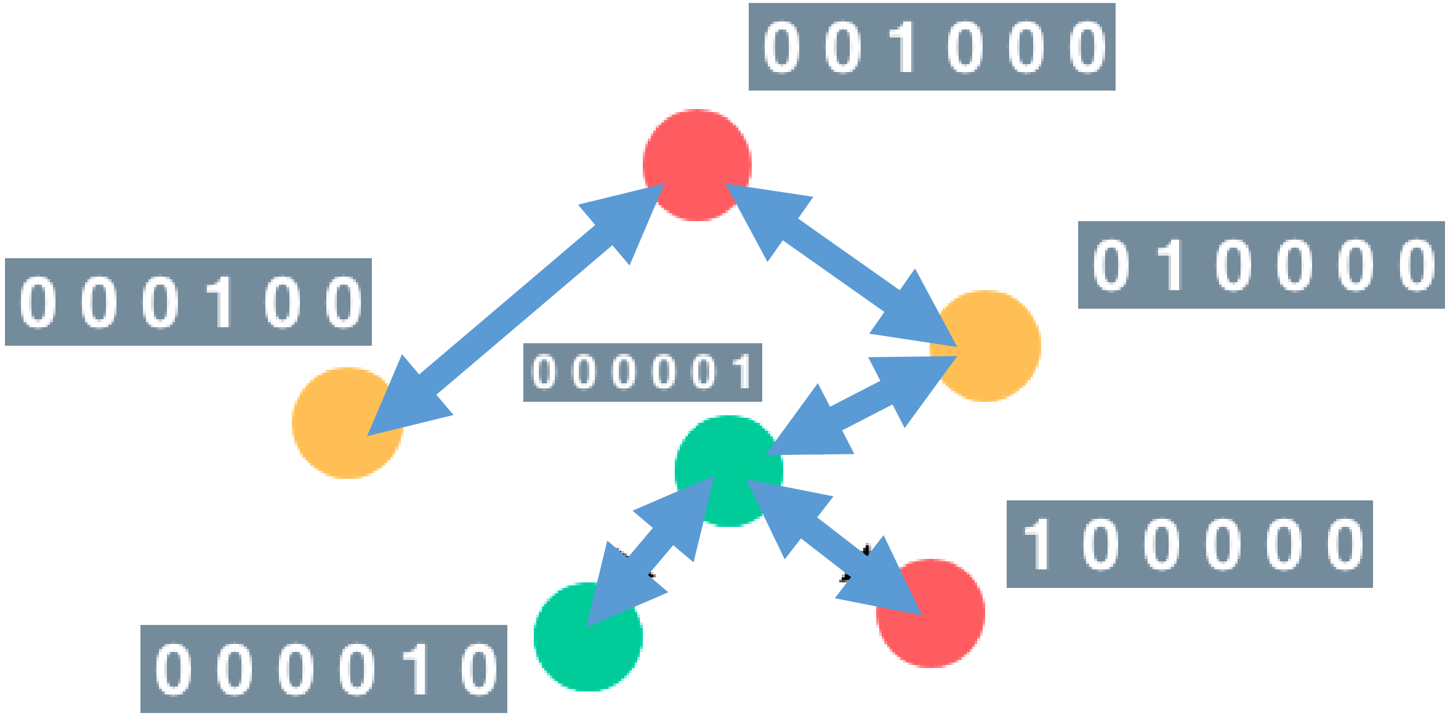
https://medium.com/dair-ai/an-illustrated-guide-to-graph-neural-networks-d5564a551783

Message-Passing Graph Neural Networks

- Message-Passing

- Tasks:
  - Global prediction
  - Node-level
  - Edge-level

- E.g., part graph for one shape:
  - Nodes: part classes
  - Edges: part relations

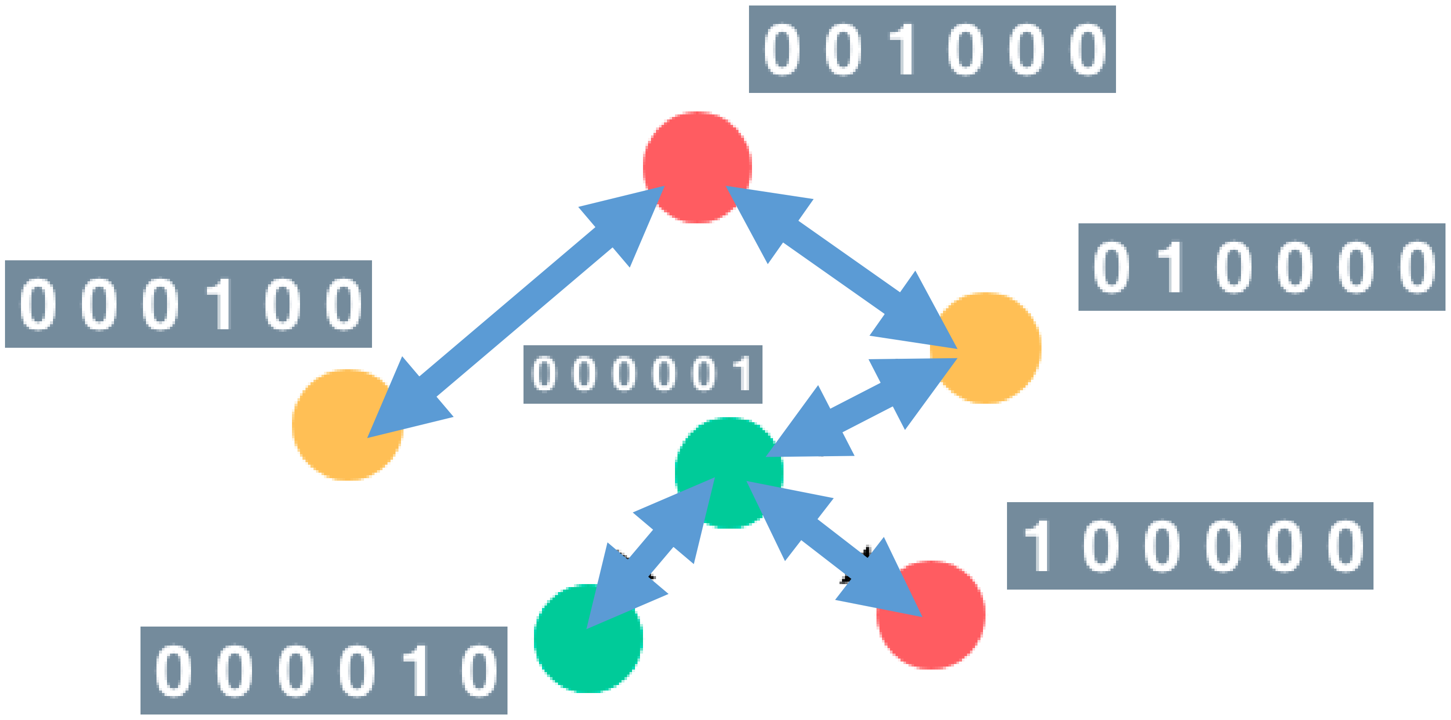


https://medium.com/dair-ai/an-illustrated-guide-to-graph-neural-networks-d5564a551783

Message-Passing Graph Neural Networks

- Message-Passing

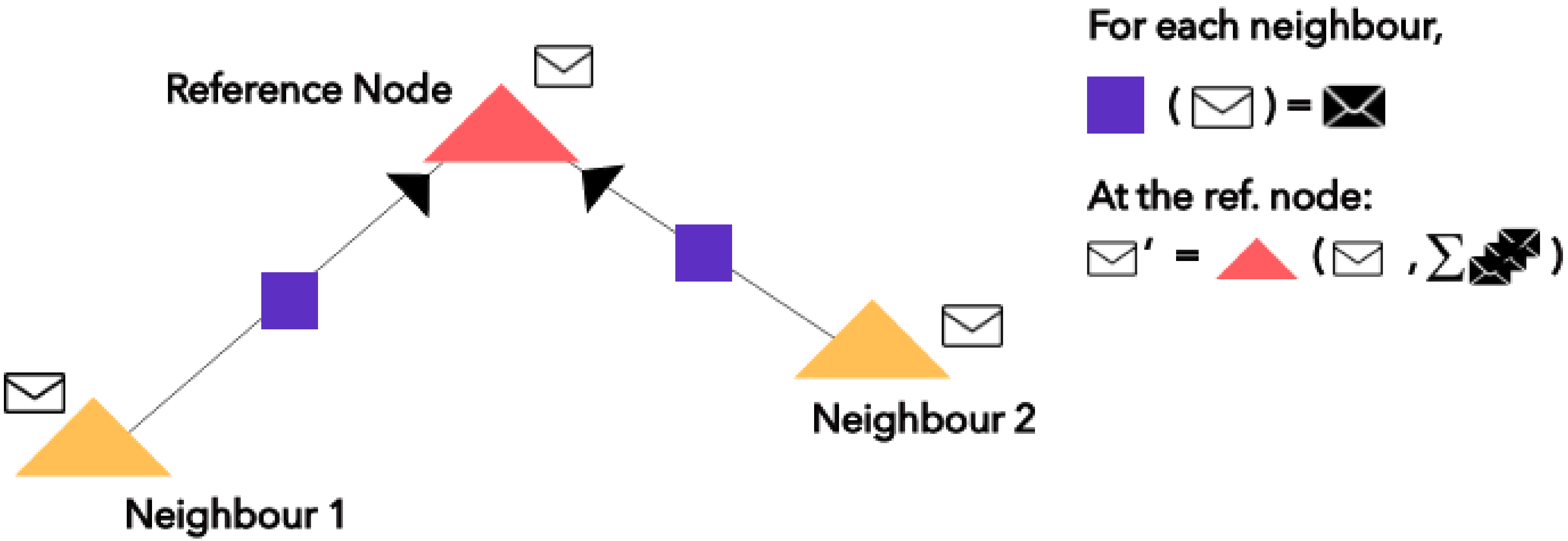
- Challenges:
  - No regular grid
  - Must be vertex/edge order agnostic



https://medium.com/dair-ai/an-illustrated-guide-to-graph-neural-networks-d5564a551783

Message-Passing Graph Neural Networks

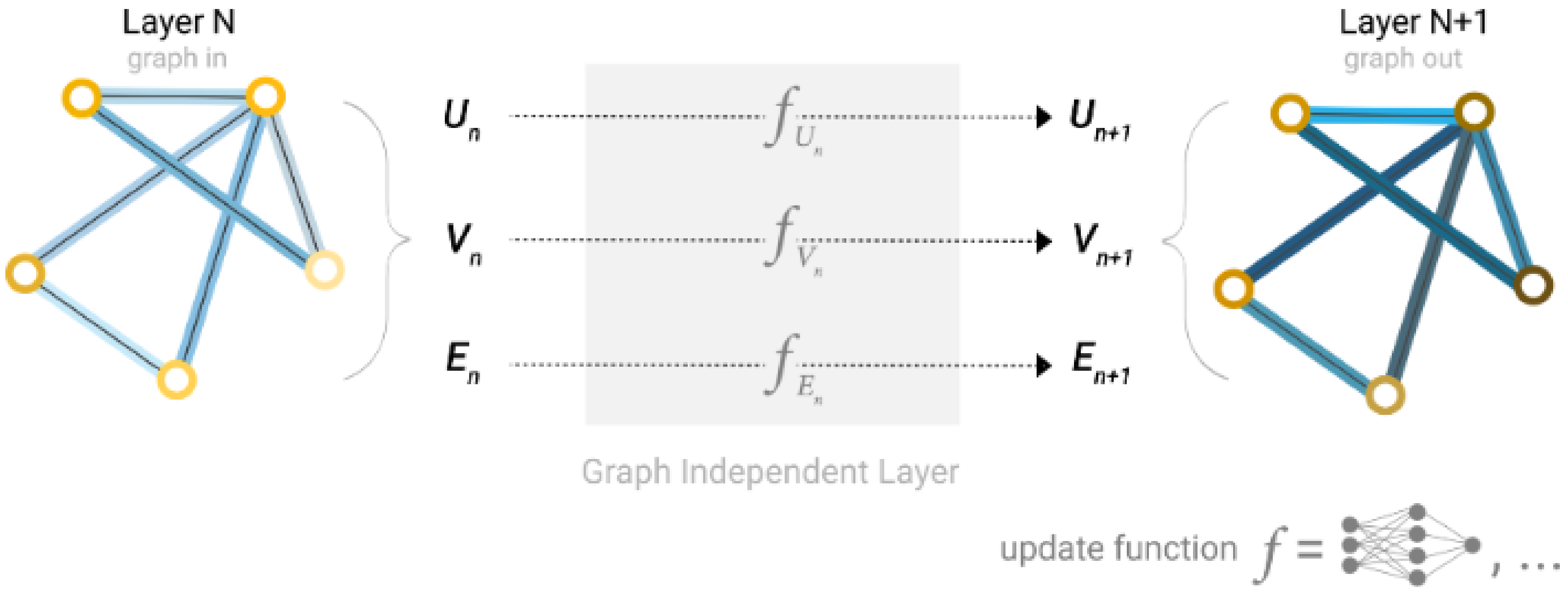
- Message-Passing



https://medium.com/dair-ai/an-illustrated-guide-to-graph-neural-networks-d5564a551783

Message-Passing Graph Neural Networks

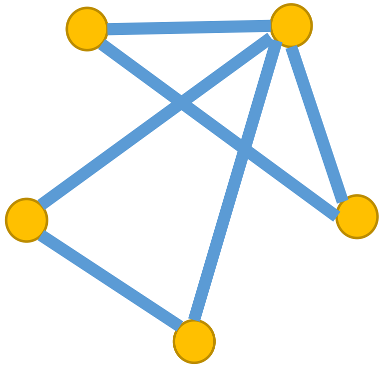
- Message-Passing



https://distill.pub/2021/gnn-intro/

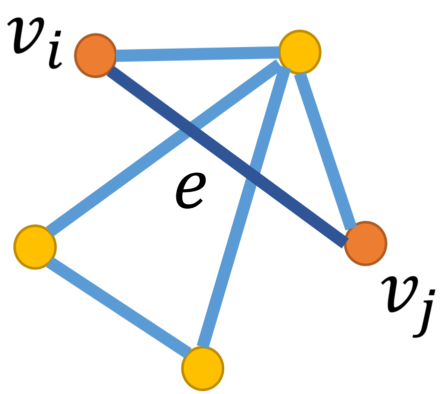
Message-Passing Graph Neural Networks

- Message-Passing



Message-Passing Graph Neural Networks

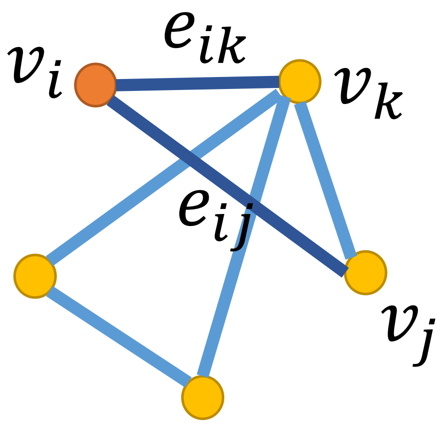
- Message-Passing
- For each edge: aggregate information from edge vertices



$$f_e = h(v_i, v_j)$$

Message-Passing Graph Neural Networks

- Message-Passing
- For each vertex: aggregate information from all incident edges

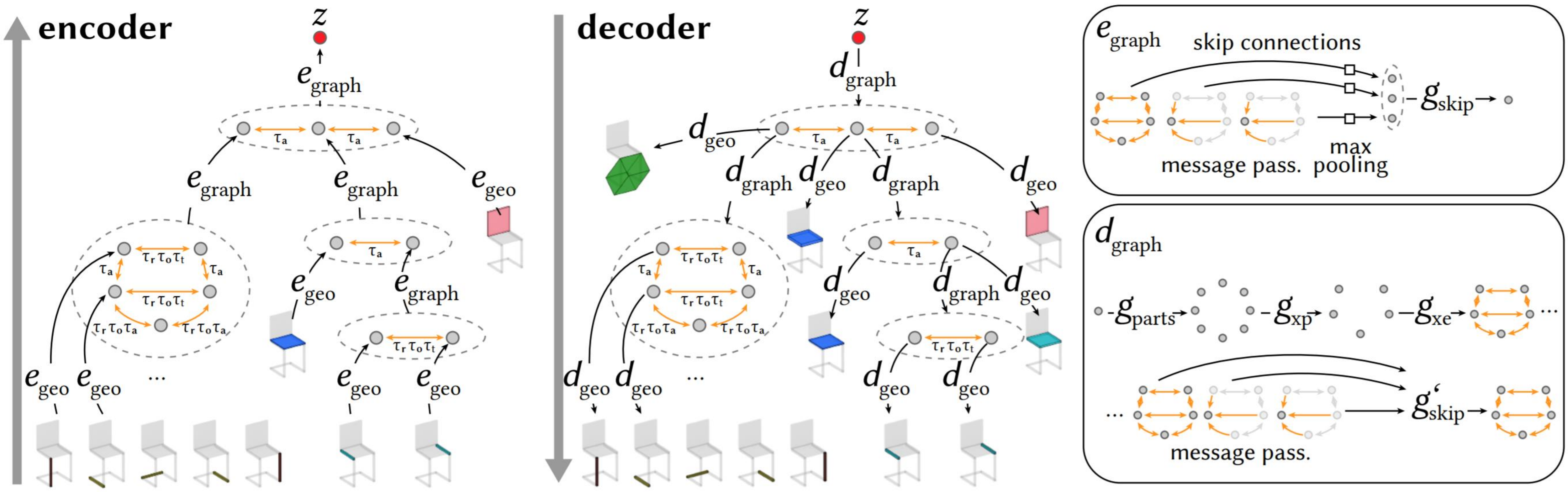


$$f_v = h(e_{ij}, e_{ik}, \dots, e_{in})$$

- Note: aggregation function must be order agnostic!

StructureNet

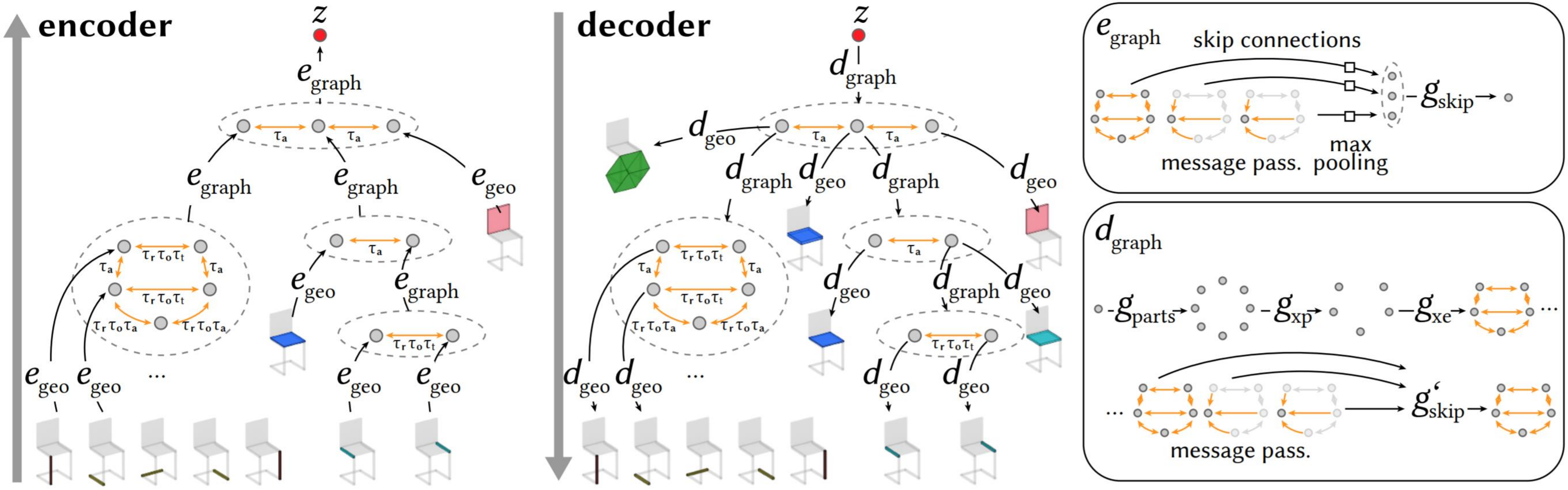
- Part hierarchy graph -> graph variational autoencoder



[Mo et al. ’19]

StructureNet

- Part hierarchy graph -> graph variational autoencoder



[Mo et al. '19]

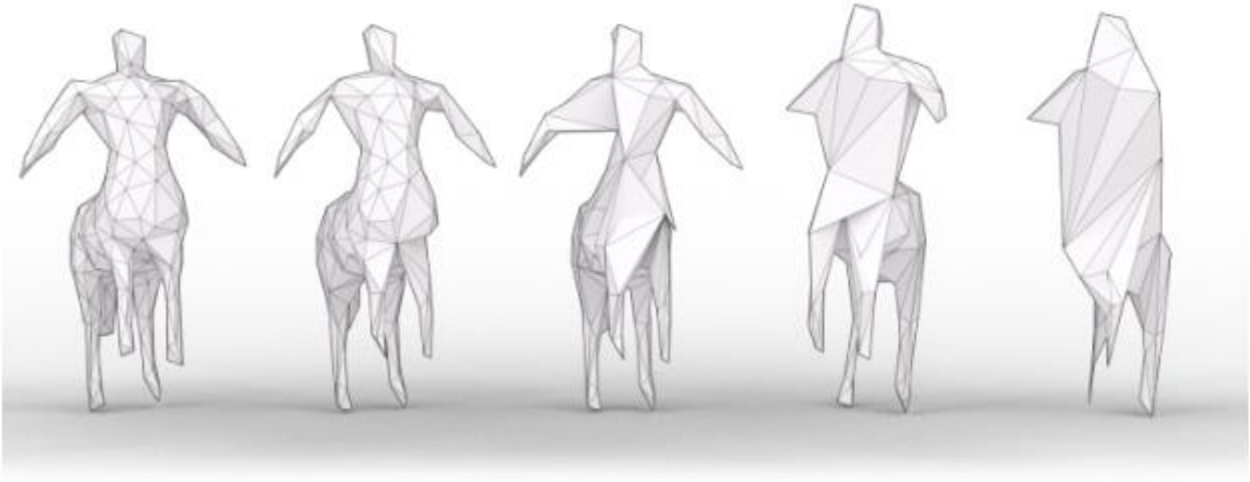
StructureNet

- Predict part hierarchy and geometry for real-world scans
- Train autoencoder on shape parts; then train encoder for real scanned objects into the autoencoder space

[Mo et al. '19]

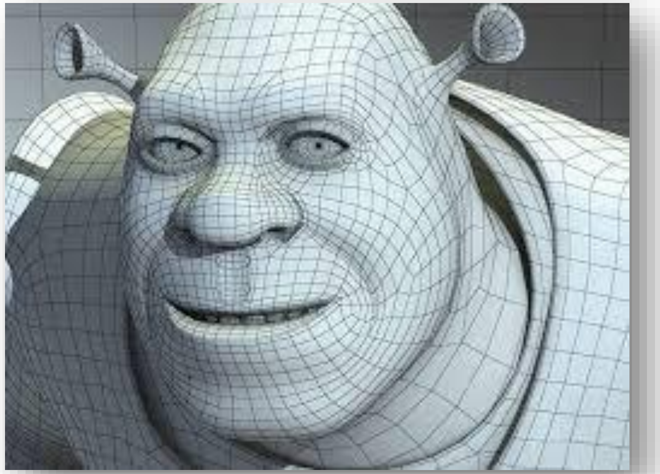
Interpret Shapes as Meshes

- MeshCNN: defines edge-based convolutions and pooling
- Applied to shape classification and segmentation



[Hanocka et al. '19]

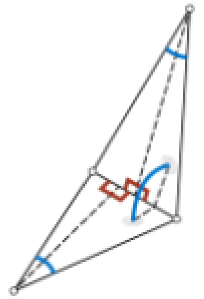
Interpret Shapes as Meshes



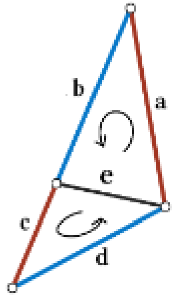
- Shape meshes are graphs
- Much larger than part graphs
- Cannot just use message-passing on raw shape mesh

MeshCNN: Edge-based Convolution

- Triangle mesh
- Features per edge:
    - Each edge characterized by dihedral angle, 2 inner angles, 2 edge-length ratios



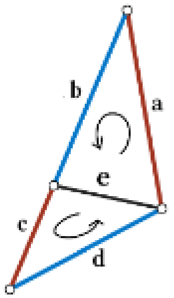
- Convolution on edge neighborhood
    - Each edge incident to two faces
    - Four edge neighbors per edge



[Hanocka et al. ’19]

MeshCNN: Edge-based Convolution

- Convolution on edge neighborhood
  - Each edge incident to two faces
  - Four edge neighbors per edge
  - Four neighbors either (a, b, c, d) or (c, d, a, b)
  - For order-invariance: apply symmetric functions to ambiguous pairs

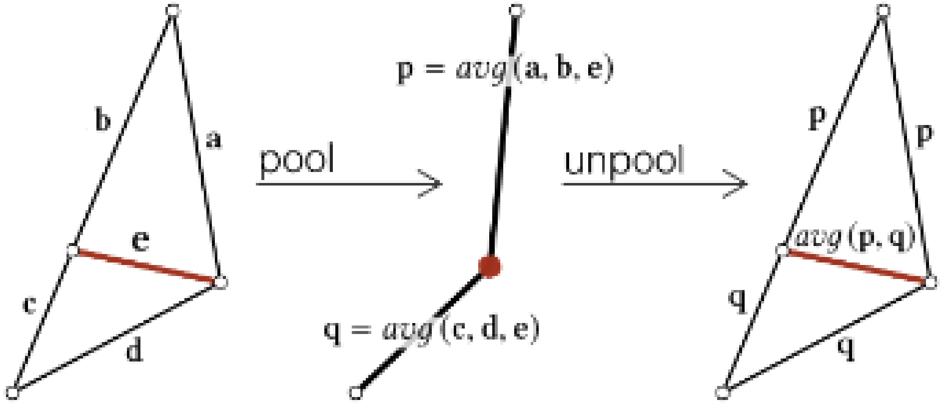


[Hanocka et al. ’19]

MeshCNN: Edge-based Convolution

- Mesh Pooling

- Downsampling operation, enabling encoder (-decoder) like structures in traditional CNNs



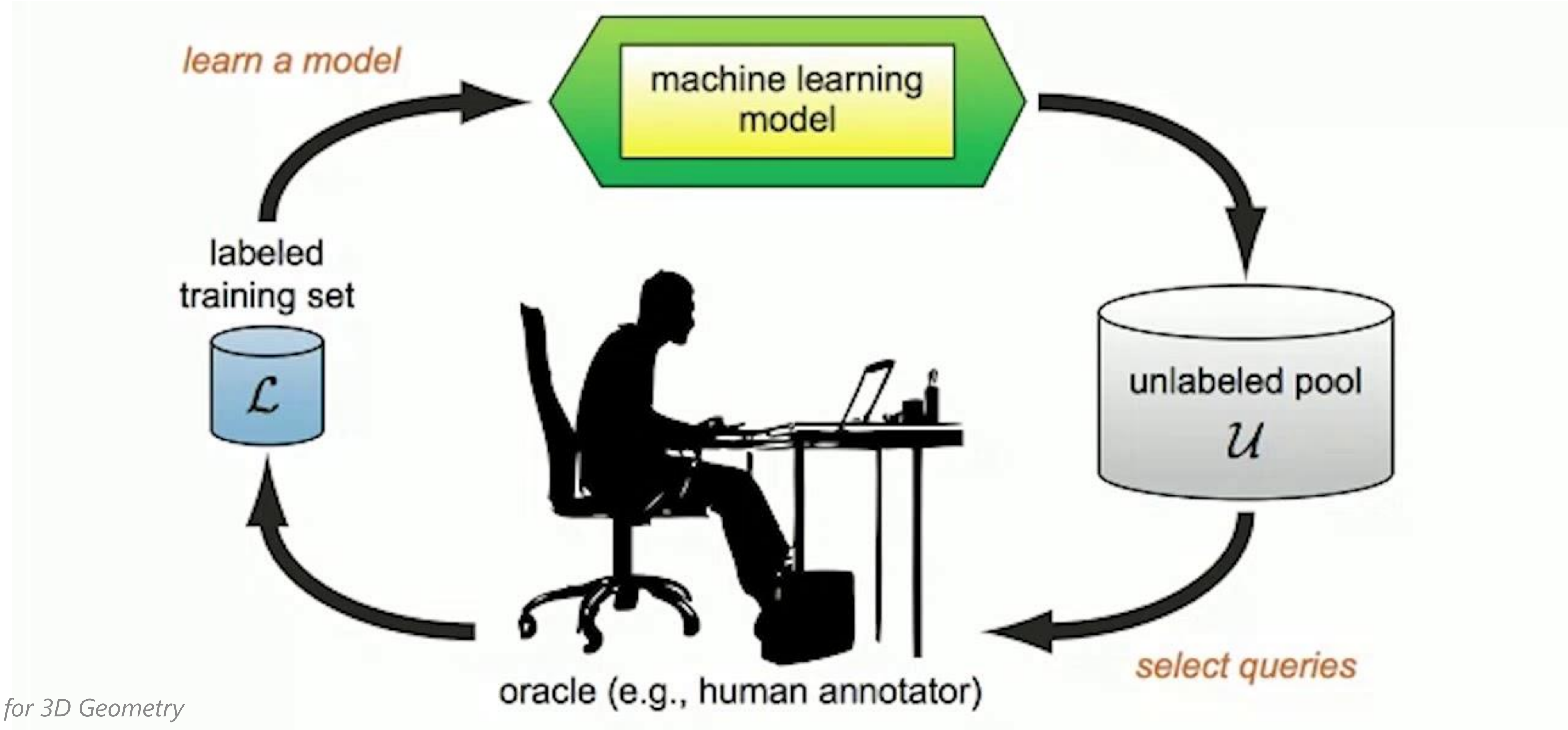
[Hanocka et al. ’19]

Deep networks on meshes

- Can potentially maintain original data structure
- Challenge: with meshes, want to understand surface, not graph structure
- Challenge: difficult to scale to practical mesh sizes
  - E.g., MeshCNN limited to ~2000 edges

Active Learning

- Human in the loop!



Active Part Segmentation Learning

- Data-driven performance is great
- Annotations for large datasets is expensive

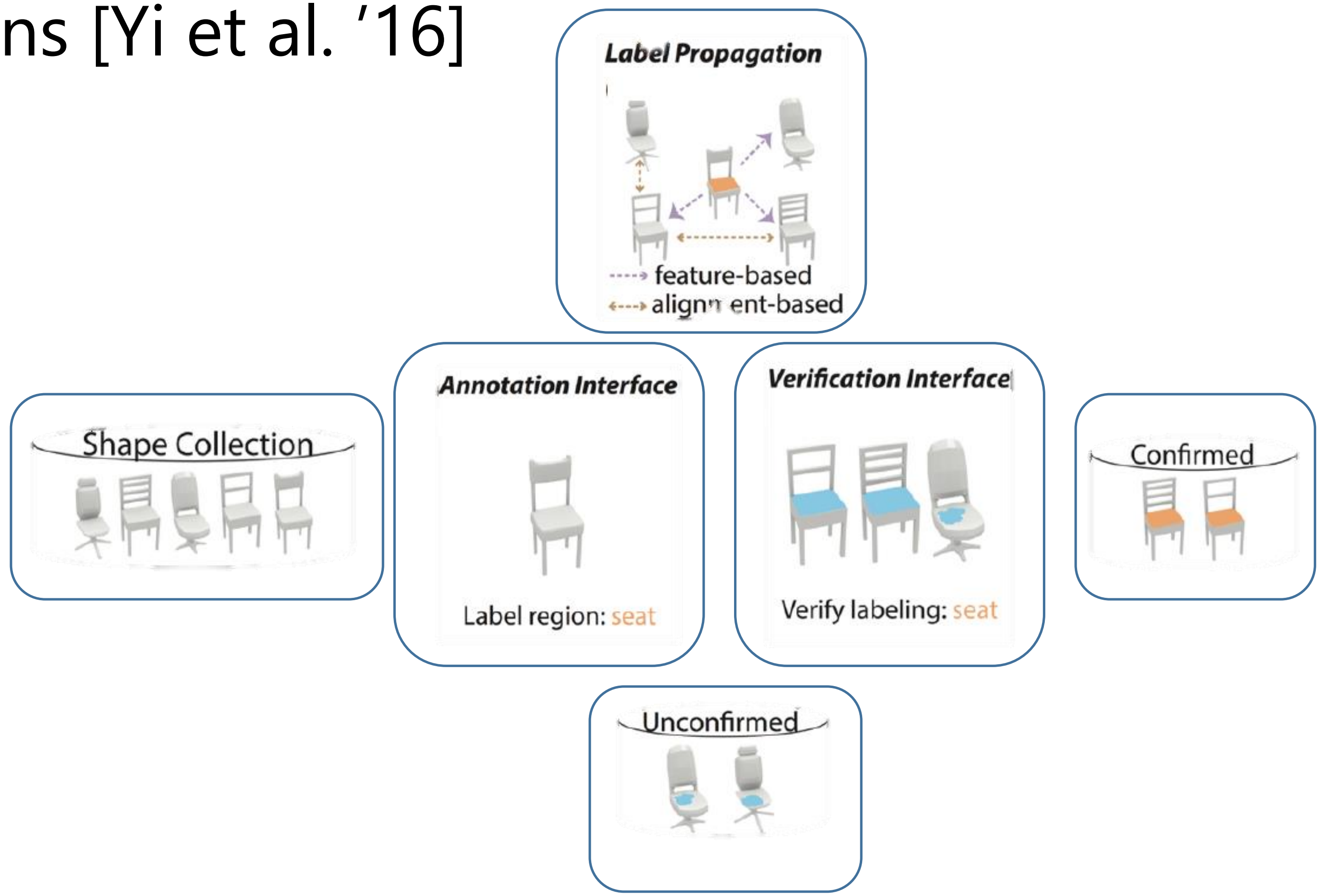
- Automatic annotations are cheap, but unreliable

- Combine automated annotations with human in the loop
    - Automated algorithm to propose where to annotate next
    - Human verification of automated algorithm to generate annotations
    - etc.

➢ Re-train automated algorithm based on new annotations

Active Part Segmentation Learning

- A Scalable Active Framework for Region Annotation in 3D Shape Collections [Yi et al. '16]



Optimize for best use of annotater time per accurate labels

So far:

- Discriminative tasks (classification, segmentation) on shapes
- Understanding on shapes as voxels, points, graphs, meshes
- How to tackle generative tasks?
- How to scale to scenes?
  - Challenges: no fixed sizes, no canonical spaces

Useful Shape Datasets

- ShapeNet [Chang et al. ’15]
  - https://shapenet.org/
- PartNet [Mo et al. ’19]
  - https://partnet.cs.stanford.edu
- Amazon Berkeley Objects Dataset [Collins et al. ’22]
  - https://amazon-berkeley-objects.s3.amazonaws.com/index.html
- Objaverse [Deitke et al. ’22]
  - https://objaverse.allenai.org/