

# L9: 3D Instance Segmentation

Hao Su

Ack: Jiayuan Gu and Zhan Ling for helping to prepare slides

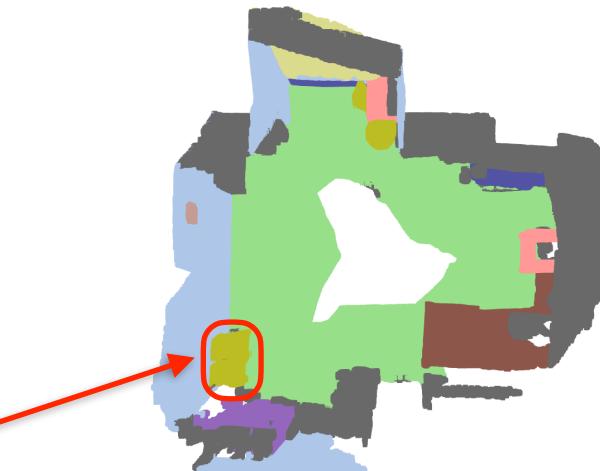
# Agenda

- Introduction
- Metric
- Top-down approaches
- Bottom-up approaches

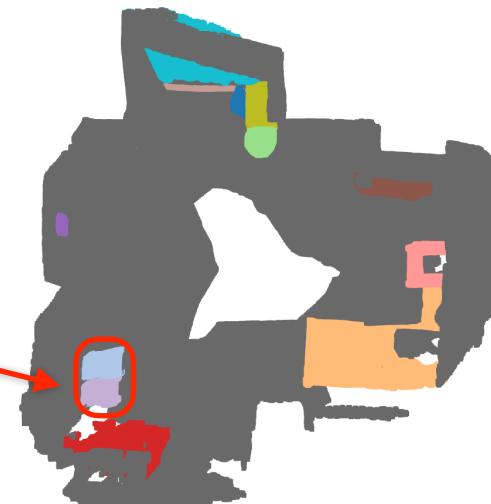
# Semantic Segmentation v.s. Instance Segmentation



Input



Semantic segmentation

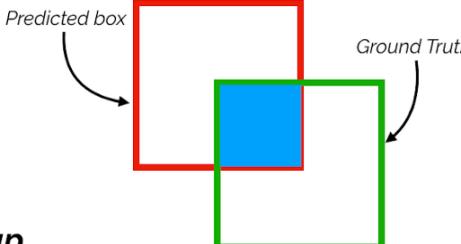


# Goal of Instance Segmentation

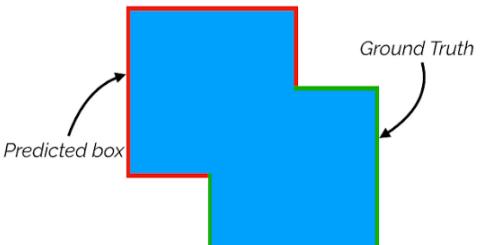
- Find as **many** objects as possible from the scene.
- Segmentation results should be as **accurate** as possible.

# Intersection-over-Union (IoU)

- For two sets  $A$  and  $B$ ,  $IoU(A, B) = \frac{|A \cap B|}{|A \cup B|}$ .



The diagram shows two overlapping rectangular boxes. A red box labeled "Predicted box" and a green box labeled "Ground Truth" overlap in a central blue area. Arrows point from the labels to their respective boxes.

$$\text{Intersection over Union (IoU)} = \frac{\text{Area of Overlap}}{\text{Area of Union}} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$


The diagram shows two non-overlapping rectangular boxes. A large blue box labeled "Predicted box" and a smaller green box labeled "Ground Truth" do not share any common area. Arrows point from the labels to their respective boxes.

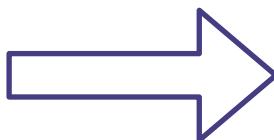
# Intersection-over-Union (IoU)

- Can also be used for measuring segmentation



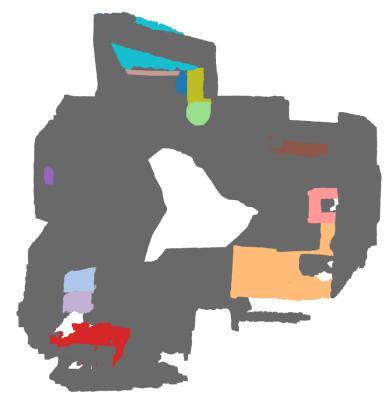
0.50	0.90	1.16
0.34	4.38	2.23
5.96	3.48	1.38
2.51	6.78	0.92
1.50	6.95	1.84
0.37	1.49	1.22
3.13	6.50	0.90
3.09	5.85	1.13
4.35	2.10	1.26
5.29	5.06	1.11
...	...	...

Point cloud



Chair1
Chair1
Chair2
Bed1
Picture1
Picture2
Chiar3
Curtain1
Chiar4
Bed2
...

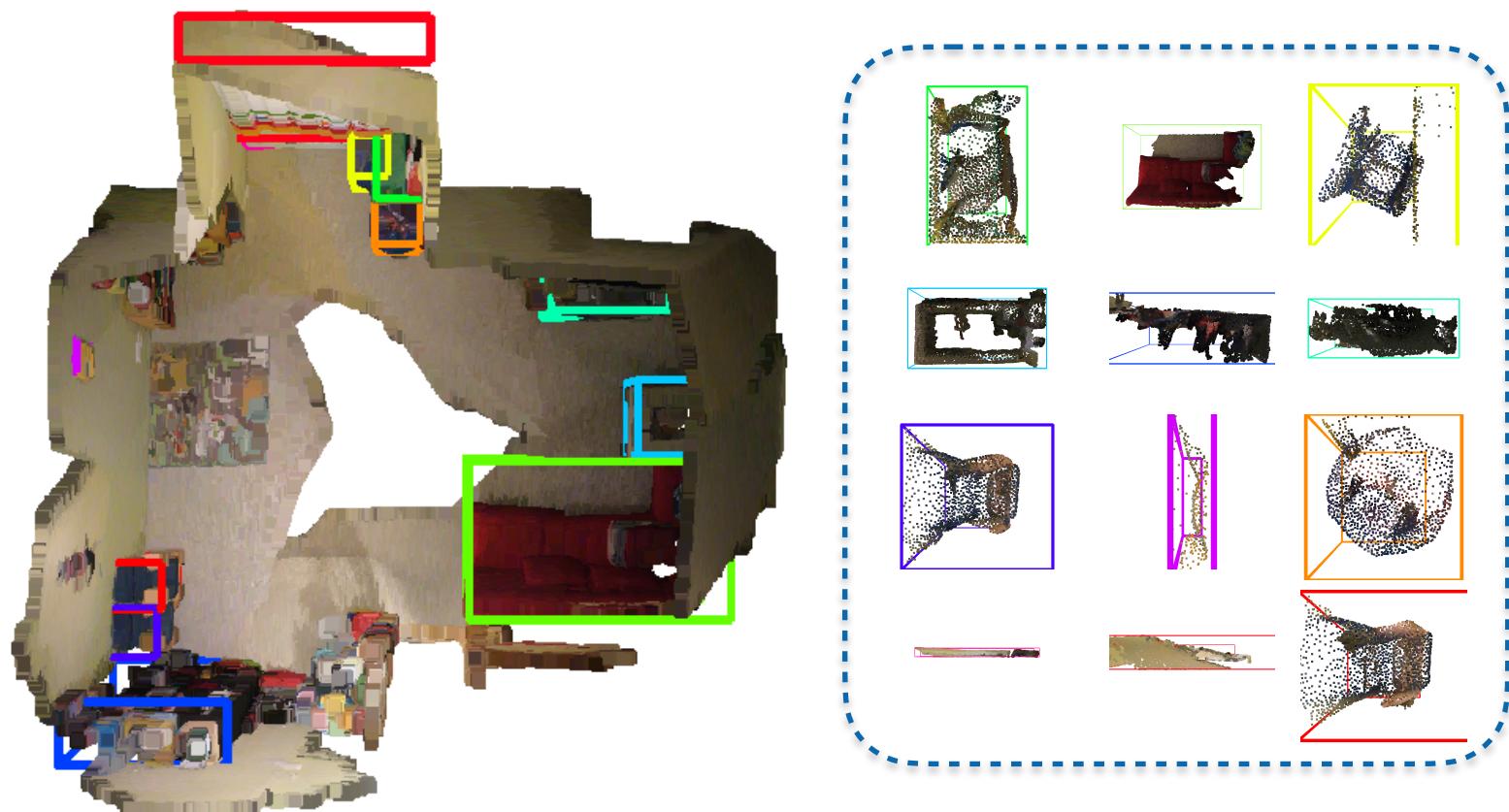
Instance label



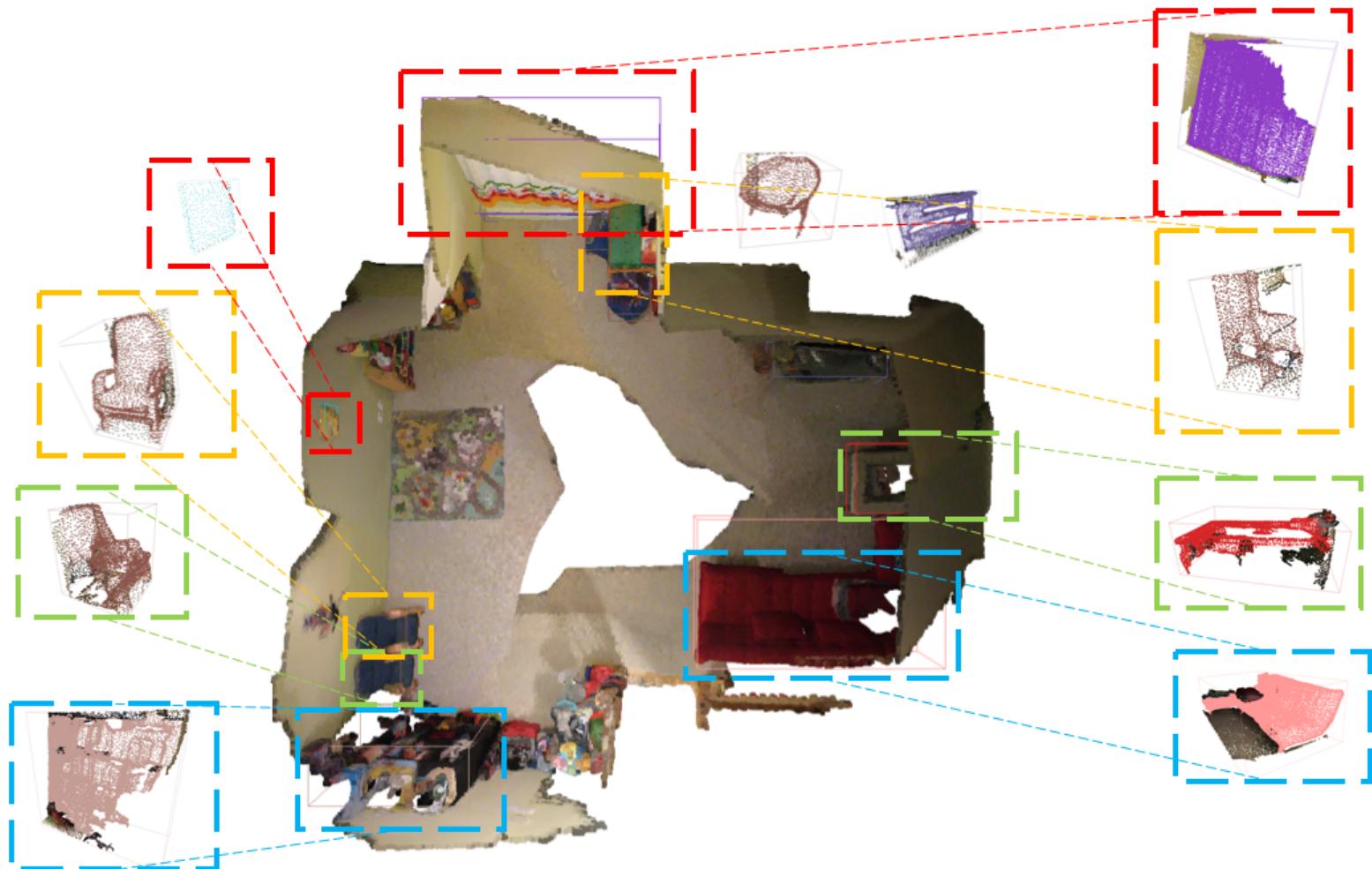
# **Top-down Approaches:**

## Proposal Generation & Point Association

# First Step: Generate Proposals (e.g., Bounding Boxes)



# Second Step: Associate Points with Proposals



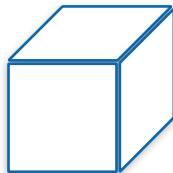
# Two Key Questions:

- How to generate (instance) proposals?
- How to associate points with proposals?

# **Details of Step 1: How to generate proposals?**

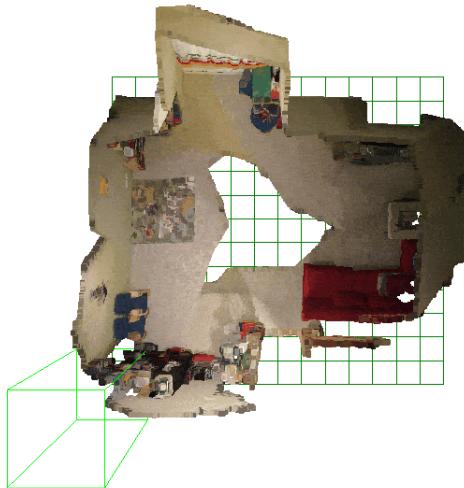
# First of all, what is a good proposal representation?

- First of all, what is the desired property of a proposal representation?
  - Easy to parameterize and predict
  - Easy to classify whether a point belongs to it
- Parameterization:
  - Primitive type
  - Parameters (position, rotation, ...)
- Common choices: 3D bounding box, spheres



# Proposal Generation: Non-Learning

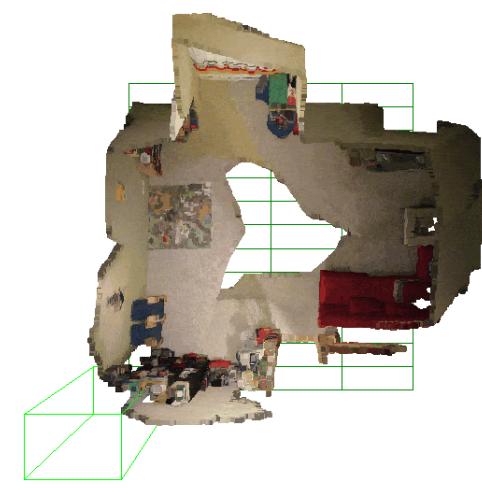
- **Sliding window:** The straightforward, heuristic method to generate proposals without learning
- Slide a (template) window over the input point cloud



stride=(0.5, 0.5)  
size=(1.5, 1.5)



stride=(1.5, 1.5)  
size=(1.5, 1.5)



stride=(1.5, 0.5)  
size=(1.5, 1.0)

# Proposal Generation: Learning-based

- To have a high recall, we need to densely slide a window
- However, too heavy burden for the association step

# Examples of Learning-based Proposal Generation

- 3D-BoNet
- GSPN

# Examples of Learning-based Proposal Generation

- 3D-BoNet
- GSPN

# 3D-BoNet Pipeline

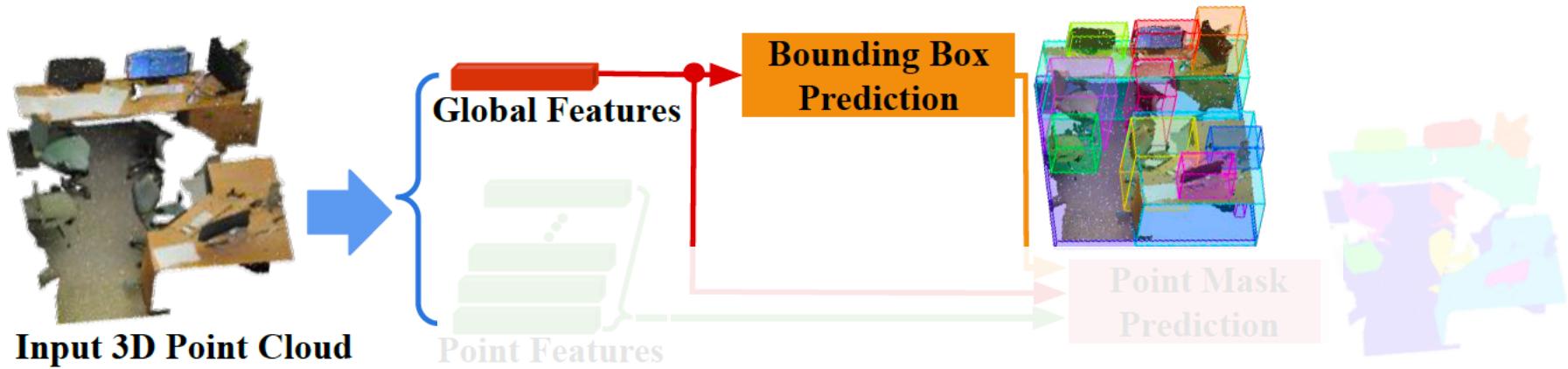
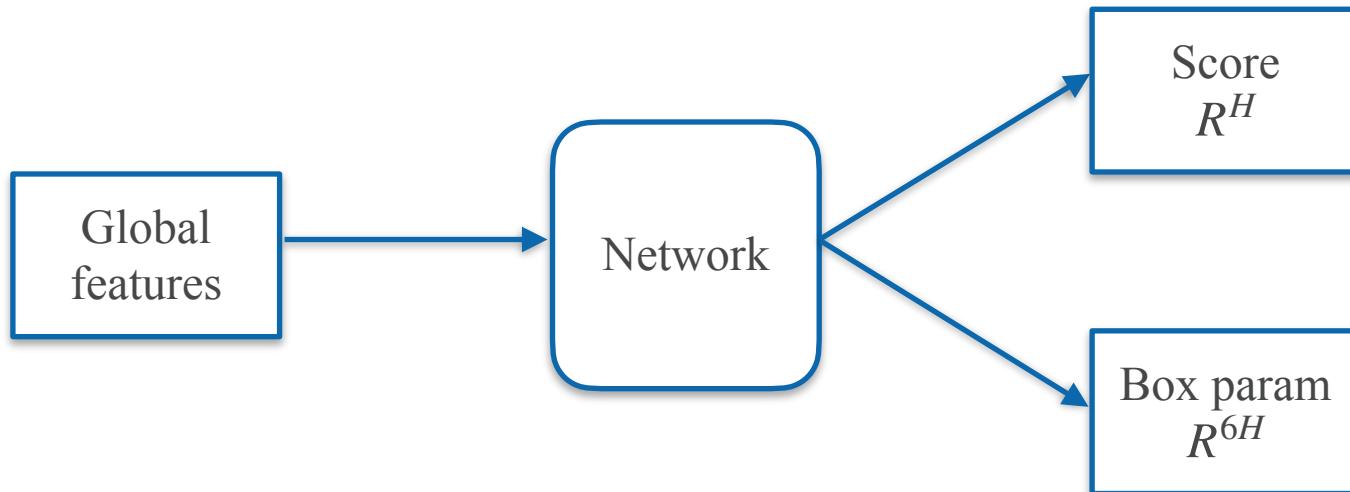


Figure 1: The 3D-BoNet framework for instance segmentation on 3D point clouds.

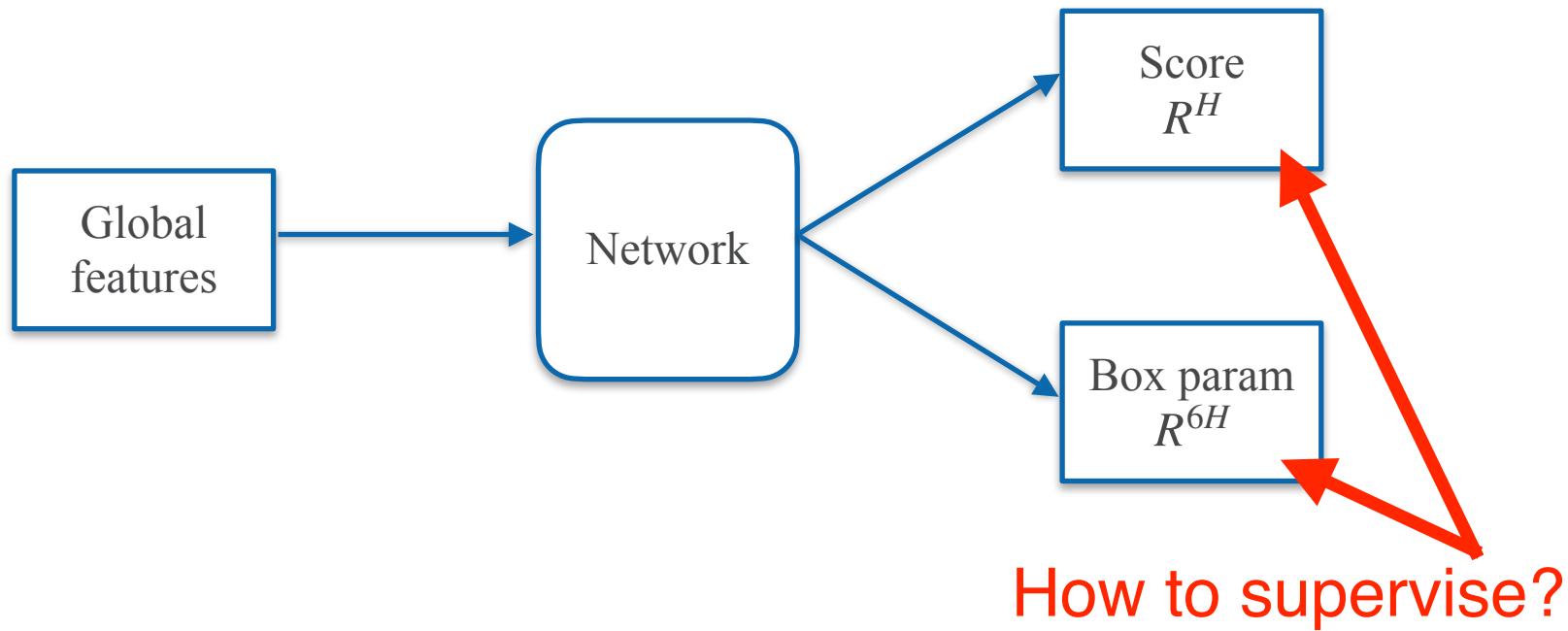
# Bounding Box Prediction

- Bounding box parameterization:  
 $\{x_{min}, y_{min}, z_{min}, x_{max}, y_{max}, z_{max}\}$
- Regress a predefined, fixed number ( $H$ ) of bounding boxes



# Bounding Box Prediction

- Bounding box parameterization:  
 $\{x_{min}, y_{min}, z_{min}, x_{max}, y_{max}, z_{max}\}$
- Regress a predefined, fixed number ( $H$ ) of bounding boxes

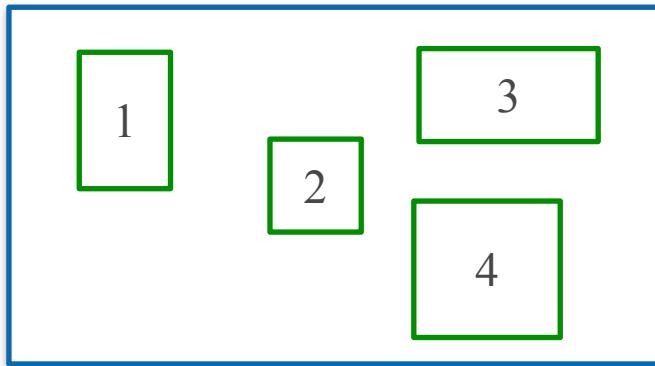


# Loss: Bounding Box Association

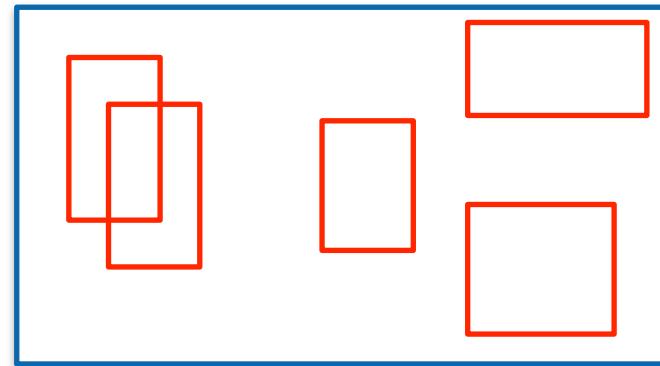
How to know the GT on-the-fly?

Find a match between the GT and predicted boxes

# Optimal Association (2D case)

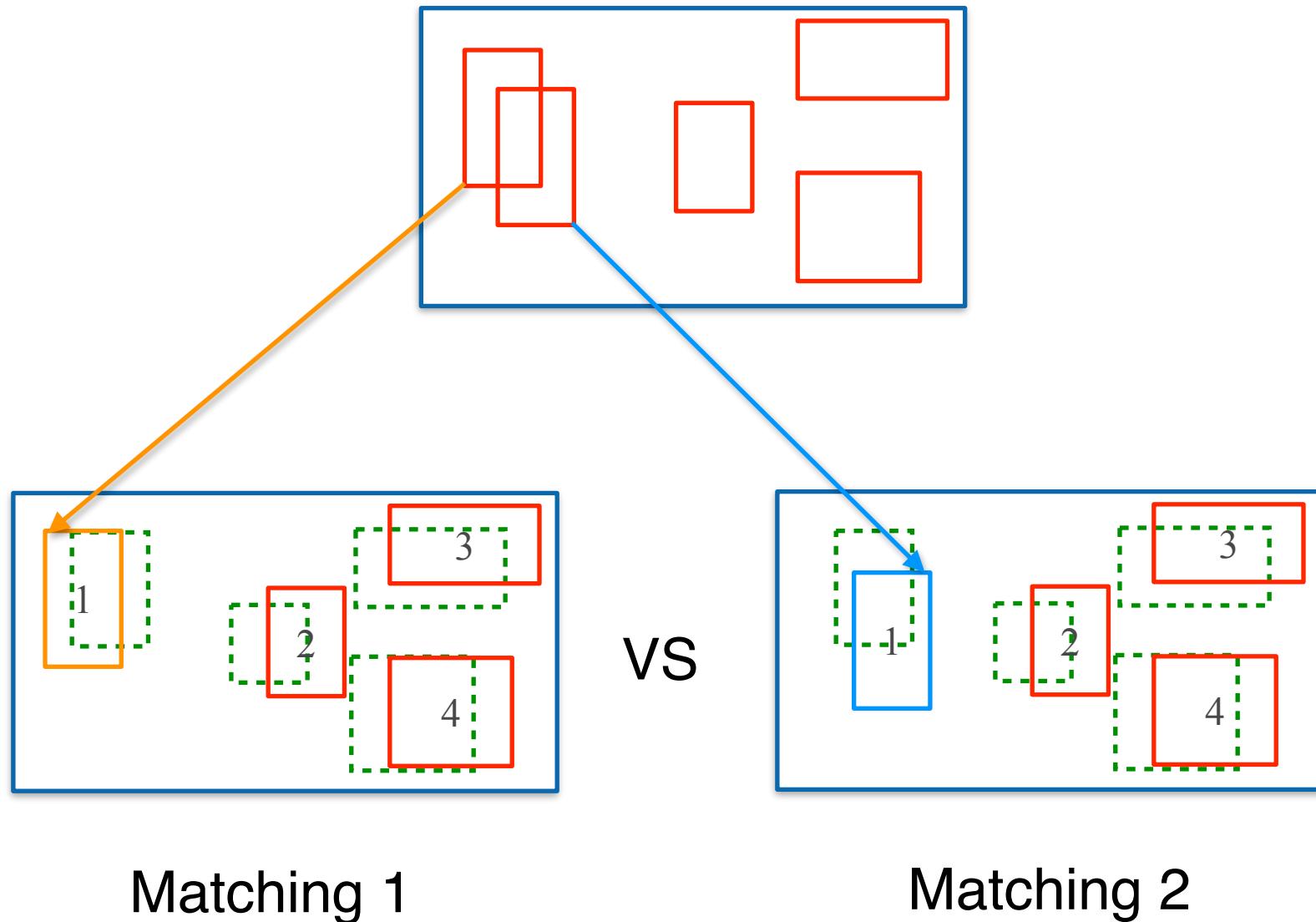


GT boxes



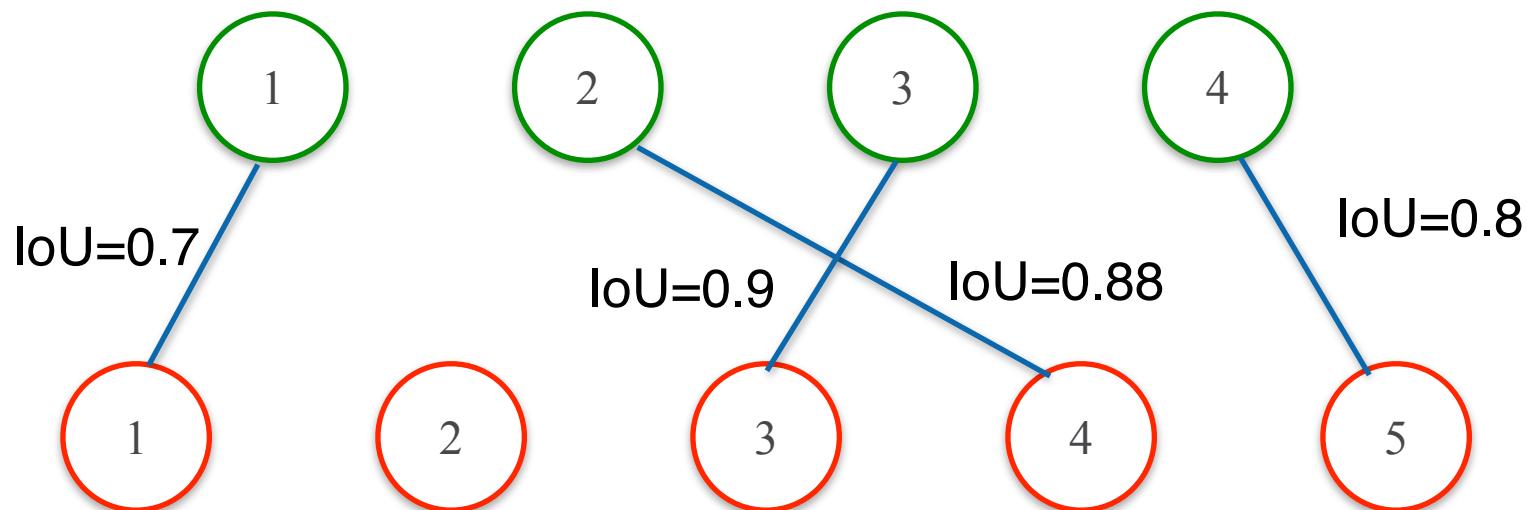
Prediction

# Optimal Association (2D case)



# Optimal Association

- Objective: minimize the overall match cost
- Hungarian algorithm can solve this problem (similar to EMD)



The overall cost is  $0.7 + 0.9 + 0.88 + 0.8$

# Association Cost

- The cost (weight of bipartite graph) should evaluate the similarity between the predicted box and GT box (e.g.,  $L_2$  over b.box vertices offset)

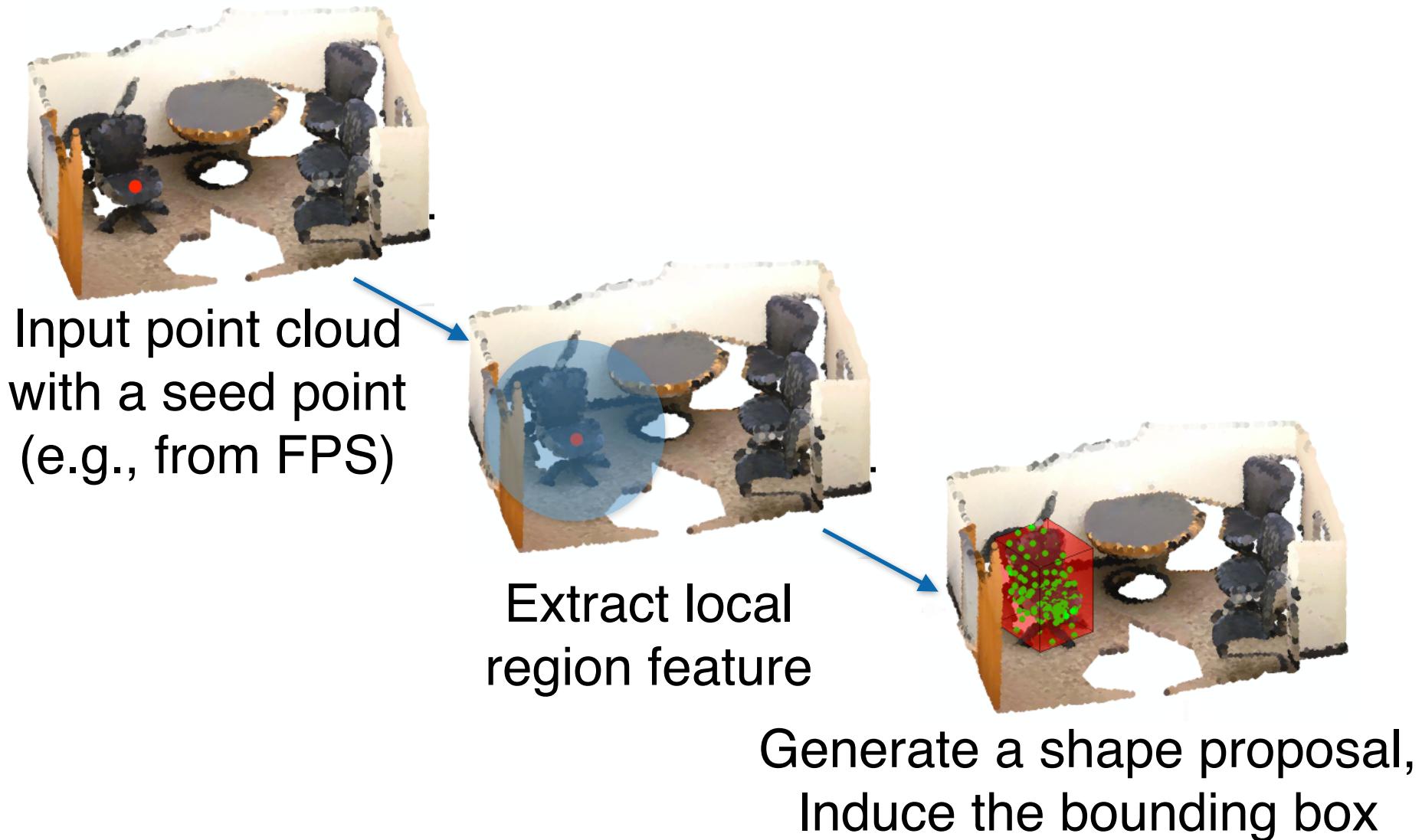
$$C_{i,j}^{ed} = \frac{1}{6} \sum (B_i - \bar{B}_j)^2$$

- Other criteria
  - Soft IoU
  - Cross-Entropy score
- The cost can be used as the loss directly

# Examples of Learning-based Proposal Generation

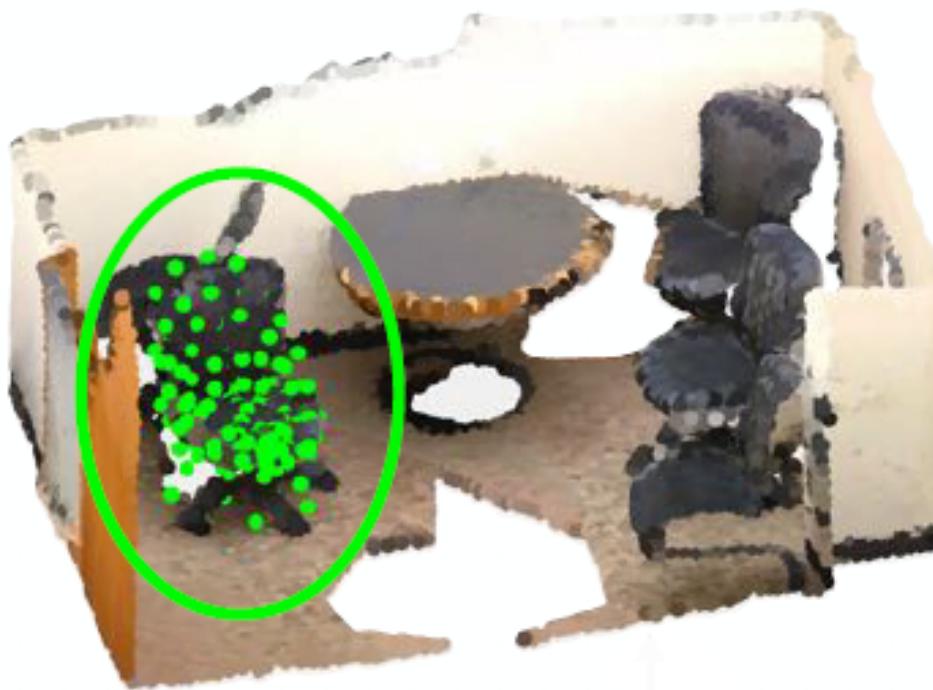
- 3D-BoNet
- GSPN

# GSPN Pipeline



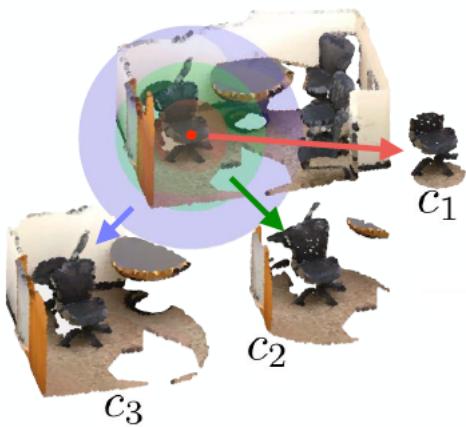
# Point Cloud as Object Proposal

- Unlike primitive-based proposals, it is possible to **generate a point cloud** as a proposal (recall the single image to point cloud work)



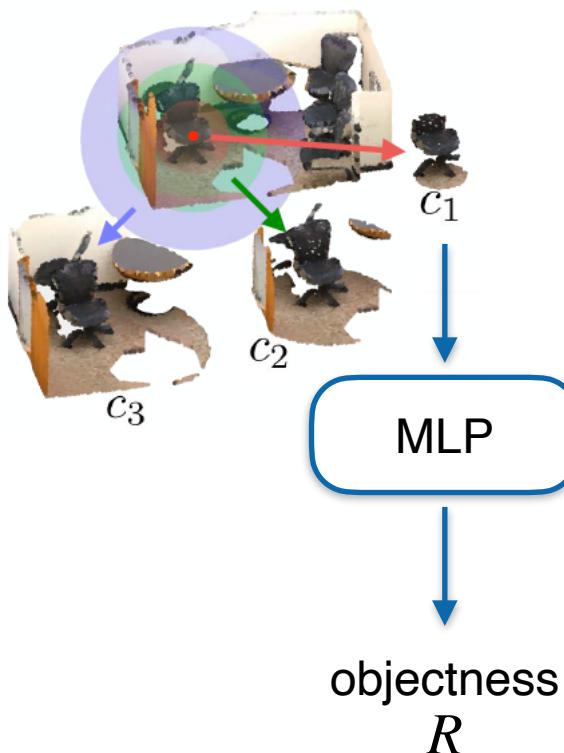
# Generate Proposal as a Point Cloud

- Take a seed point and local context of different scales



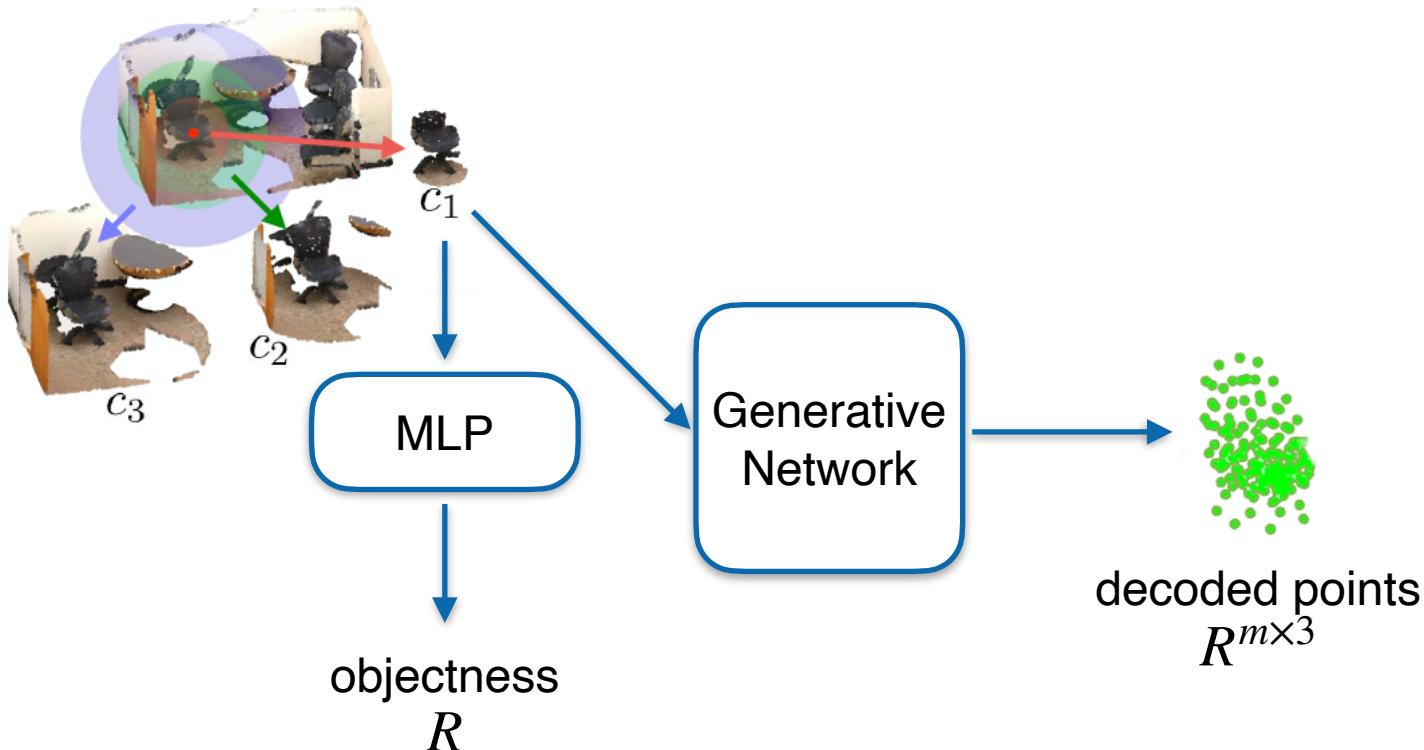
# Generate Proposal as a Point Cloud

- Predict “objectness” (object v.s. non-object)



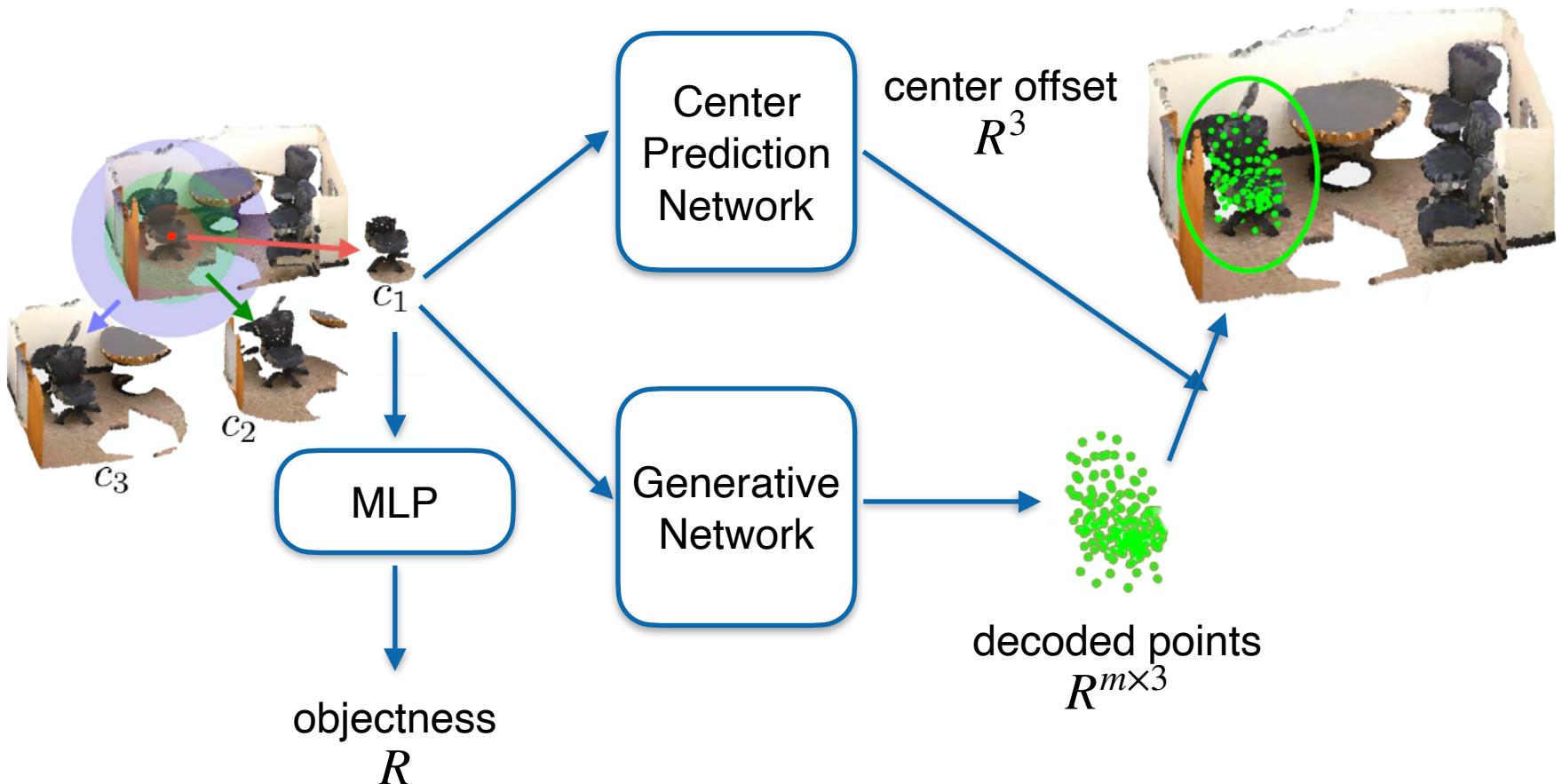
# Generate Proposal as a Point Cloud

- Decode points, e.g., by a fully-connected network, as in single-image to point cloud work



# Generate Proposal as a Point Cloud

- Predict a center offset from the seed point to the center of the instance



# Losses for Point Cloud Proposals

- Only for positive proposals
  - Center prediction loss: huber loss (smooth l1)
  - Shape generation loss: chamfer distance
- For all the proposals
  - Objectness loss: cross-entropy

# **How to associate points with proposals?**

# Basic Idea

- Given the proposal, predict a binary mask for each point whether the point belongs to the instance

# Example: 3D-BoNet

Extract per-point features  $\tilde{F}_l \in R^{N \times D}$

Get instance-aware features  $\hat{F}_l \in R^{N \times (D+7)}$ , e.g.,  
point features (D dim)  
bounding parameters (6 dim)  
confidence (1 dim)

Predict point-wise mask  $M_i \in \{0,1\}^N$

# Point Label Generation and Loss

- Given the matched proposal and GT
  - For each proposal, we can induce a per-point binary mask given its corresponding GT



overall instance label



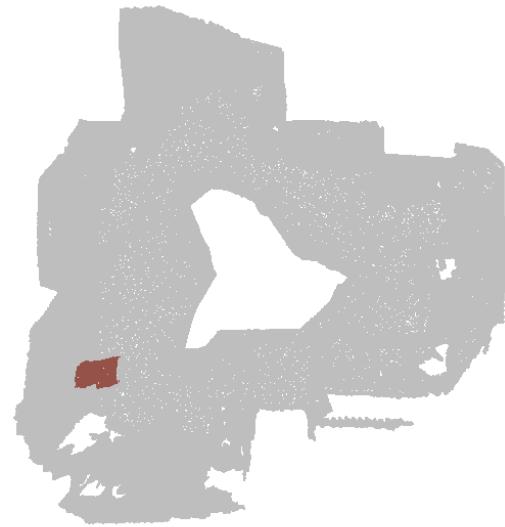
instance label for each proposal

# Point Label Generation and Loss

- Given the matched proposal and GT
  - For each proposal, we can induce a per-point binary mask given its corresponding GT
  - We use a cross-entropy loss to do per-point binary classification



overall instance label

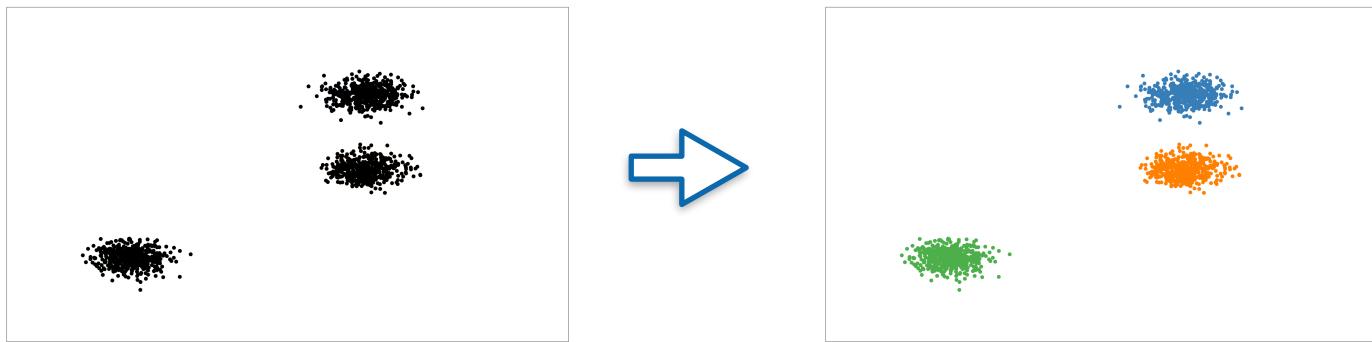


instance label for each proposal

# **Bottom-up Approaches**

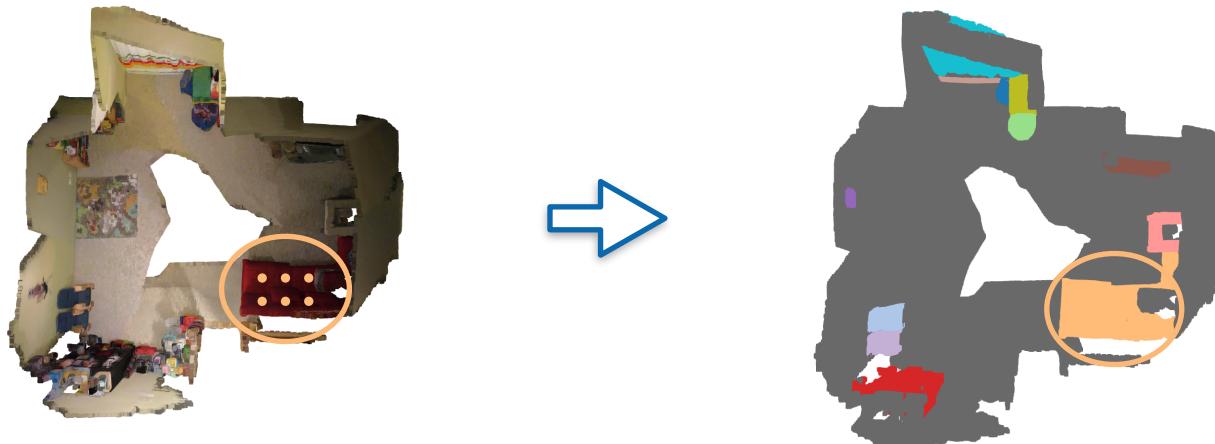
# What is Bottom-up?

- A bottom-up approach is grouping the pieces of the points together to form an object.



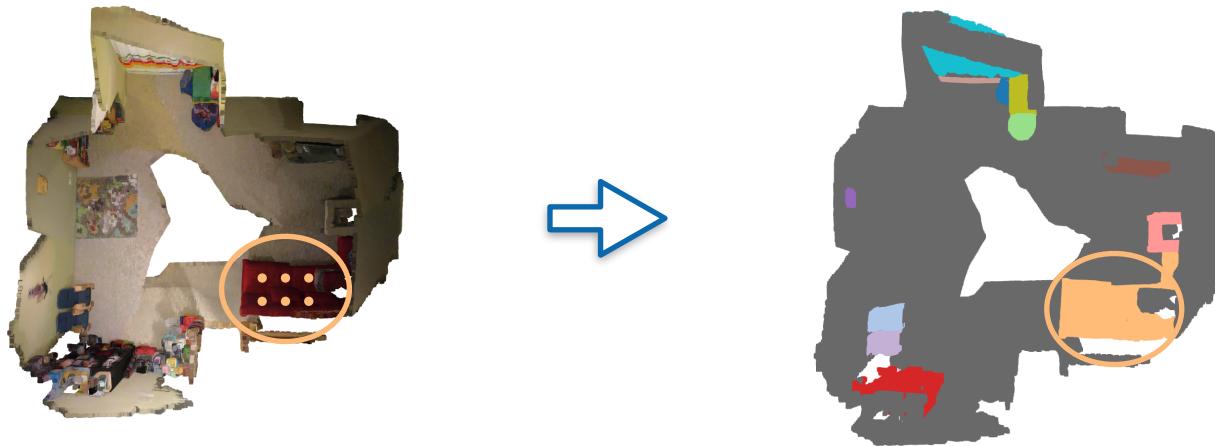
# What is Bottom-up?

- A bottom-up approach is grouping the pieces of the points together to form an object.



# What is Bottom-up?

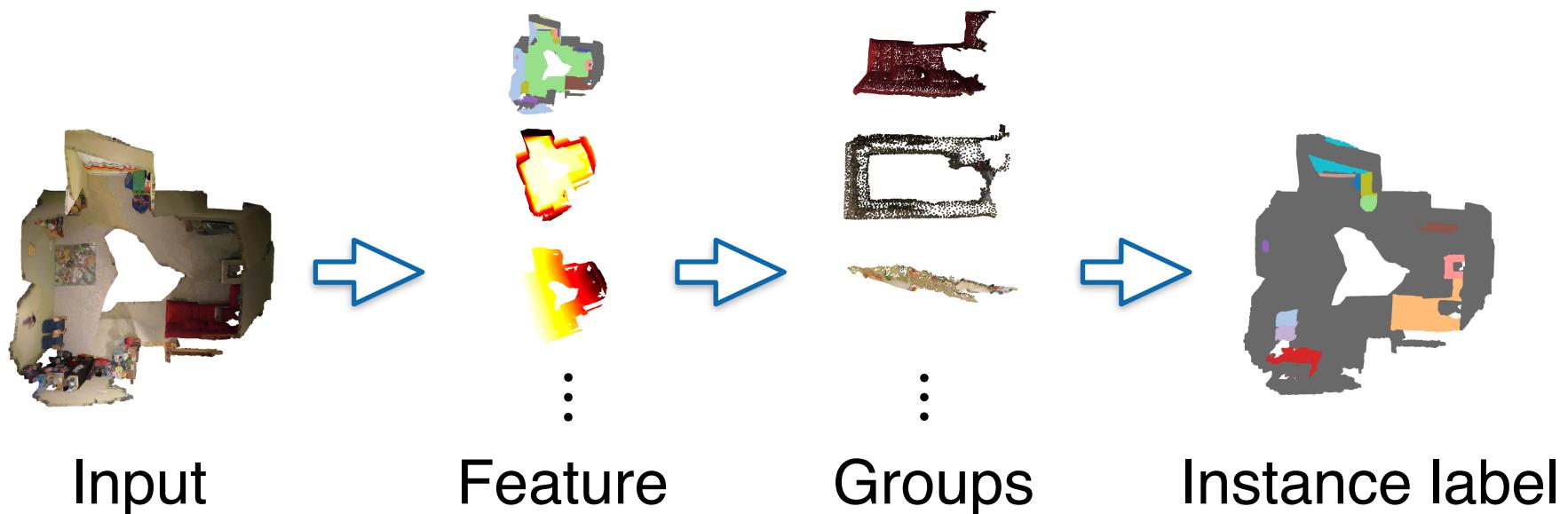
- A bottom-up approach is grouping the pieces of the points together to form an object.



- In contrast, top-down: directly predict a proposal as object proxy and verify

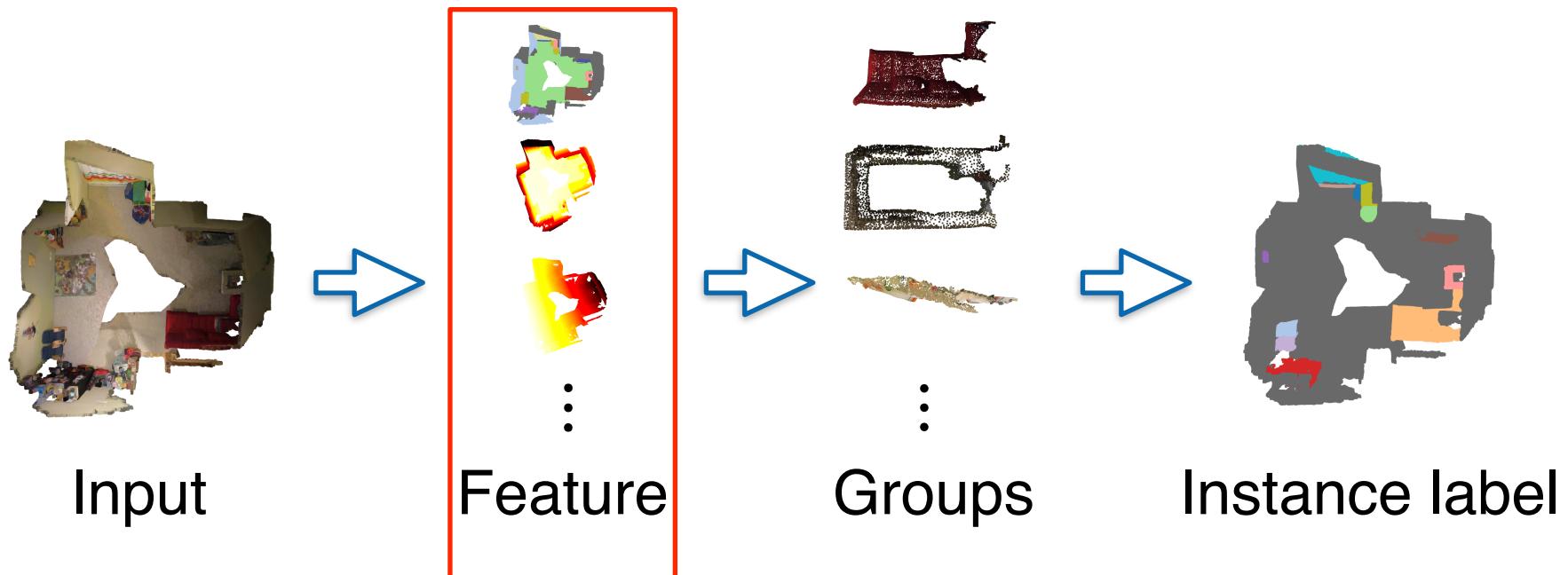
# Grouping-based Instance Segmentation

- Criteria for points/fragments to be grouped?
  - Distance function
- Group procedure
  - Grouping/Clustering algorithm
- Post-processing



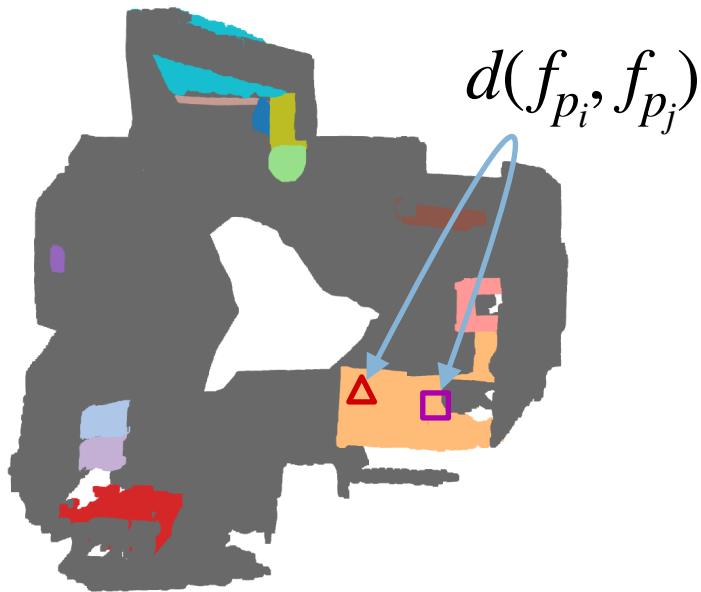
# Grouping-based Instance Segmentation

- Criteria for points/fragments to be grouped?
  - Distance function
- Group procedure
  - Grouping/Clustering algorithm
- Post-processing



# Key Ideas

- Points in the same instance should be close in the **feature** space, such that clustering can be applied.

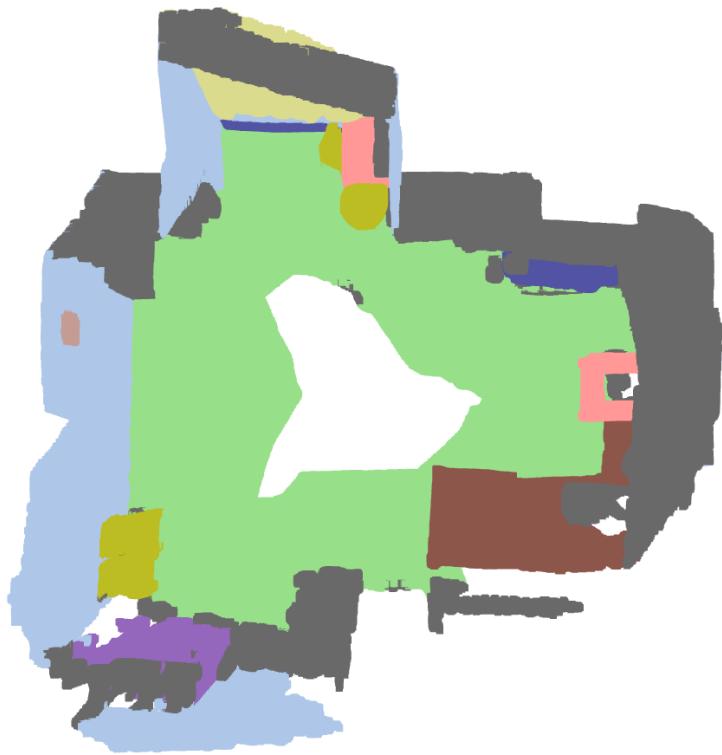


# Distance in Feature Space

- Common choice:  $L_p$ -distance
  - e.g.,  $L_1$ -distance:  $\|F_i - F_j\|_1$
- Potential features to consider:
  - Semantic features (about semantic label)
  - Spatial feature (about point location)
  - Instance feature (to distinguish instances)

# Candidate I: Semantic Feature

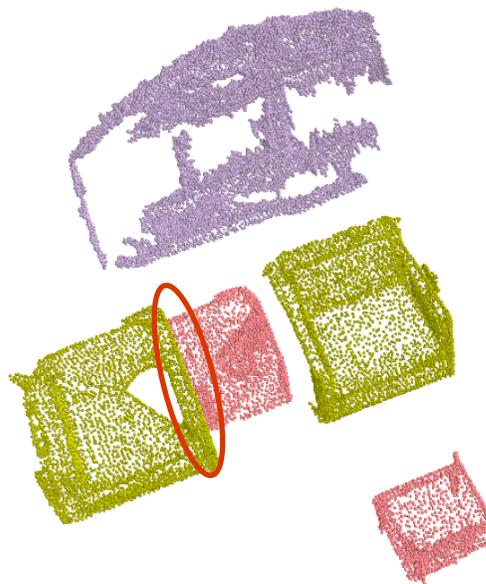
- Learn semantic feature for each point by point cloud segmentation loss.



MLAJiang, Li, et al. "Pointgroup: Dual-set point grouping for 3d instance segmentation." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020.

# Candidate II: Spatial Feature

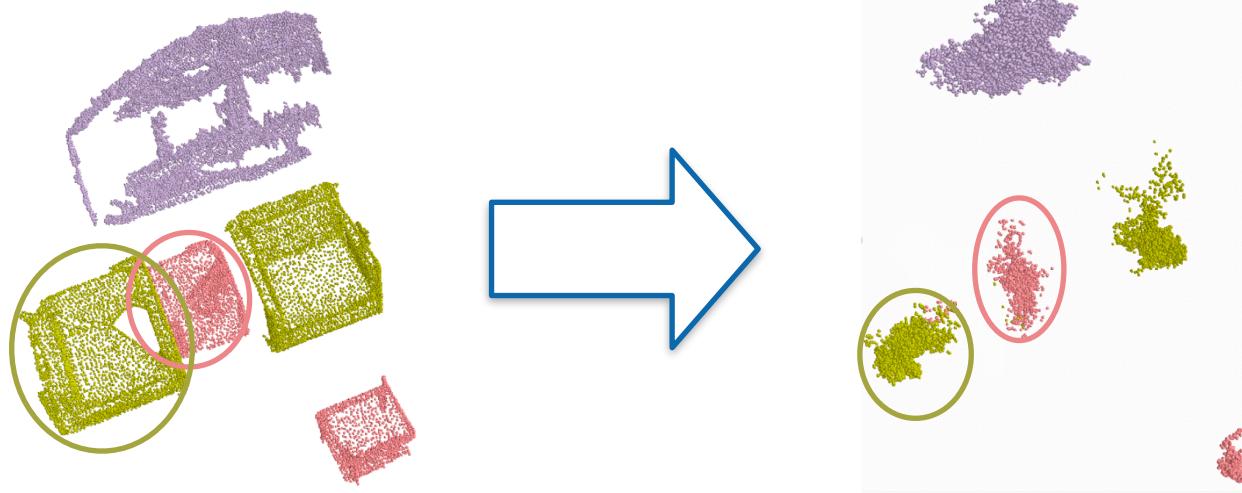
- Use 3D coordinates of points?
  - Reasonable, however,
  - Fails for points around object boundaries



MLAJiang, Li, et al. "Pointgroup: Dual-set point grouping for 3d instance segmentation." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020.

# Candidate II: Spatial Feature

- Learn to predict object center coordinates, and use the predicted object center as the spatial feature



Predicted Object Centers

MLA Jiang, Li, et al. "Pointgroup: Dual-set point grouping for 3d instance segmentation." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020.

# Candidate III: Instance Features

- Fundamentally, we hope that the feature can be powerful enough to distinguish different instances
- Why not directly design a loss to learn it?!

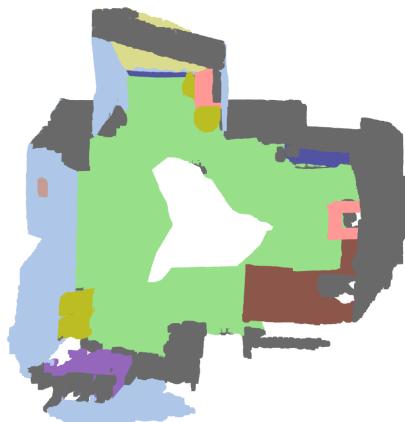


Color map of distances between the given point and rest points (darker means closer)

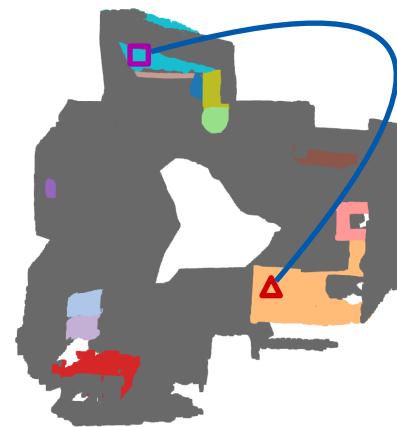
Wang, Weiyue, et al. "Sgpn: Similarity group proposal network for 3d point cloud instance segmentation." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

# Contrastive Loss

- Use double-hinge loss for each **pair of points** to train point features.



Semantic label

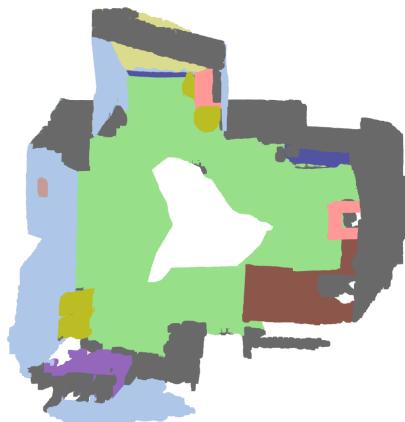


Instance label

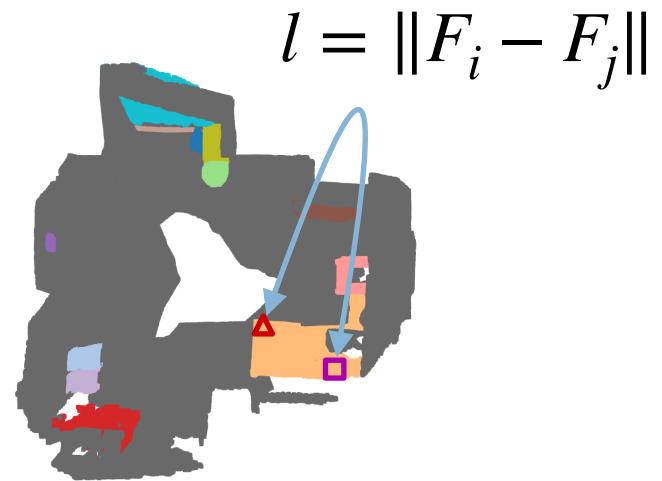
Wang, Weiyue, et al. "Sgpn: Similarity group proposal network for 3d point cloud instance segmentation." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

# Same Instance Case

- Point  $i$  and point  $j$  belongs to in the same instance.



Semantic label



Instance label

Wang, Weiyue, et al. "Sgpn: Similarity group proposal network for 3d point cloud instance segmentation." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

# Same Instance Case

- Point  $i$  and point  $j$  belongs to different instances with the same semantic label.

$$l(i, j) = \alpha \max(0, K_1 - \|F_i - F_j\|)$$

“If the feature distance is below  $K_1$ , it is penalized”



Semantic label

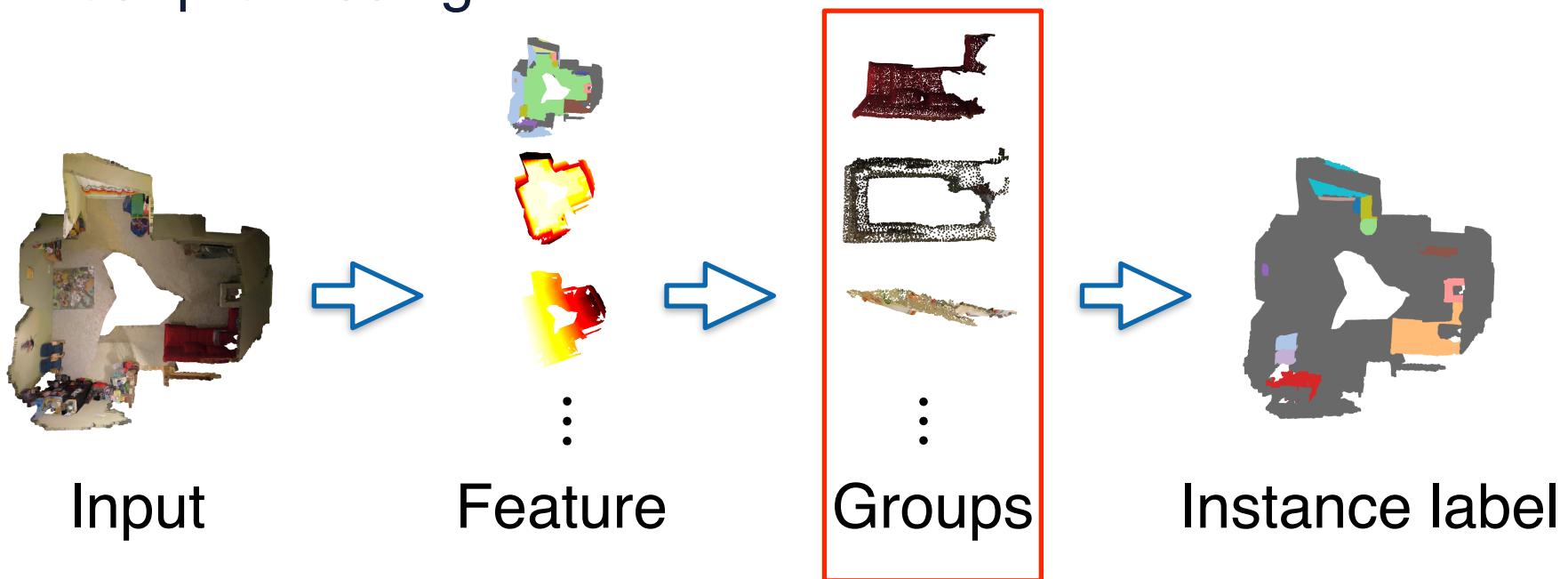


Instance label

Wang, Weiyue, et al. "Sgpn: Similarity group proposal network for 3d point cloud instance segmentation." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

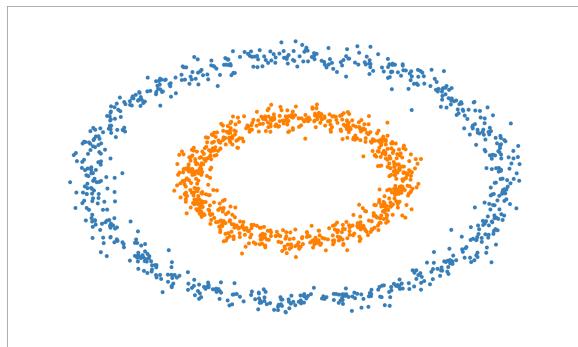
# Grouping-based Instance Segmentation

- Criteria for points/fragments to be grouped?
  - Distance function
- Group procedure
  - Grouping/Clustering algorithm
- Post-processing

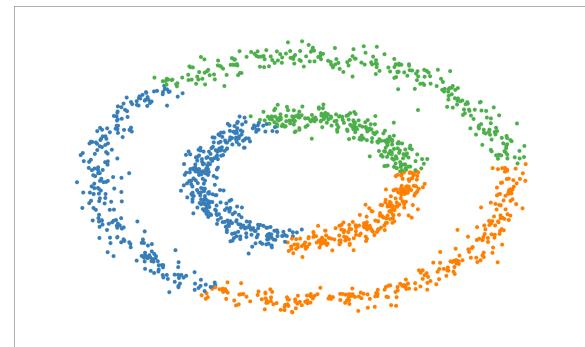


# Grouping by Clustering Point Features

- Choose your favorable clustering algorithm
  - DBSCAN
  - Mean shift
  - ...



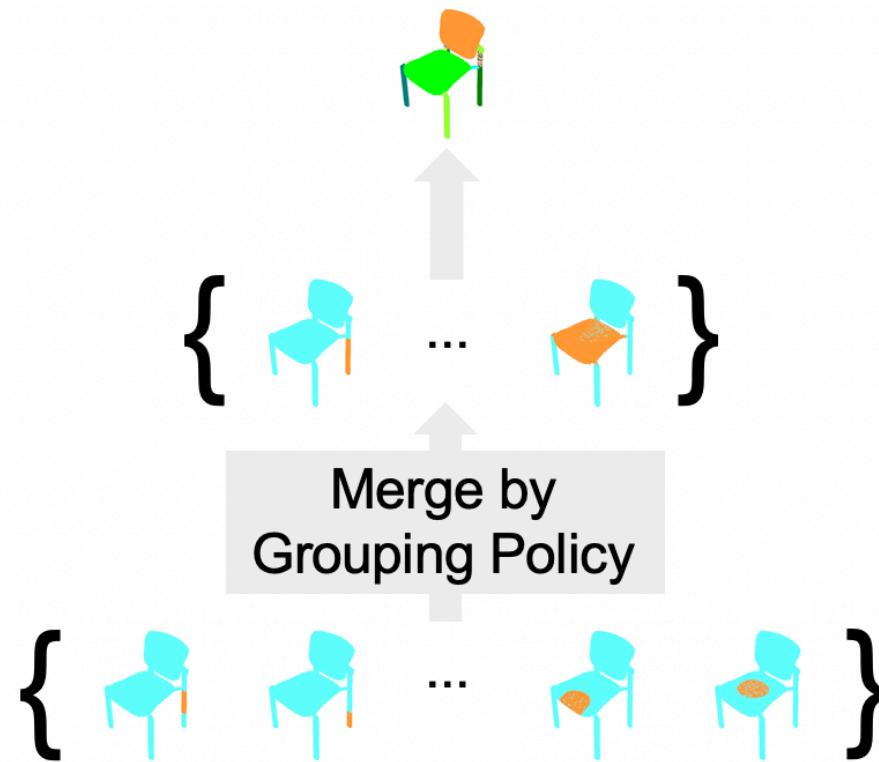
DBSCAN



Mean shift

# Learning to Cluster?

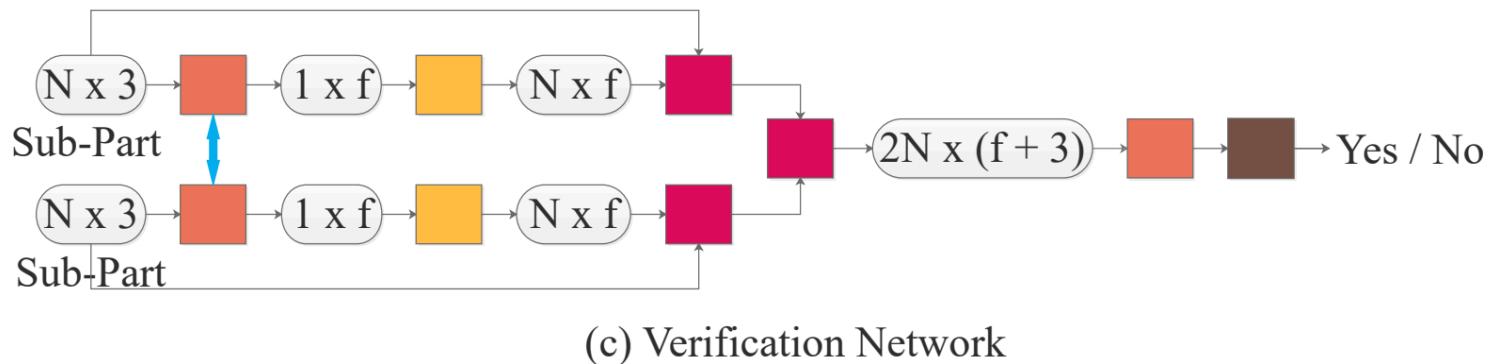
- E.g., modify **classical** agglomerative clustering to **learning-based** agglomerative clustering



Luo T, Mo K, Huang Z, et al. Learning to group: a bottom-up framework for 3d part discovery in unseen categories[J]. arXiv preprint arXiv:2002.06478, 2020.

# Learning to Group

- Assuming the instance consists of some parts.
- Core idea: use a neural network to predict if two parts should be merged into one instance.



# Post-Processing by IoU Score Prediction

- We can use a network to predict a score which can represent IoU between prediction and ground truth.
- Remove instances with low score.

