Paper Review

SoftGroup for 3D Instance Segmentation on Point Clouds

CVPR, 2022

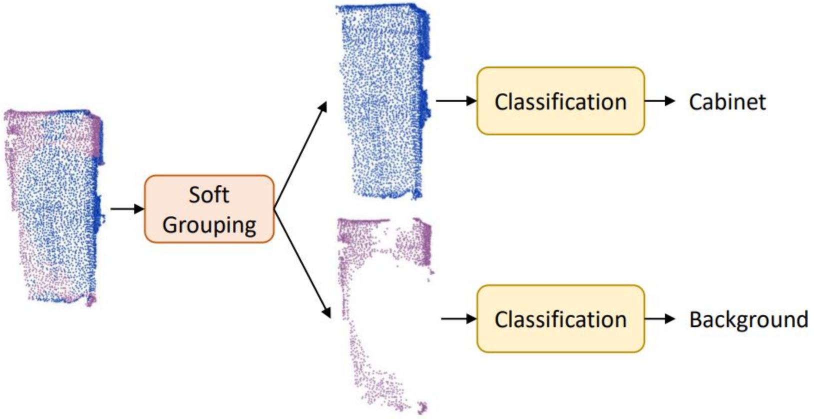


Figure 2. The `cabinet` in Figure 1 is extracted to illustrate the high-level pipeline of our method. The soft grouping module based on soft semantic scores to output more accurate instance (the upper one). The classifier processes each instance and suppress the instance from wrong semantic prediction (the lower one).

Related Work

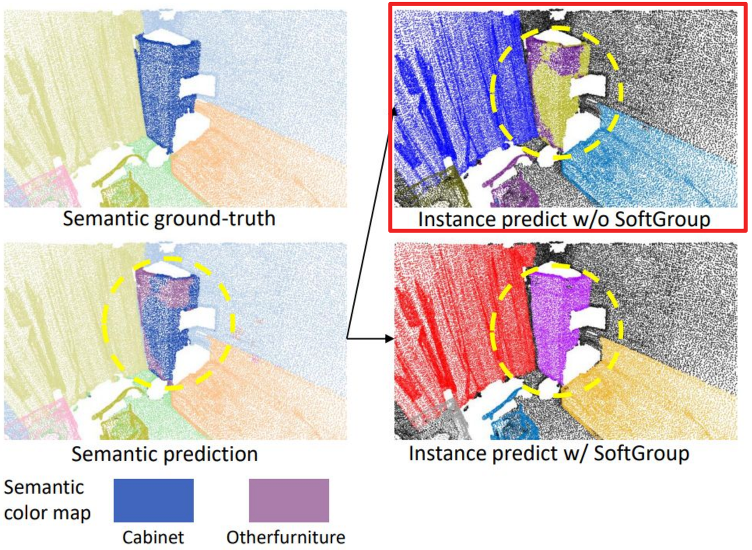
1. Bottom-up pipeline = Grouping method

   : learn the point-wise semantic labels and then group points of the same labels with small geometric distances into instances

- Problems of Grouping algorithms

  : using hard semantic predictions

  - Low overlap between predicted instance and the ground-truth
  - Extra false-positive instances from wrong semantic regions



The semantic prediction error is propagated to instance prediction. As a result, the predicted *cabinet* instance has low overlap with the ground truth, and the *other furniture* instance is a false positive

Related Work

2. Top-down pipeline = Proposal-based method

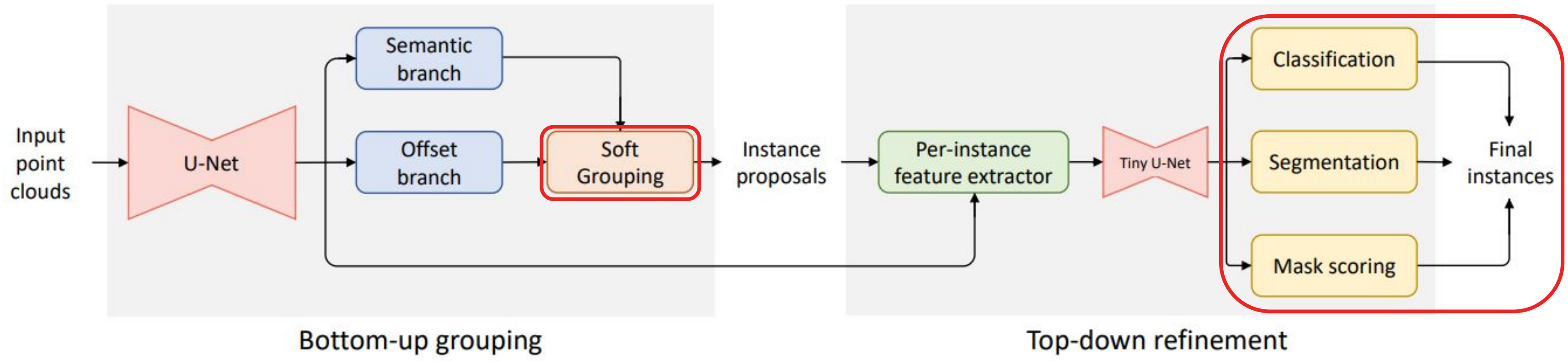
   : Generates region proposals and then segments the object within each proposal

	Top-down	Bottom-up
Pros	process each object proposal independently	process the whole scene without proposal generation, enabling fast inference
Cons	difficulties in generating high-quality proposals since the point only exist on the object surface	highly depend on semantic segmentation

Idea

: Use soft semantic scores to perform grouping instead of hard one-hot semantic predictions

Architecture

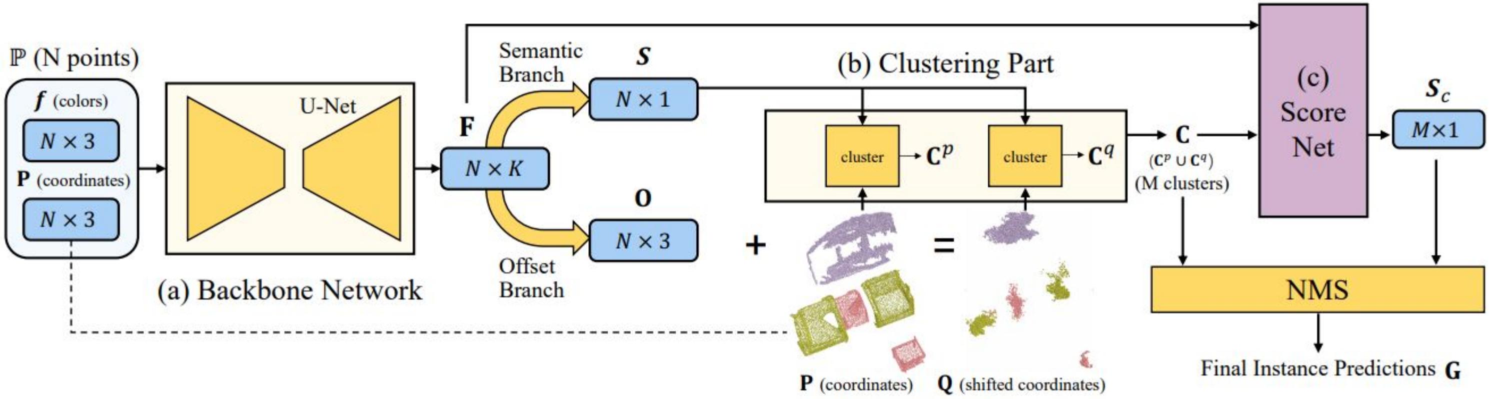


1. Point-wise prediction network : produce point-wise semantic labels and offset vector
2. Soft grouping module : produce preliminary instance proposals
3. Top-down refinement stage : based on the proposals, predict classes, instances mask, and mask scores as the final result

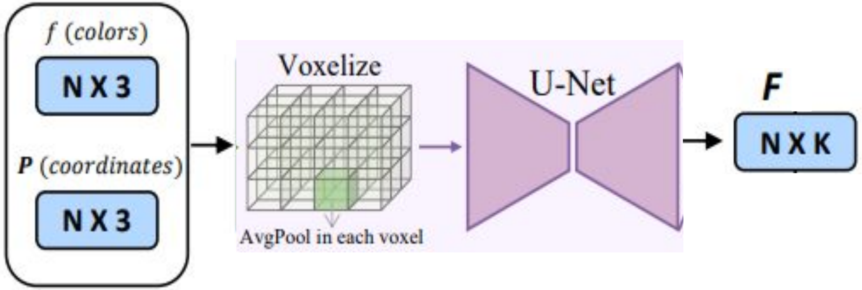
Paper Review

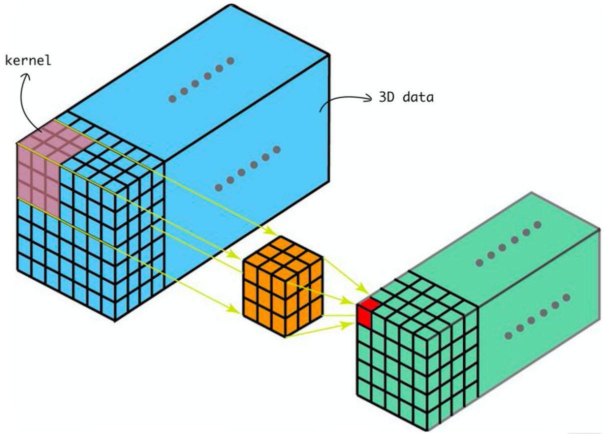
PointGroup : Dual-Set Point Grouping for 3D Instance Segmentation

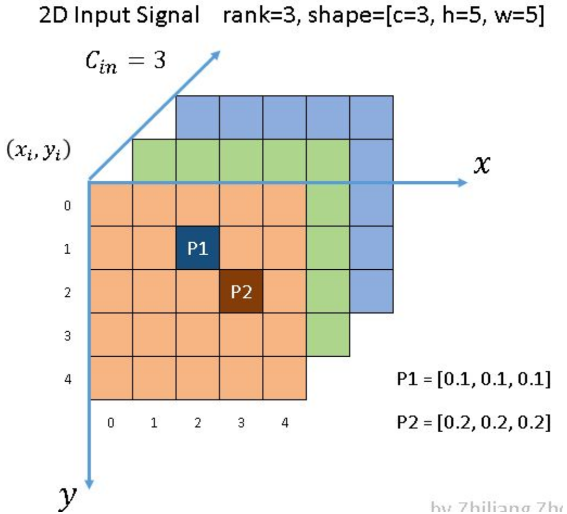
CVPR 2020



U-Net







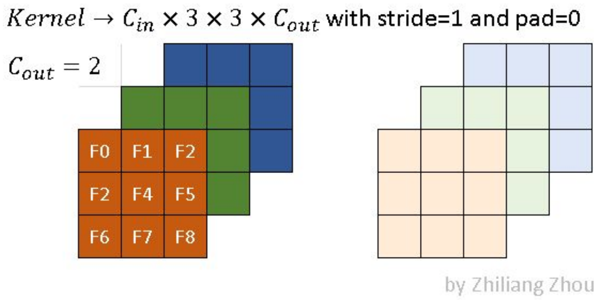
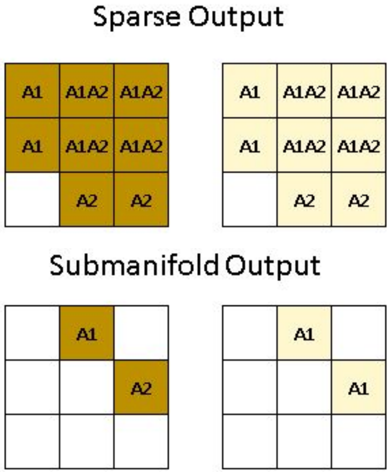
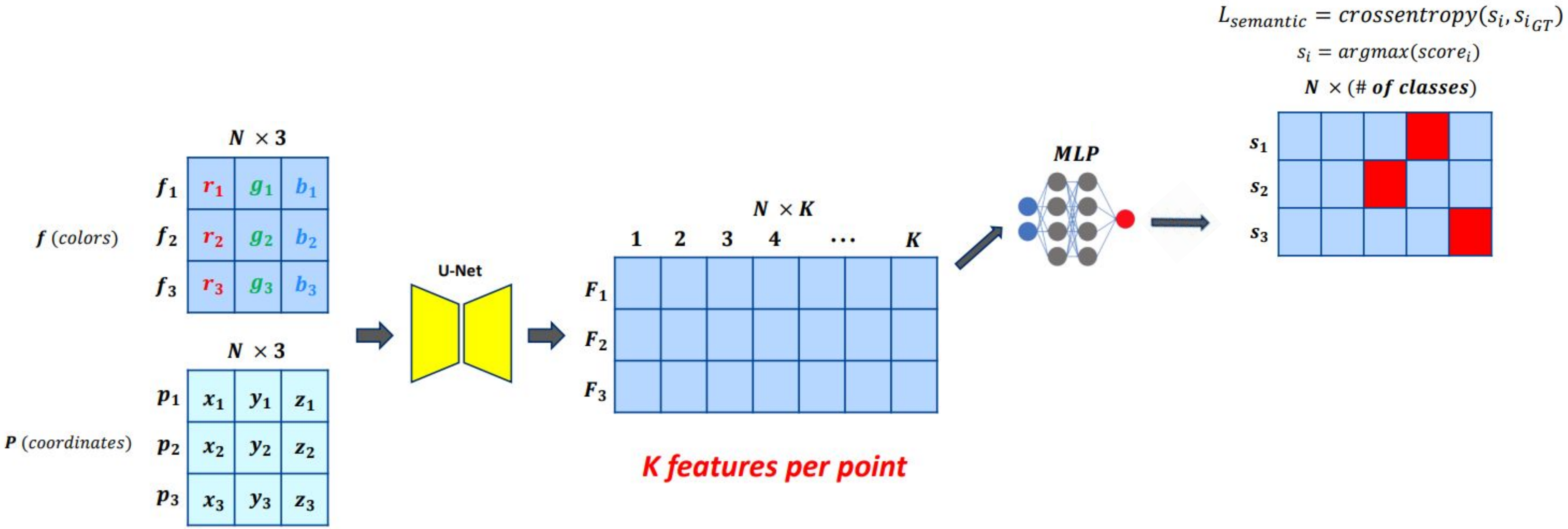


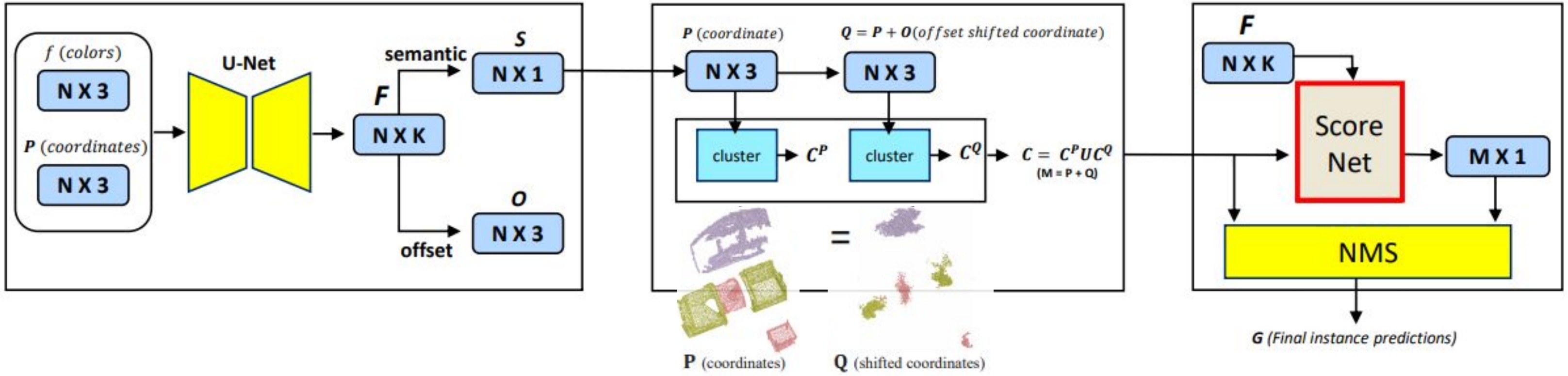
Figure 6: Sparse convolution kernel



Segmentation Branch

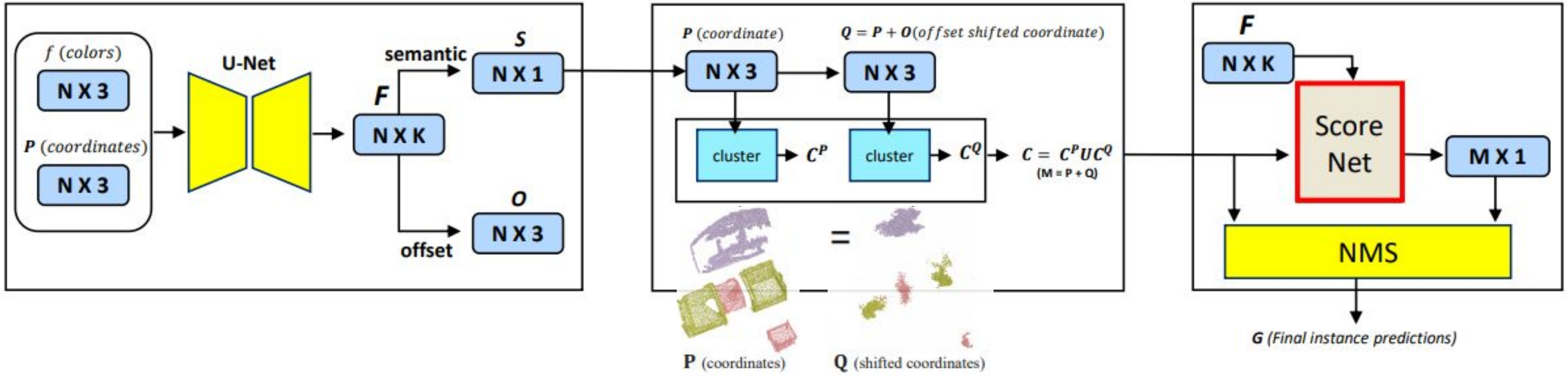


Clustering



- Grouping points directly based on the point coordinate set P may fail to separate same category objects that are close to each other

Offset Branch



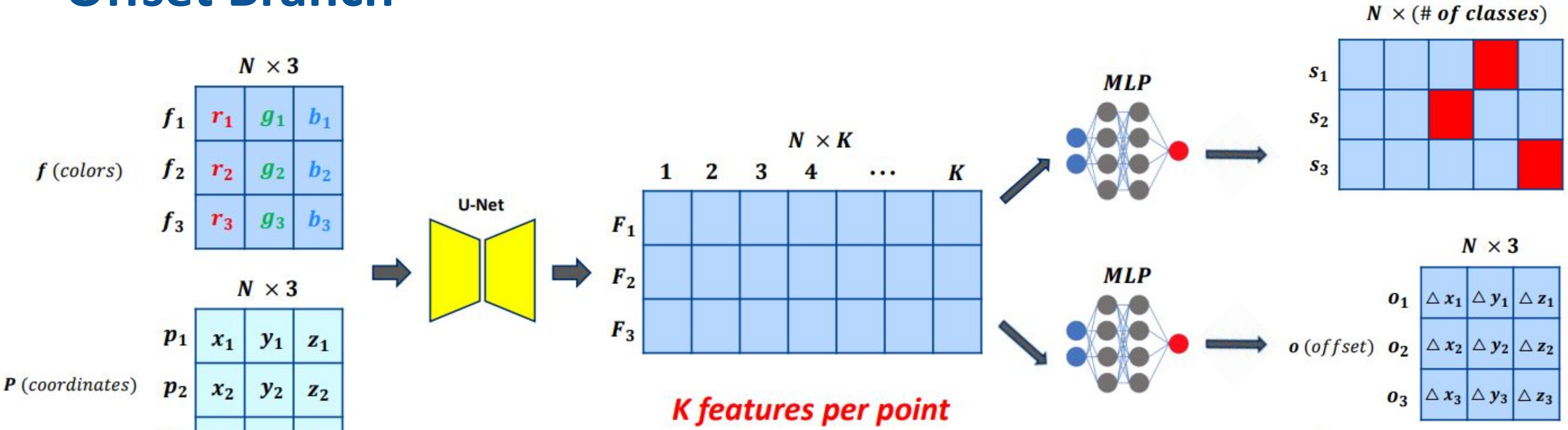
: Cluster based on shifted coordinate set Q, separate nearby objects better, even though they have the same semantic labels

- Offset : shift point towards its respective instance centroid
- However, for points near object boundary, the predicted offsets may not be accurate.
  - So employs “dual” point coordinate

Offset Branch

$$L_{semantic} = crossentropy(s_i, s_{i_{GT}})$$

$$s_i = argmax(score_i)$$



$$c_i = \frac{1}{N^I_{g(i)}} \sum_{j \in I_{g(i)}} p_j$$

Boundary points of **large size object** are hard to regress offset, since these points are relatively far from the instance centroids.

👉 **Direction loss** : constrain the direction of predicted offset vectors

$$\frac{o_i}{\|o_i\|_2} : e_{pred} \qquad \frac{c_i - p_i}{\|c_i - p_i\|_2} : e_{GT}$$

$$e_{pred} \cdot e_{GT} = \begin{cases} 1 \; if \; directions \; are \; same \\ 0 \; if \; directions \; are \; opposite \end{cases}$$

offset centroid 1 if instance

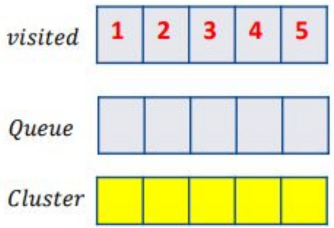
$$L_{o_{REG}} = \sum \|o_i - (c_i - p_i)\| \cdot m_i$$

coordinate 0 if not instanc

$$L_{o_{dir}} = -\sum \frac{o_i}{\|o_i\|_2} \cdot \frac{c_i - p_i}{\|c_i - p_i\|_2} \cdot m_i$$

Clustering

: based on the void space between objects.



Breadth-First Search(BFS)

Get points **within the ball of radius r**

Group points with **the same semantic labels**

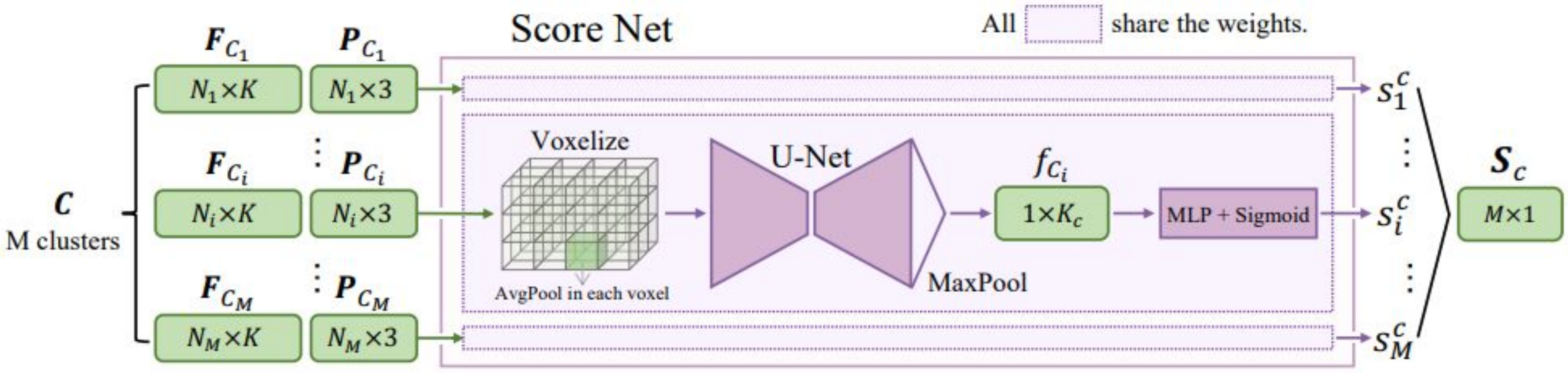
Algorithm 1 Clustering algorithm. N is the number of points. M is the number of clusters found by the algorithm.

Input: clustering radius r;
cluster point number threshold N_θ;
coordinates $\mathbf{X} = \{x_1, x_2, ..., x_N\} \in \mathbb{R}^{N\times3}$; and
semantic labels $\mathbf{S} = \{s_1, ..., s_N\} \in \mathbb{R}^N$.

Output: clusters $\mathbf{C} = \{C_1, ..., C_M\}$.

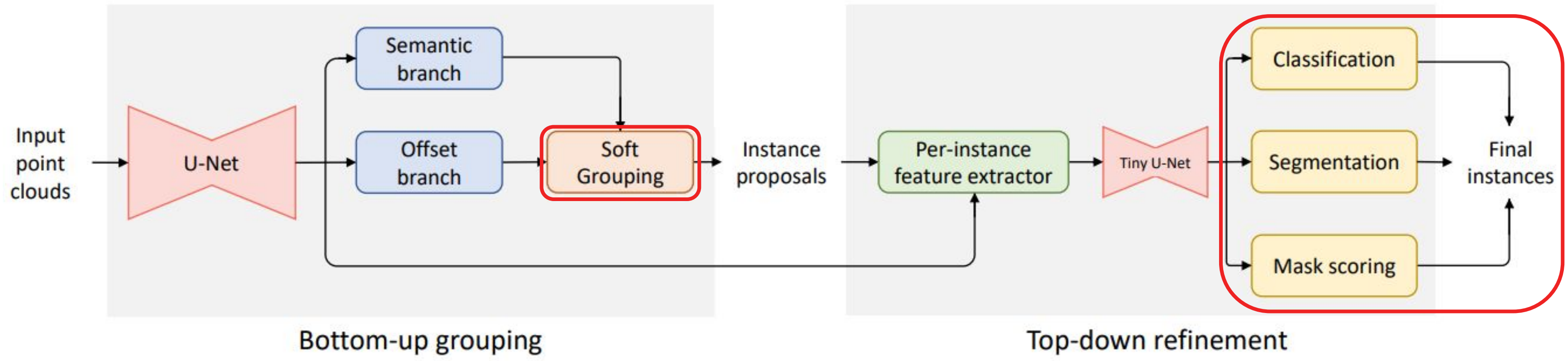
initialize an array v (visited) of length N with all zeros
initialize an empty cluster set $\mathbf{C}$
for $i = 1$ to N **do**
  if s_i is a stuff class (*e.g.*, wall) **then**
    $v_i = 1$
for $i = 1$ to N **do**
  if $v_i == 0$ **then**
    initialize an empty queue Q
    initialize an empty cluster C
    $v_i = 1$; Q.enqueue(i); add i to C
    while Q is not empty **do**
      $k = Q$.dequeue()
      for $j \in [1, N]$ with $||x_j - x_k||_2 < r$ **do**
        if $s_j == s_k$ and $v_j == 0$ **then**
          $v_j = 1$; Q.enqueue(j); add j to C
    if number of points in $C > N_\theta$ **then**
      add C to $\mathbf{C}$
return C

Score Net



1. Predict a score for each cluster to indicate the quality of the associated cluster proposal
2. Reserve the better clusters in NMS

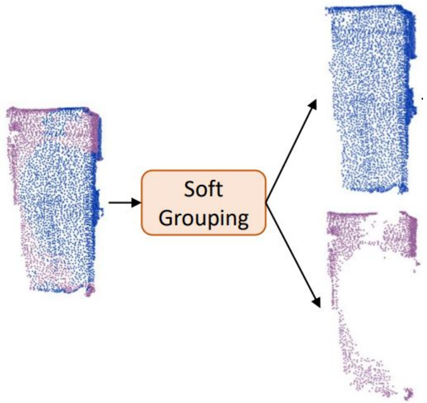
Architecture



1. Point-wise prediction network : produce point-wise semantic labels and offset vector
2. Soft grouping module : produce preliminary instance proposals
3. Top-down refinement stage : based on the proposals, predict classes, instances mask, and mask scores as the final result

Soft Grouping

- Score threshold τ : determine which semantic classes a point belongs to, allowing the multiple classes



for i = 0 **to** N_{class}

for j = 0 **to** #(N_{class} points)

Create the links between points having a geometric distance smaller than r to get the instance proposals.

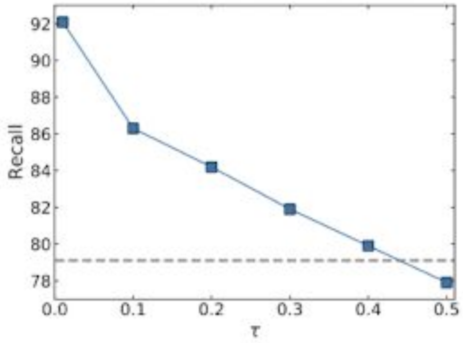
: For each iteration, the grouping is performed on a point subset of the whole scan

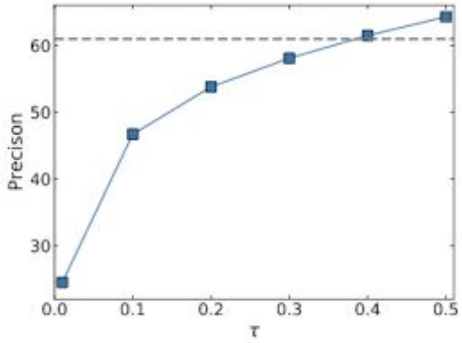
→ Ensure fast inference

- Relies on point-level proposals which are inherit the scattered property of point clouds.

Soft Grouping

- Score threshold τ : determine which semantic classes a point belongs to, allowing the multiple classes





$$\text{recall}_j = \sum_{i=1}^{N} \frac{(s_{ij} > \tau) \wedge (s_i^* = j)}{s_i^* = j}$$

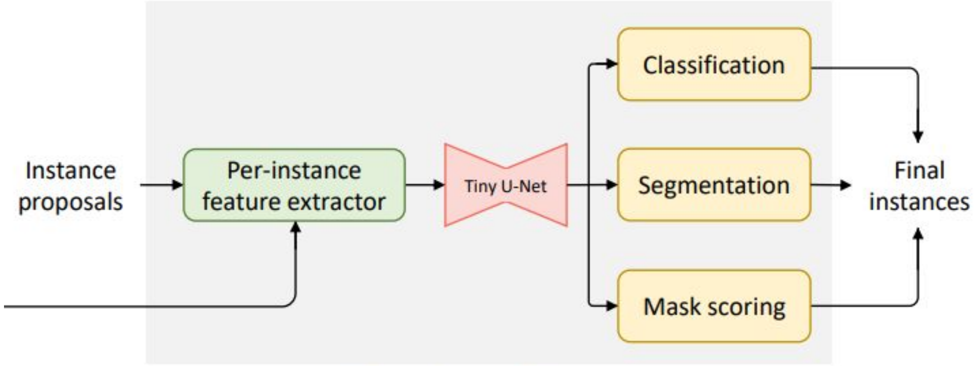
$$\text{precision}_j = \sum_{i=1}^{N} \frac{(s_{ij} > \tau) \wedge (s_i^* = j)}{s_{ij} > \tau}$$

τ	AP	AP_{50}	AP_{25}
None	44.3	65.4	78.1
0.01	40.1	58.5	69.2
0.1	45.3	66.5	78.5
0.2	**46.0**	**67.6**	**78.9**
0.3	45.2	66.8	78.5
0.4	44.7	46.1	78.3
0.5	43.9	64.8	77.7

- Recall increases as the score threshold decreases
- Small score threshold lead to low precision

∴ Propose a top-down refinement stage

Top-Down Refinement



Top-down refinement

$$L_{\text{class}} = \frac{1}{K}\sum_{k=1}^{K}\text{CE}(\boldsymbol{c}_k, c_k^*)$$

$$L_{\text{mask}} = \frac{1}{\sum_{k=1}^{K}\mathbb{1}_{\{\boldsymbol{m}_k\}}}\sum_{k=1}^{K}\mathbb{1}_{\{\boldsymbol{m}_k\}}\text{BCE}(\boldsymbol{m}_k, \boldsymbol{m}_k^*)$$

$$L_{\text{mask\_score}} = \frac{1}{\sum_{k=1}^{K}\mathbb{1}_{\{\boldsymbol{r}_k\}}}\sum_{k=1}^{K}\mathbb{1}_{\{\boldsymbol{r}_k\}}\|\boldsymbol{r}_k - \boldsymbol{r}_k^*\|_2.$$

: treat all instance proposals having IoU with a ground-truth instance higher than 50% as the positive samples and the rest as negative

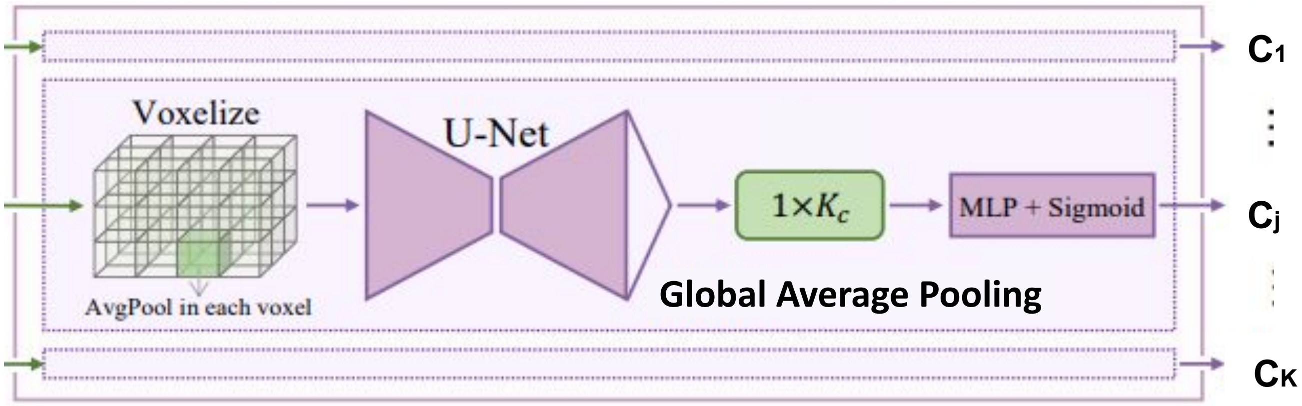
Multitask Learning $L = L_{\text{semantic}} + L_{\text{offset}} + L_{\text{class}} + L_{\text{mask}} + L_{\text{mask\_score}}$

Top-Down Refinement

Classification Branch

: positive sample is the category of the corresponding ground-truth instance

- Classification score $\boldsymbol{C} = \{\boldsymbol{c}_1, ..., \boldsymbol{c}_K\} \in \mathbb{R}^{K \times (N_{\text{class}}+1)}$
- K : # instances
- N_{class}(foreground classes) + 1(background class)



Top-Down Refinement

Classification Branch

- The instance with wrong semantic prediction will be suppressed by learning to categorize it as background.
- Refine the positive sample and suppress the negative one.

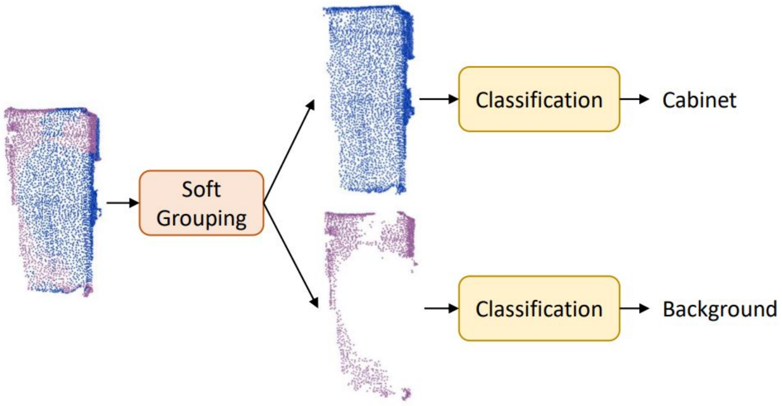


Figure 2. The `cabinet` in Figure 1 is extracted to illustrate the high-level pipeline of our method. The soft grouping module based on soft semantic scores to output more accurate instance (the upper one). The classifier processes each instance and suppress the instance from wrong semantic prediction (the lower one).

Top-Down Refinement

Classification Branch

- directly uses the output of the classification branch as the instance class
- aggregates all point features of the instance and classifies the instance with a single label, leading to more reliable prediction

Category from class branch?	AP	AP_{50}	AP_{25}
N	45.0	65.6	76.2
Y	**46.0**	**67.6**	**78.9**

Table 8. Ablation study on instance category. "N" indicates that the instance category is taken from majority vote of semantic prediction. "Y" indicates that the instance category is taken from classification branch

Top-Down Refinement

Segmentation Branch

- Only trained with positive sample
- Predict an instance mask within each proposal
- point-wise MLP of two layers
- output : instance mask m_k for each instance k

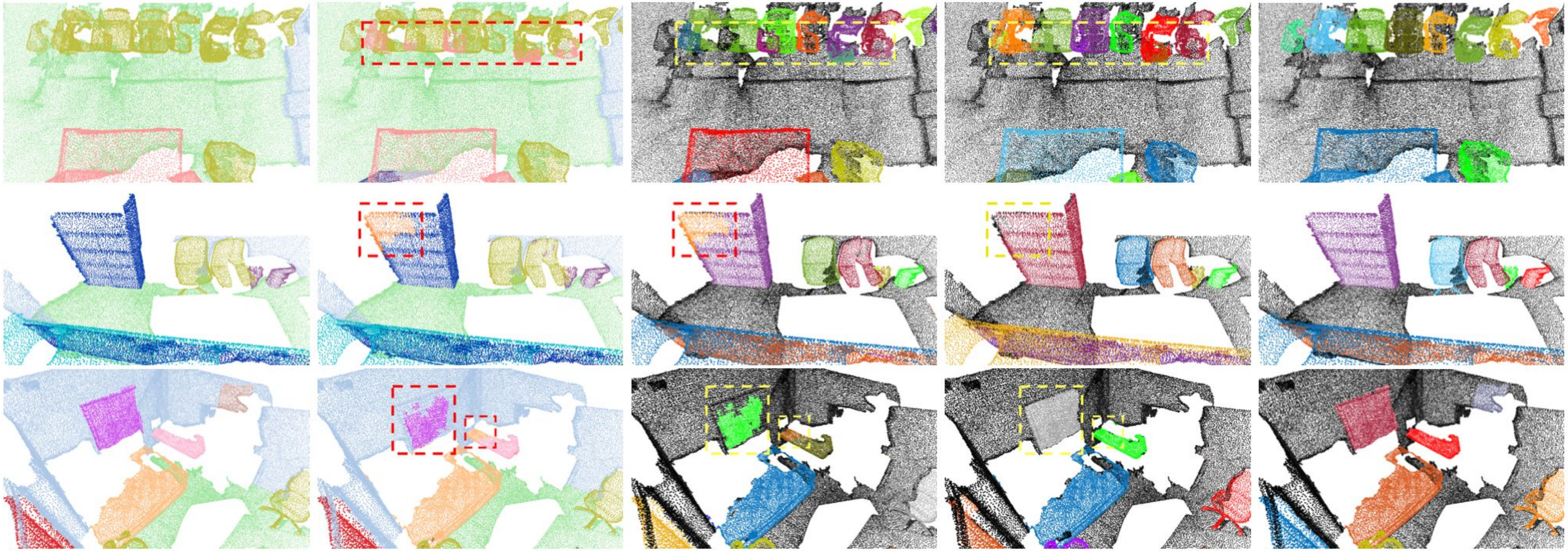
Top-Down Refinement

Mask Scoring Branch

- Only trained with positive sample
- Estimate the IoU of a predicted mask with the ground truth
- Output : mask scores $\boldsymbol{E} = \{\boldsymbol{e}_1, ..., \boldsymbol{e}_K\} \in \mathbb{R}^{K \times N_{\text{class}}}$
- same structure as classification branch
- Every positive sample is assigned to a ground-truth instance with the highest IoU

- $$\begin{aligned} ClassSpecificConfidenceScore &= ConditionalClassProbability * ConfidenceScore \\ &= Pr(Classi|Object) * Pr(Object) * IOU_{pred}^{truth} \\ &= Pr(Classi) * IOU_{pred}^{truth} \end{aligned}$$

Result



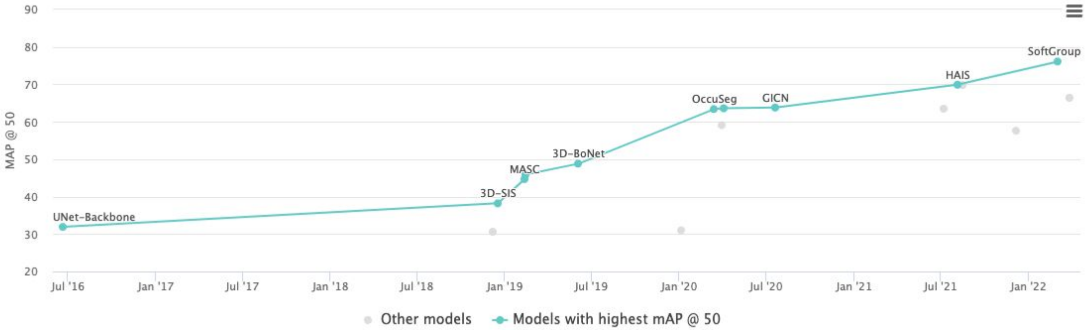
Semantic GT　Semantic pred　Instance pred w/o SoftGroup　Instance pred w/ SoftGroup　Instance GT

Result

3D Instance Segmentation on ScanNet(v2)

Leaderboard   Dataset

View mAP @ 50 by Date



Filter: ResNet untagged

Edit Leaderboard

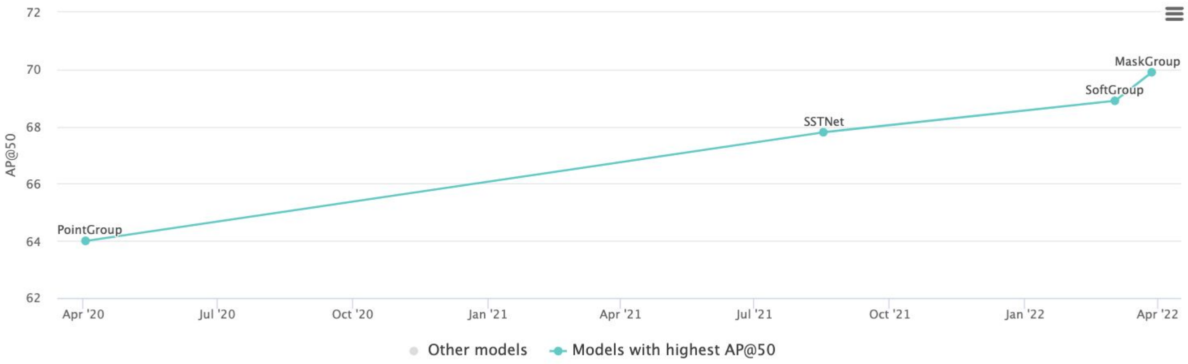
Rank	Model	mAP @ 50	mAP	mRec	Paper	Code	Result	Year	Tags
1	SoftGroup	76.1	50.4		SoftGroup for 3D Instance Segmentation on Point Clouds			2022	
2	HAIS	69.9	45.7		Hierarchical Aggregation for 3D Instance Segmentation			2021	
3	SSTNet	69.8	50.6		Instance Segmentation in 3D Scenes using Semantic Superpoint Tree Networks			2021	
4	MaskGroup	66.4	43.4		MaskGroup: Hierarchical Point Grouping and Masking for 3D Instance Segmentation			2022	
5	GICN	63.8			Learning Gaussian Instance Segmentation in Point Clouds			2020	
6	PointGroup	63.6	40.7		PointGroup: Dual-Set Point Grouping for 3D Instance Segmentation			2020	

Result

3D Instance Segmentation on S3DIS

Leaderboard   Dataset

View AP@50 by Date



Filter: untagged

Edit Leaderboard

Rank	Model	AP@50	mAP	mPrec	mRec	mIoU	mAcc	mCov	mWCov	Paper	Code	Result	Year	Tags
1	MaskGroup	69.9		66.6	69.6					MaskGroup: Hierarchical Point Grouping and Masking for 3D Instance Segmentation			2022	
2	SoftGroup	68.9	54.4	75.3	69.8			69.3	71.7	SoftGroup for 3D Instance Segmentation on Point Clouds			2022	
3	SSTNet	67.8	54.1	73.5	73.4					Instance Segmentation in 3D Scenes using Semantic Superpoint Tree Networks			2021	
4	PointGroup	64.0		69.6	69.2					PointGroup: Dual-Set Point Grouping for 3D Instance Segmentation			2020	