Point Cloud Instance Segmentation using Probabilistic Embeddings
Supplemental

Biao Zhang
KAUST
biao.zhang@kaust.edu.sa

Peter Wonka
KAUST
pwonka@gmail.com

1. Network architecture

Following the notation of PointNet++ [4], we give the architecture of the feature network:

$$
\begin{aligned}
&SA(512, 0.2, [64, 64, 128]),\\
&SA(128, 0.4, [128, 128, 256]),\\
&SA([256, 512, 1024]),\\
&FP(256, 256),\\
&FP(256, 128),\\
&FP(128, 128),
\end{aligned}
$$

where SA and FP are *set abstraction* and *feature propagation* module in PointNet++ [4]. The output head network is:

$$
\begin{aligned}
&FullyConnected(128, 256),\\
&BatchNorm(256),\\
&ReLU(),\\
&FullyConnected(256, 128),\\
&BatchNorm(128),\\
&ReLU(),\\
&FullyConnected(128, 128),\\
&BatchNorm(128),\\
&ReLU(),\\
&FullyConnected(128, C).
\end{aligned}
$$

2. Implementation

We implemented our method using PyTorch [3] and the geometric deep learning library PyTorch Geometric [1]. The final objective function is

$$\mathcal{L} = \mathcal{L}_{Ins} + \mathcal{L}_{Score} + 0.001 \cdot \mathcal{L}_{Reg} \tag{1}$$

3. Results with different IoU thresholds

We report detailed results of IoU using thresholds of 25% and 75% in Table 1 and Table 2. The metric is mean Average Precision (mAP).

4. Qualitive Results

We present more qualitive results in Figure 1 which shows the instance-awareness of our method. We also demonstrate the 3D models in the attached video.

5. Differences to learnable margin

[2] proposed to use a learnable margin for image instance segmentation, which is similar in formulation to our proposed probabilistic embedding. Although we differ in several aspects:

1. The intuition behind learnable margin comes from the hinge loss: to give different hinge margin to objects of different sizes. However, our intuition comes from modeling neural network outputs as random variables to estimate uncertainty.

2. The parameters have a different meaning in our method compared to [2]. In learnable margin, σ is an instance-specific bandwidth (or margin) per cluster. In our work σ are uncertainties per point.

3. The bandwidth σ is influenced by the size of instances (large instances have large σ). In contrast, our uncertainty σ encodes per-point uncertainty close to the boundary of instances (see Fig. 7).

4. [2] add a loss term to enforce the bandwidths from the same instance to be close. By contrast, we don't have this kind of restriction. Also, uncertaintes from the same instance can be different as along as they have similar spatial embeddings.

		Avg	Bag	Bed	Bottle	Bowl	Chair	Clock	Dish	Disp	Door	Ear	Faucet	Hat	Key	Knife	Lamp	Laptop	Micro	Mug	Fridge	Scis	Stora	Table	Trash	Vase
PartNet	1	70.2	**89.4**	**82.3**	65.2	63.1	78.1	48.0	79.1	97.1	64.9	64.6	77.3	73.9	58.9	59.2	42.5	**100.0**	50.0	92.9	50.0	96.3	57.7	**59.3**	82.7	**52.6**
	2	46.7	-	44.5	-	-	43.0	-	71.3	-	**49.3**	-	-	-	-	-	32.2	-	51.2	-	45.2	-	46.7	36.5	-	-
	3	45.6	-	29.0	52.6	-	35.3	39.6	59.9	89.3	27.1	56.9	55.0	-	-	49.0	22.6	-	56.9	-	35.6	-	36.3	28.6	44.8	**57.0**
	Avg	62.8	**89.4**	51.9	58.9	63.1	52.1	43.8	70.1	93.2	47.1	60.8	66.2	73.9	58.9	54.1	32.4	**100.0**	52.7	92.9	43.6	96.3	46.9	**41.5**	63.8	**54.8**
Ours	1	**72.7**	82.8	79.6	**65.6**	**72.0**	**82.8**	**49.1**	**83.8**	**98.3**	**75.5**	**74.3**	**83.2**	**79.5**	**59.9**	**78.8**	**45.2**	**100.0**	**50.5**	**95.4**	**51.6**	**96.9**	**60.9**	44.6	**82.9**	51.1
	2	**51.4**	-	**55.4**	-	-	**47.1**	-	**78.0**	-	48.1	-	-	-	-	-	**39.3**	-	**54.4**	-	**48.8**	-	**53.7**	**37.7**	-	-
	3	**51.6**	-	**44.4**	**57.2**	-	**43.2**	**45.7**	**64.8**	**90.7**	**34.6**	**59.3**	**67.2**	-	-	**53.0**	**26.0**	-	**60.0**	-	**51.5**	-	**44.4**	**31.7**	**50.0**	53.9
	Avg	**66.5**	82.8	**59.8**	**61.4**	**72.0**	**57.7**	**47.4**	**75.6**	**94.5**	**52.7**	**66.8**	**75.2**	**79.5**	**59.9**	**65.9**	**36.8**	**100.0**	**55.0**	**95.4**	**50.6**	**96.9**	**53.0**	38.0	**66.5**	52.5

Table 1: **Instance segmentation results on PartNet**. The metric is mAP (%) with IoU threshold 0.25.

		Avg	Bag	Bed	Bottle	Bowl	Chair	Clock	Dish	Disp	Door	Ear	Faucet	Hat	Key	Knife	Lamp	Laptop	Micro	Mug	Fridge	Scis	Stora	Table	Trash	Vase
PartNet	1	47.4	39.7	**14.6**	60.6	41.4	58.3	**28.8**	58.3	84.7	35.6	49.1	48.2	66.3	**10.7**	48.7	**29.6**	**98.0**	47.8	76.1	50.0	35.1	29.9	**43.2**	**42.2**	40.5
	2	22.0	-	4.2	-	-	21.4	-	37.2	-	**22.4**	-	-	-	-	-	19.6	-	32.1	-	16.7	-	22.8	22.0	-	-
	3	23.5	-	3.9	37.9	-	16.6	**17.6**	29.8	63.2	8.1	27.6	25.8	-	-	31.0	13.6	-	23.9	-	12.1	-	18.2	16.4	**19.7**	34.5
	Avg	38.9	39.7	7.6	49.2	41.4	32.1	**23.2**	41.7	73.9	22.0	38.4	37.0	66.3	**10.7**	39.8	20.9	**98.0**	34.6	76.1	26.3	35.1	23.6	**27.2**	**31.0**	37.5
Ours	1	**50.0**	**40.3**	13.3	**60.2**	**60.2**	**59.3**	28.2	**61.9**	**90.6**	**39.1**	**59.6**	**54.2**	**69.3**	7.4	**65.7**	28.5	**98.0**	**47.9**	**77.1**	**50.5**	**42.8**	**30.1**	34.8	40.7	**41.1**
	2	**23.8**	-	**7.1**	-	-	22.8	-	**37.4**	-	21.3	-	-	-	-	-	**22.0**	-	**35.5**	-	**20.6**	-	**26.1**	**21.4**	-	-
	3	**25.7**	-	**7.3**	**38.8**	-	**20.5**	17.2	**30.0**	**66.8**	**10.8**	**28.2**	**33.2**	-	-	**31.5**	**14.1**	-	**25.6**	-	**17.1**	-	**21.0**	**17.4**	19.4	**38.0**
	Avg	**41.7**	**40.3**	**9.2**	**49.5**	**60.2**	**34.2**	22.7	**43.1**	**78.7**	**23.7**	**43.9**	**43.7**	**69.3**	7.4	**48.6**	**21.5**	**98.0**	**36.4**	**77.1**	**29.4**	**42.8**	**25.7**	24.5	30.0	**39.6**

Table 2: **Instance segmentation results on PartNet**. The metric is mAP (%) with IoU threshold 0.75.

References

[1] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019. 1

[2] Davy Neven, Bert De Brabandere, Marc Proesmans, and Luc Van Gool. Instance segmentation by jointly optimizing spatial embeddings and clustering bandwidth. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8837–8845, 2019. 1

[3] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035, 2019. 1

[4] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*, pages 5099–5108, 2017. 1

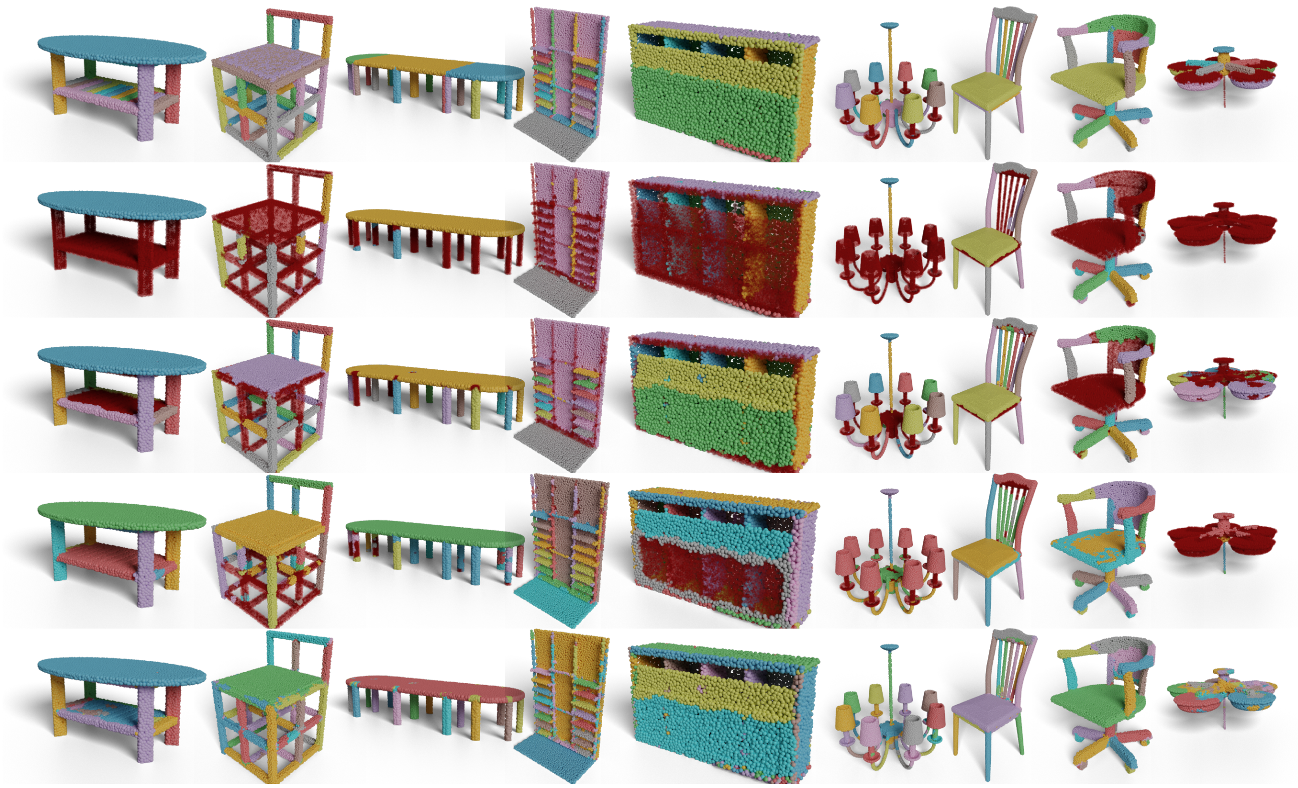

Figure 1: **Top row**: ground-truth (background points are shown in transparent red). **Second and third row**: PartNet and Ours (only *true positives* are shown, and false detections are shown in transparent red). **Fourth and fifth row**: PartNet and Ours (*all* detected instances, unclassified points are shown in transparent red). PartNet can group instance points together but fails to give the correct class labels in some cases (*e.g.*, in the first and the third subfigures from left to right, points of table legs are grouped together (fourth row) but they are not true positives (second row). Besides, in the sixth subfigure from left to right, PartNet fails to distinguish different instances of lamp covers (second and fourth row). Our method performs clearly better in these cases.