ToThePoint:

Efficient Contrastive Learning of 3D Point Clouds via Recycling

Xinglin Li, Jiajing Chen, Jinhui Ouyang, Hanhui Deng,

Senem Velipasalar, Di Wu

湖南大学 HUNAN UNIVERSITY

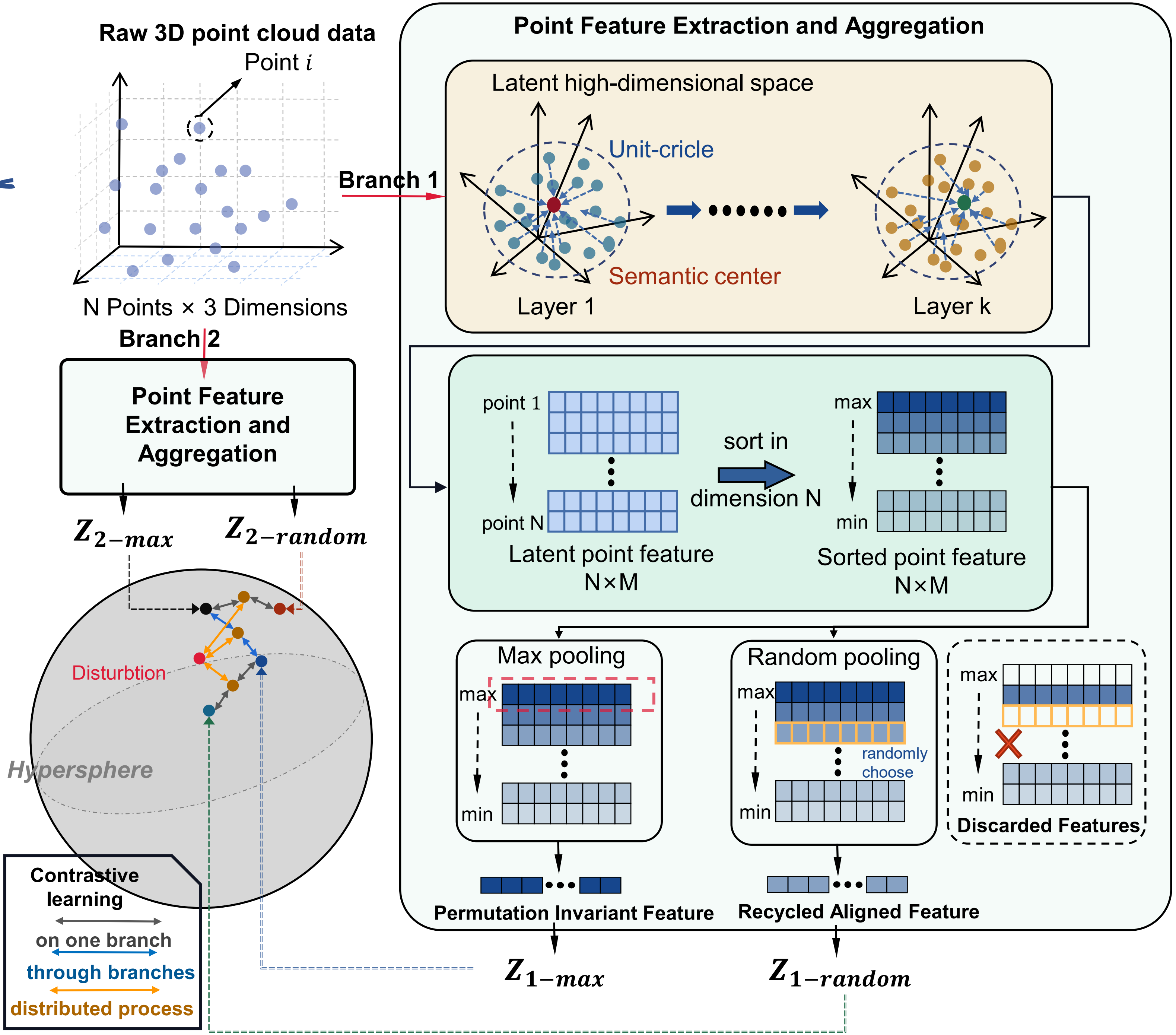
SYRACUSE UNIVERSITY · FOUNDED 1870 · SUOS CULTORES SCIENTIA CORONAT

JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

THU-PM-112

Outline

- Raw 3D point cloud data is streamed through two branches
- in each branch, normalization and data augmentation are performed followed by traditional max-pooling operation
- In recycling mechanism, features are sorted and a row of features is randomly selected as the recycled aligned features to assist the representation of permutation invariant features
- The four features extracted from the two branches are next subjected to two stages of contrastive learning
- Then the learning result would be mapped on the hypersphere



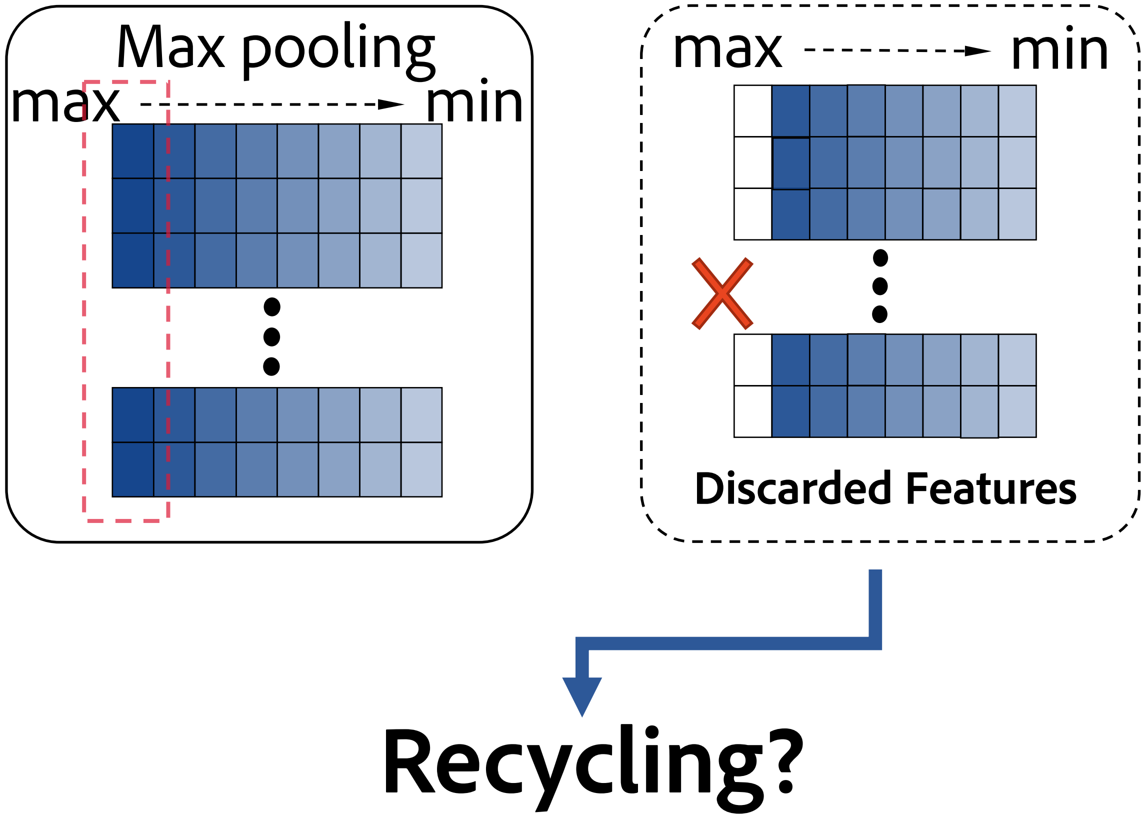
Background

3D point cloud self-supervised pre-training

Backbones: PointNet, DGCNN, etc.

Max-pooling —> permutation-invariant features

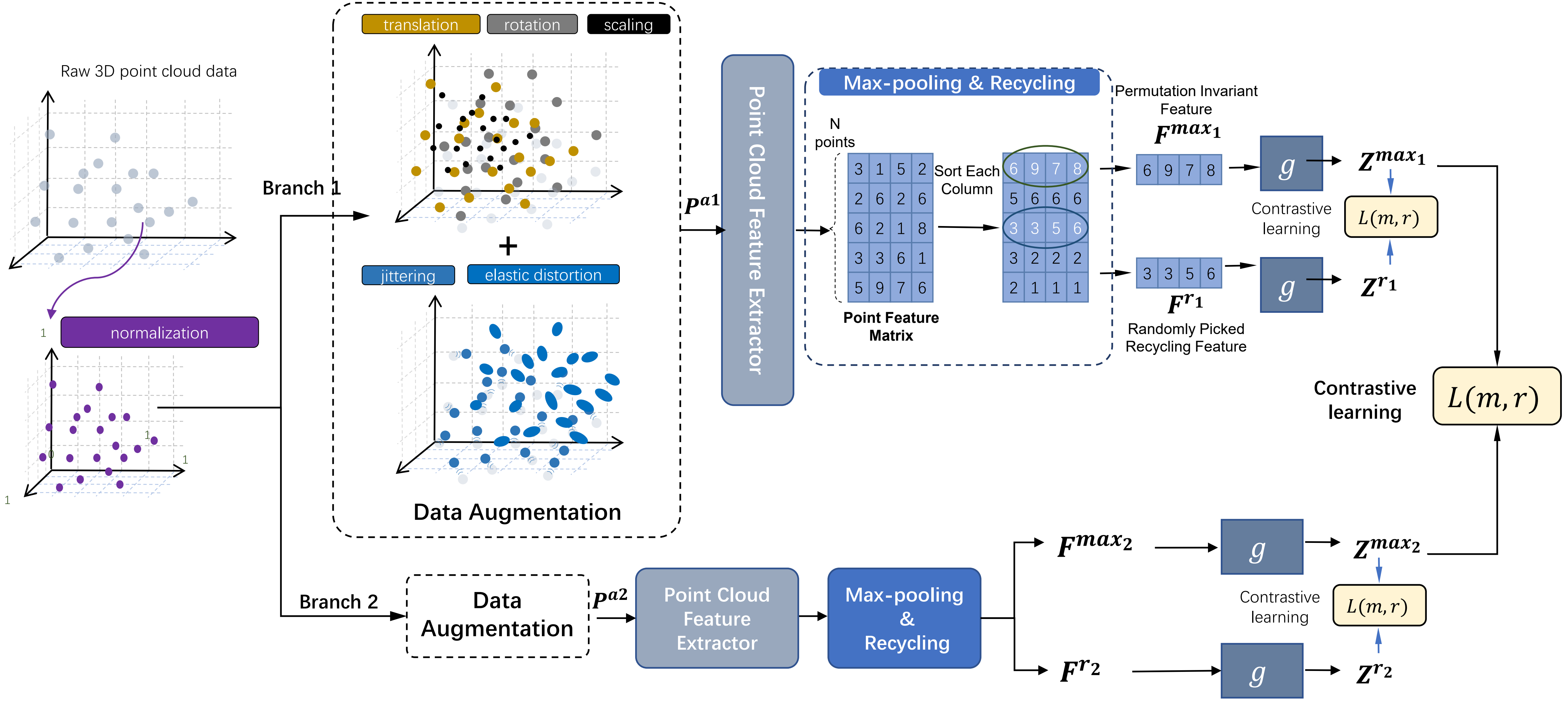
In max-pooling, a large number of points and their features are discarded



Our Work

- We first demonstrate that the point cloud features, discarded by the max-pooling module of a point cloud network, can be **recycled and used as a feature augmentation method** for contrastive learning.
- We propose a **two-branch contrastive learning framework**, which incorporates a cross-branch contrastive learning loss and an intra-branch contrastive learning loss.
- We perform experiments to evaluate our proposed method on downstream tasks including **object classification**, **few-shot learning**, and **part segmentation.** Compared to the state-of-the-art baselines, our work obtain competitive performance with significantly **less training time** and **fewer training samples**.
- We perform **ablation studies** analyzing the effects of individual loss terms and their combinations on the performance.

ToThePoint Framework



Algorithms

(1) NT-Xent loss between maximum features and recycled point features：

$$L(i,m,r)=-\log\frac{\exp(s(z_i^{max_j},z_i^{r_j})/\tau)}{\sum_{\substack{k=1\\k\neq i}}^{B}\exp(s(z_i^{max_j},z_k^{max_j})/\tau)+\sum_{k=1}^{B}\exp(s(z_i^{max_j},z_k^{r_j})/\tau)}$$

(2) Intra-branch contrastive loss：

$$\mathcal{L}_j^{ib-cl}=\frac{1}{2B}\sum_{i=1}^{B}[L(i,m,r),L(i,r,m)]$$

(3) NT-Xent loss between maximum features of first and second branches：

$$L(i,max_1,max_2)=-\log\frac{\exp(s(z_i^{max_1},z_i^{max_2})/\tau)}{\sum_{\substack{k=1\\k\neq i}}^{B}\exp(s(z_i^{max_1},z_k^{max_1})/\tau)+\sum_{k=1}^{B}\exp(s(z_i^{max_1},z_k^{max_1})/\tau)}$$

(4) Inter-branch contrastive loss：

$$\mathcal{L}^{cb-cl}=\frac{1}{2B}\sum_{i=1}^{B}[L(i,max_1,max_2),L(i,max_2,max_1)]$$

(5) Total loss：

$$\mathcal{L}=\mathcal{L}_1^{ib-cl}+\mathcal{L}_2^{ib-cl}+\mathcal{L}^{cb-cl}$$

Evaluation Settings

Pre-training:

- **Dataset**: ShapeNet
- **Backbone**: PointNet, DGCNN

Downstream Tasks

- **Dataset**: **3D object classification** - ModelNet40, ModelNet40C, canObjectNN;
  Few-shot 3D object classification - ModelNet40, ScanObjectNN;
  3D object part segmentation - ShapeNet-Part.
- **Baselines**: Jigsaw, OcCo, STRL, CrossPoint, cTree, etc.

Evaluation Results - 3D object classification

Self-supervised Learning				Downstream Task			
Method	Num of Samples	Running Time (s.)		ModelNet40C		ScanObjectNN	
		DGCNN	PointNet	DGCNN	PointNet	DGCNN	PointNet
Rand	/	/	/	81.82±0.07	79.63±0.25	85.66±0.45	78.16±0.54
Jigsaw [20]	9.8K	161.27	17.97	83.19±0.26	80.14±0.35	86.33±0.06	79.46±0.17
OcCo [26]	9.8K	2762.7	1307.73	82.47±0.15	79.89±0.52	86.19±0.39	79.63±0.16
STRL [11]	57.4 k	238.65	175.4	82.66±0.38	80.43±0.19	86.17±0.32	80.32±0.21
CrossPoint [2]	43.7 k pnts & 1.05M img	1115.86	344.95	83.69±0.29	**81.12±0.44**	86.32±025	79.90±0.03
ToThePoint	**260**	**5.38**	**1.3**	**83.80±0.32**	80.97±0.27	**86.46±0.21**	**80.64±0.31**

Table 2. **3D object classification comparison.**. We report mean and standard deviation over 3 runs ToThePoint outperforms all the other methods on the ScanObject dataset with both backbones. On the ModelNet40C dataset, ToThePoint provides the best and second-best performance when DGCNN and PointNet are used as backbones, respectively. ToThePoint achieves these accuracies with only a fraction of training samples needed by other methods.

Method	Accuracy	
	ModeNet40	**ScanObjectNN**
3D-GAN [29]	81.85	37.01
Latent-GAN [1]	87.64	71.94
3D-PointCapsNet [37]	76.62	53.70
SO-Net [14]	87.03	/
PointNet + Jigsaw [20]	51.90	35.11
PointNet + OcCo [26]	86.67	68.84
PointNet + STRL [11]	88.05	73.67
PointNet + CrossPoint [2]	**88.82**	72.29
PointNet + ToThePoint (Ours)	85.62	**74.70**
DGCNN + Jigsaw [20]	55.71	36.31
DGCNN + OcCo [26]	88.61	78.14
DGCNN + STRL [11]	**90.60**	78.14
DGCNN + CrossPoint [2]	90.03	81.43
DGCNN + ToThePoint (Ours)	89.22	**81.93**

Table 3. **SVM classification results on ModelNet40 and ScanObjectNN.** We perform the SVM evaluation method [1], to compare ToThePoint and baselines with PointNet and DGCNN used as backbones. On the more challenging ScanObjectNN dataset, proposed ToThePoint achieves the best performance with both backbones. On ModelNet40 dataset, ToThePoint provides the 3rd best performance after CrossPoint and STRL, which require a lot more training samples.

Evaluation Results - Few-shot 3D object classification

Self-supervised Learning				Downstream Task (Few-shot point cloud classification)			
Method	Num of Samples	Running Time (s.)		5-way		10-way	
		DGCNN	PointNet	10 shot	20 shot	10 shot	20 shot
3D-GAN [29]	/	/	/	87.72 ± 5.44	91.98 ± 3.91	81.31 ± 4.75	84.87 ± 5.10
DGCNN Rand	/	/	/	81.13 ± 8.68	85.96 ± 6.60	72.86 ± 7.33	81.03 ± 5.12
DGCNN cTree [21]	200	**2.53**	2.53	86.37 ± 6.29	89.60 ± 5.62	81.03 ± 4.14	83.98 ± 4.75
DGCNN Jiasaw	9.8K	161.27	17.97	87.06 ± 5.93	88.60 ± 6.07	79.20 ± 4.41	83.21 ± 4.40
DGCNN OcCo [26]	9.8K	2762.7	1307.73	88.46 ± 8.15	94.13 ± 3.73	85.21 ± 3.91	87.11 ± 3.93
DGCNN CrossPoint [2]	43.7 k pnts & 1.05 M images	1115.86	344.95	91.12 ± 4.93	94.56 ± 3.23	86.29 ± 4.77	88.96 ± 4.39
DGCNN ToThePoint	260	5.38	**1.3**	**92.73 ± 4.79**	**95.10 ± 2.95**	**87.91 ± 4.29**	**91.06 ± 3.58**

(a) Experiment results on ModelNet40

Self-supervised Learning				Downstream Task (Few-shot point cloud classification)			
Method	Num of Samples	Running Time (s.)		5-way		10-way	
		DGCNN	PointNet	10 shot	20 shot	10 shot	20 shot
3D-GAN [29]	/	/	/	68.20 ± 7.84	72.68 ± 9.76	53.93 ± 4.73	59.62 ± 4.66
DGCNN Rand	/	/	/	61.80 ± 7.60	64.10 ± 8.67	42.13 ± 3.96	49.11 ± 6.08
DGCNN cTree [21]	200	**2.53**	2.53	50.76 ± 7.11	72.68 ± 9.76	37.46 ± 4.03	41.76 ± 4.72
DGCNN Jiasaw	9.8K	161.27	17.97	67.16 ± 8.32	72.76 ± 9.39	50.75 ± 5.40	58.75 ± 5.33
DGCNN OcCo [26]	9.8K	2762.7	1307.73	75.80 ± 5.48	82.06 ± 5.90	63.43 ± 4.60	71.48 ± 4.28
DGCNN CrossPoint [2]	43.7 k pnts & 1.05 M images	1115.86	344.95	77.24 ±7.13	83.68 ±5.51	66.61 ±3.96	73.57±4.72
DGCNN ToThePoint	260	5.38	**1.3**	**78.13 ± 7.29**	**83.80 ± 6.07**	**66.71 ± 5.40**	**74.21 ± 4.95**

(b) Experiment Results on ScanObjectNN

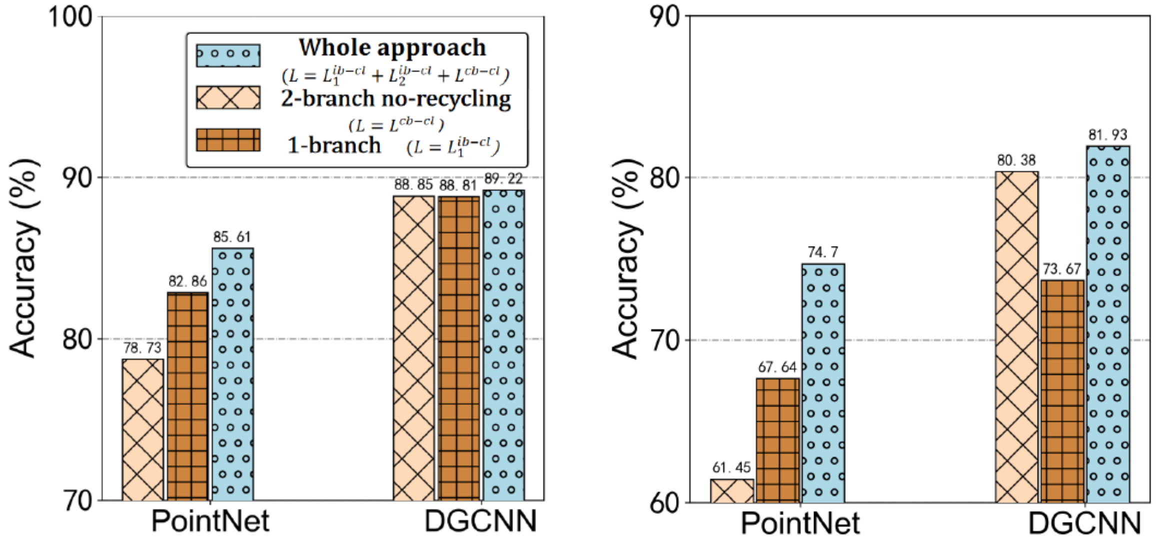
Table 4. **Few-shot object classification results.** We report mean and standard deviation over 30 runs. The top results for each backbone are shown in bold. Our proposed ToThePoint needs only a few samples in the few-shot learning task and improves the few-shot accuracy in all the reported settings.

Evaluation Results - 3D object part segmentation

Self-supervised Learning				Downstream Task (Part segmentation)			
Method	Num of Samples	Running Time (s.)		DGCNN		PointNet	
		DGCNN	PointNet	Mean IoU	OA	Mean IoU	OA
Rand	/	/	/	85.16	94.43	84.48	93.82
Jigsaw [20]	9.8K	161.27	17.97	85.34	94.42	84.27	93.67
OcCo [26]	9.8K	2762.7	1307.73	85.32	**94.5**	84.56	93.77
CrossPoint [2]	43.7 k pnts & 1.05 M images	1115.86	344.95	85.38	94.44	84.77	93.97
ToThePoint	**260**	**5.38**	**1.3**	**85.5**	94.44	**84.91**	**94.05**

Table 5. **Part segmentation results on ShapeNet-Part dataset.** Mean IoU and overall accuracy (OA) are reported. All self-supervised models are initialized with pre-trained feature extractors.

Ablation Studies



(a) Accuracy on ModelNet40 (b) Accuracy on ScanObjectNN

Figure 4. **The ablation study results on effects of individual loss terms**. Blue bar represents the whole approach using all 3 loss terms, light orange corresponds to using two branches but no recycling and dark orange corresponds to using one branch with recycling. Classification accuracies are presented on ModelNet40 and ScanObjectNN datasets.

$$\mathcal{L}^{cb-cl} = \frac{1}{2B}\sum_{i=1}^{B}[L(i, max_1, max_2) + L(i, max_2, max_1)]. \quad (4)$$

Component	2-Branch, no recycling		1-branch with recycling	
Backbone	PointNet	DGCNN	PointNet	DGCNN
ModelNet40	6.89%	0.37%	2.76%	0.41%
ScanObjectNN	13.25%	1.56%	7.06%	8.27%

Table 6. **The accuracy reduction caused by different configurations.**

Conclusion & Future Work

- We have proposed ToThePoint as a novel and very efficient contrastive learning framework. In addition to using traditional data augmentation, ToThePoint performs feature augmentation by recycling point cloud features, which would be discarded after max-pooling operation of a point cloud feature extraction network.

- In future work, we will investigate whether using more branches or recycling more features can provide additional benefit

Thanks for your attention!

湖南大學
HUNAN UNIVERSITY

JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

SYRACUSE UNIVERSITY
SUOS CULTORES SCIENTIA CORONAT
FOUNDED 1870