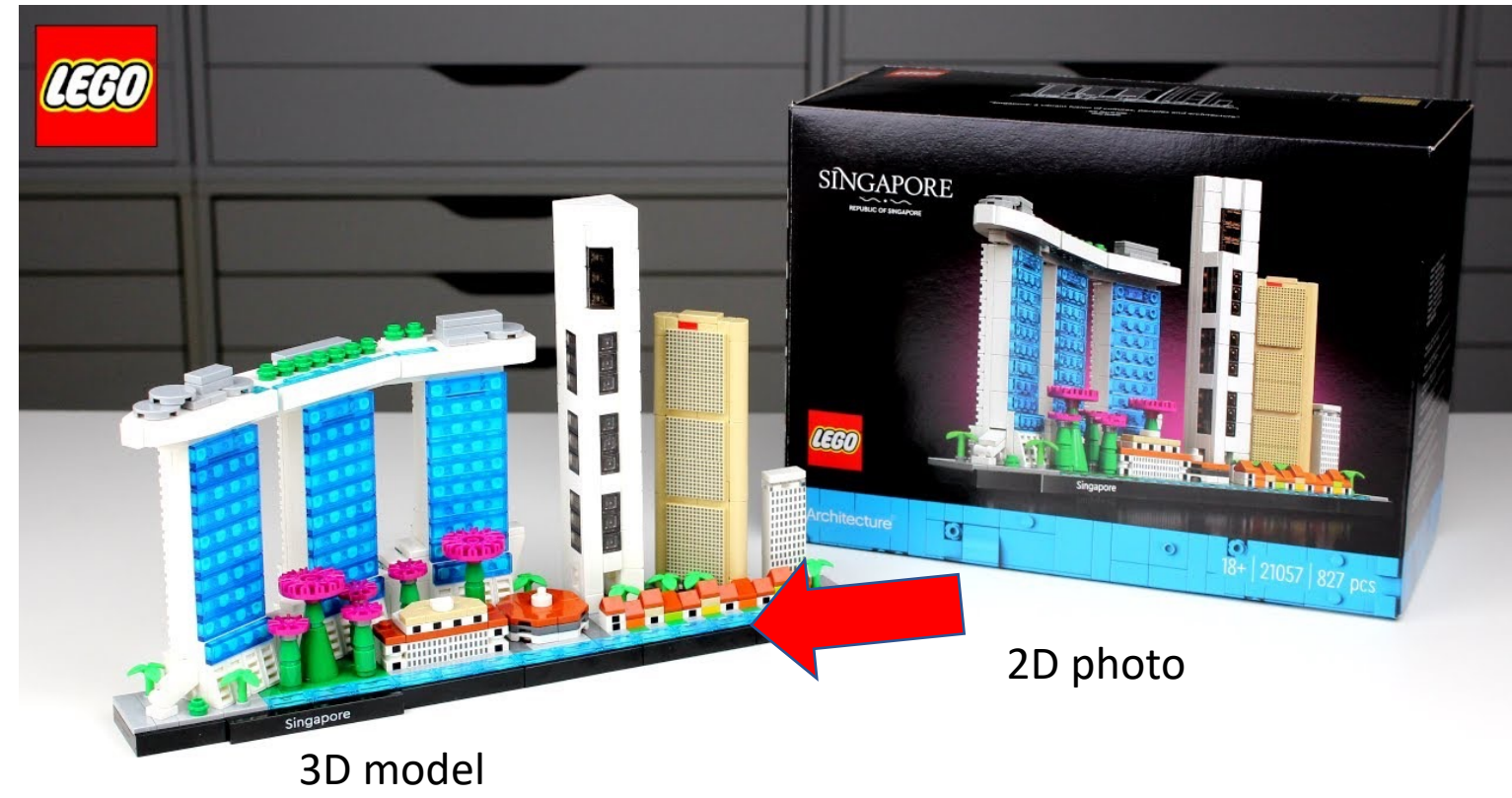


➤ Motivation

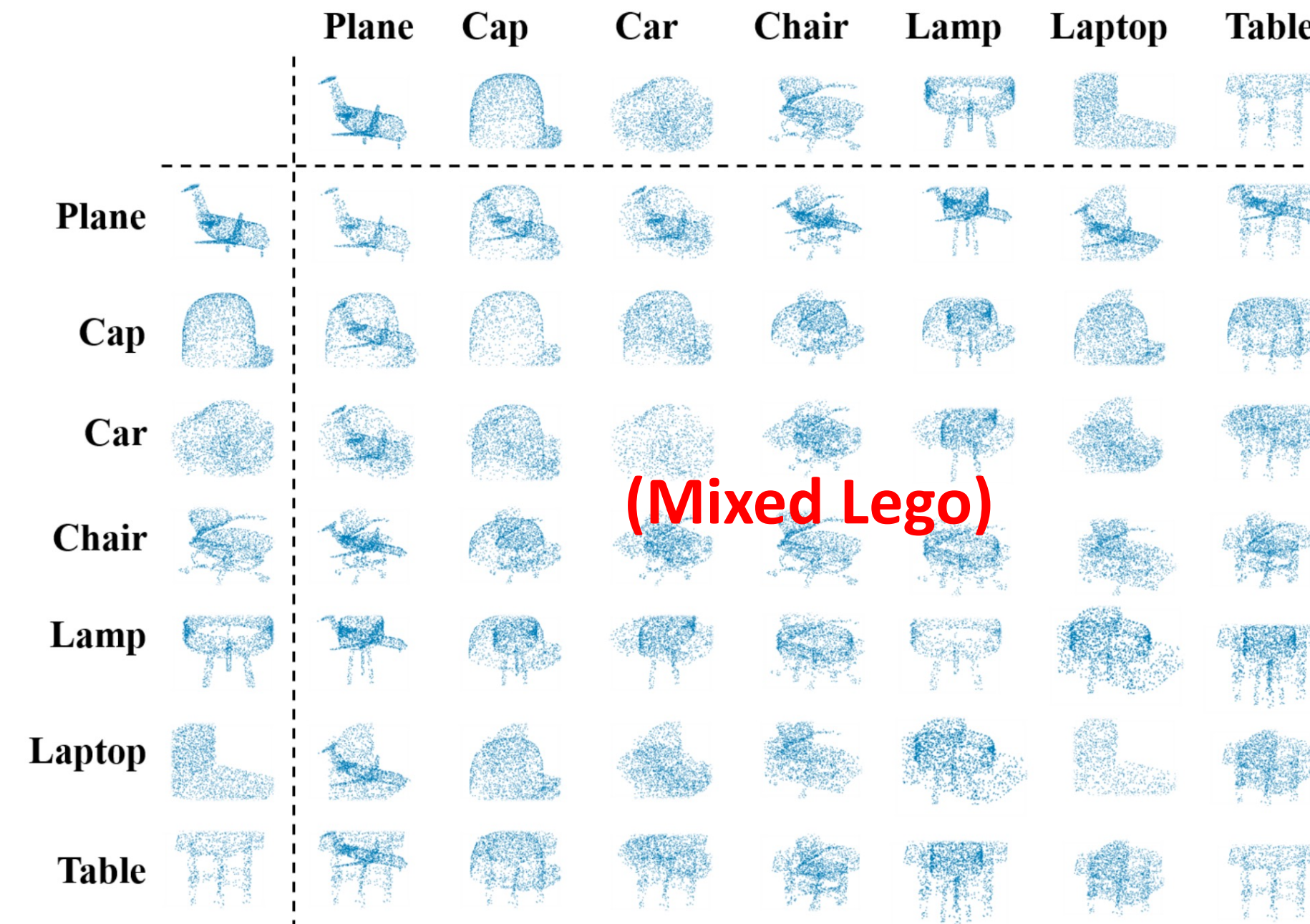
People can reconstruct the 3D models (left) according to 2D projection (right). **How about we ask the deeply-learned model to do that as pretext task?**

**➤ Contribution**

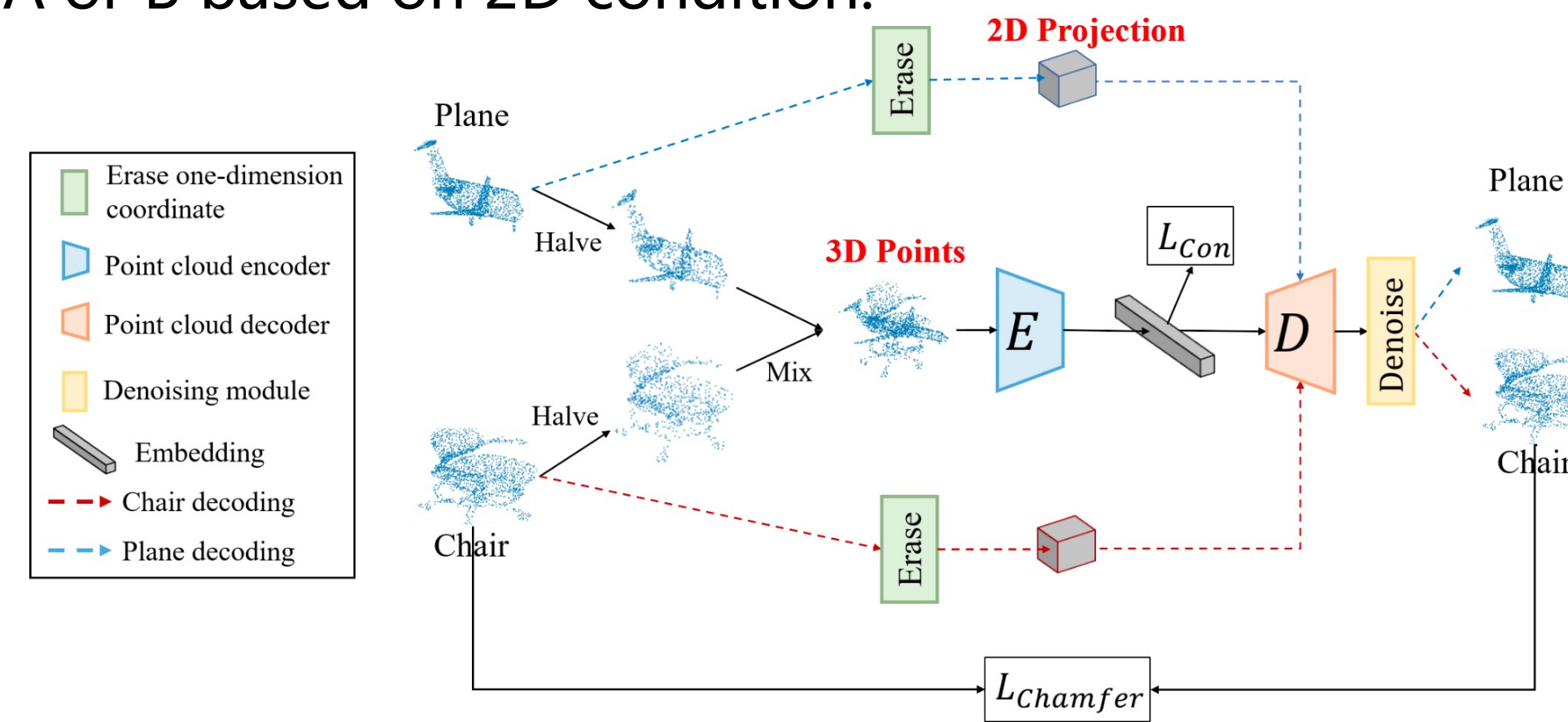
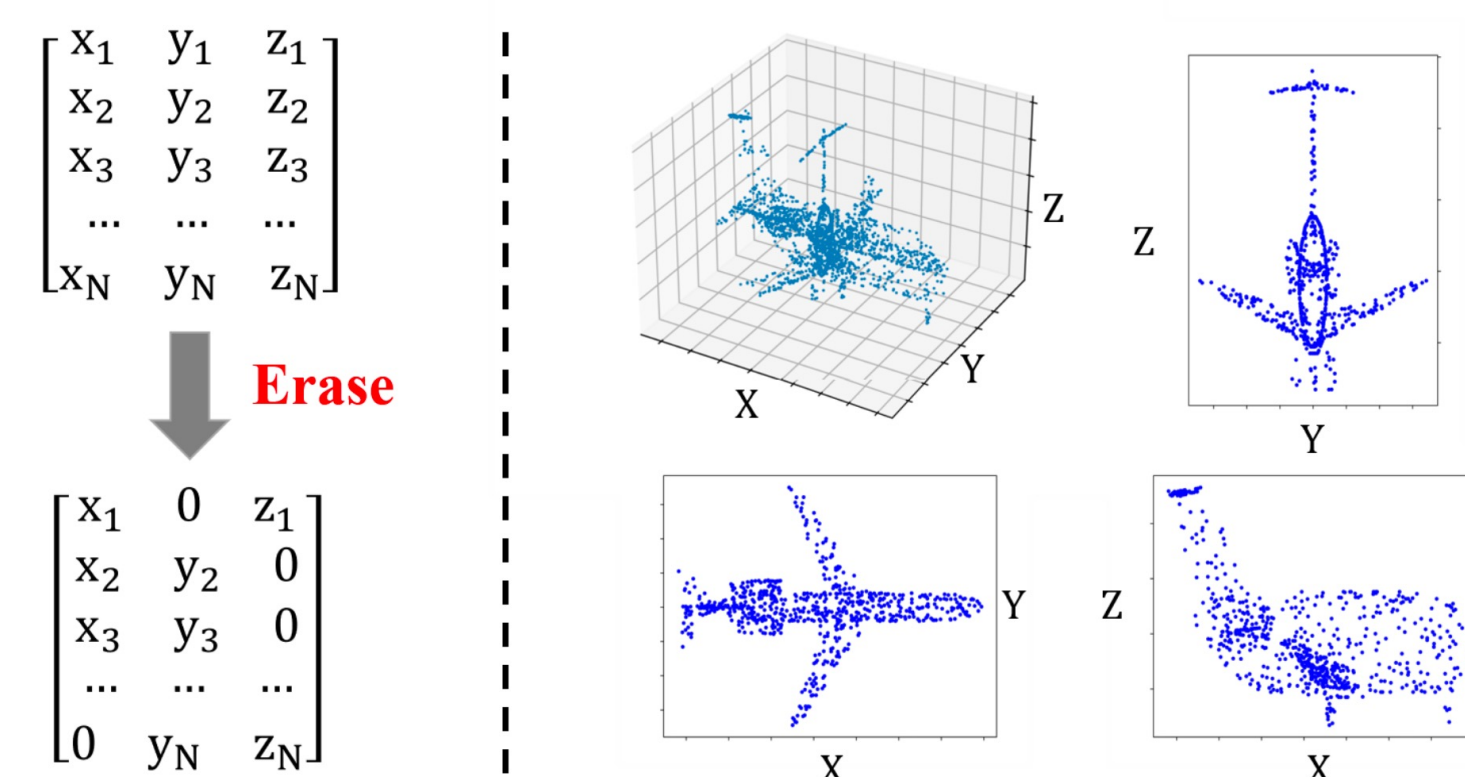
- We leverage the mixing process to generate large-scale mixed data for 3D point cloud training. Inspired by the human ability to figure out two shapes from one mixed object, we propose **a new self-supervised learning method on the point cloud, called Mixing and Disentangling (MD)**, to learn the geometric prior knowledge without the requisite of annotations
- We implement one basic pipeline to verify the effectiveness of the proposed self-supervised learning strategy. It contains one encoder with the learnable aggregation function and one instance- adaptive decoder, to learn from the mixed point cloud and conduct the disentanglement. Albeit simple, experimental results on two benchmarks, *i.e.*, **ModelNet-40** and **ShapeNet-Part**, show that the self-supervised learning model can effectively and efficiently improve the accuracy of classification and segmentation tasks by the pre-training and fine-tuning paradigm. **Self-supervised learning on the point cloud also reduces the network dependence on labeled data.**

➤ Method**■ Data Mixing**

Our data mixing process is simple, which randomly selects 50% points from A and 50% points from B.

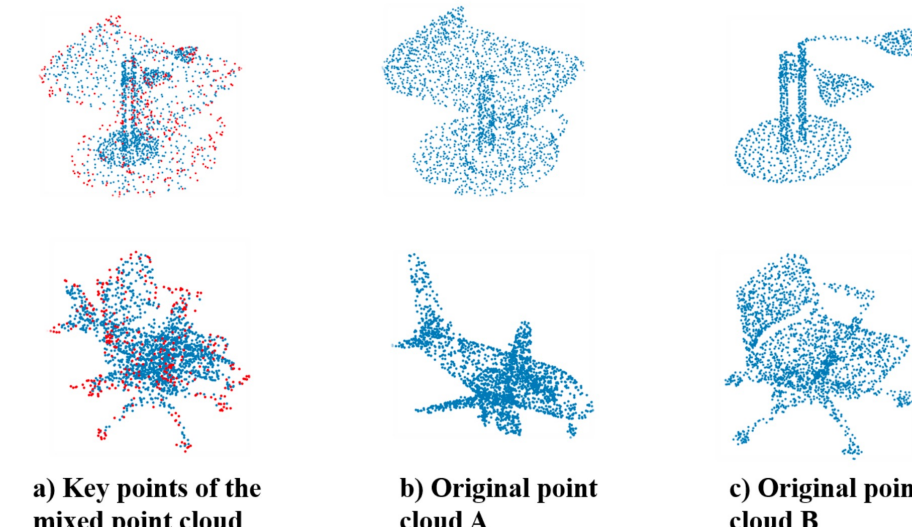
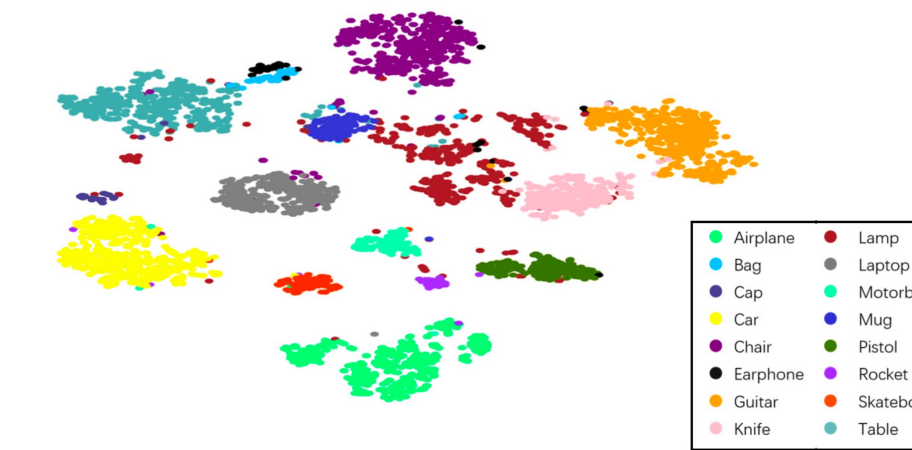
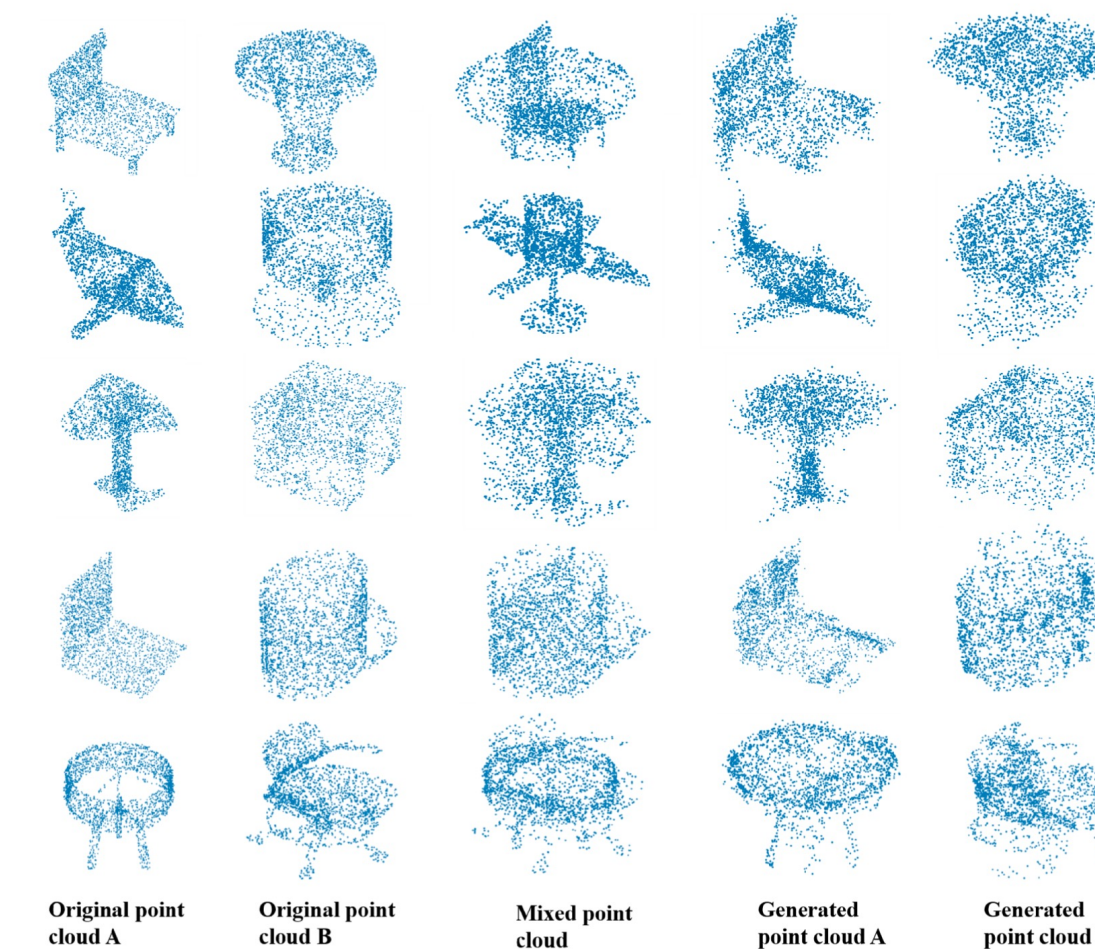
**■ Disentangle Learning**

The Encoder is to learn the salient pattern in A and B. During decoding, the Decoder is asked to reconstruct A or B based on 2D condition.

**■ 2D Projection as Condition****➤ Experiment****■ Downstream tasks
(Classification & Segmentation)**

Methods	Publication	ModelNet-40 Classification		ShapeNet-Part Segmentation
		OA (%)	mA (%)	
3DShapeNets [11]	CVPR'15	84.7	77.3	-
VoxNet [31]	IROS'15	85.9	83.0	-
PointNet [35]	CVPR'17	89.2	86.0	83.7
PointNet++ [36]	NeurIPS'17	91.9	-	85.1
SpecGCN [73]	ECCV'18	91.5	-	-
PCNN by Ext [74]	SIGGRAPH'18	92.2	-	85.1
DGCNN [39]	TOG'19	92.9	90.2	85.1
Point Trans. [43]	ICCV'21	93.7	90.6	86.6
PACConv [41]	CVPR'21	93.9	-	86.1
CurveNet [75]	ICCV'21	94.2	-	86.8
Ours (from scratch)	-	92.74	89.88	85.27
Ours (pre-training)	-	93.39	90.26	85.50

OA: Overall Accuracy; mA: Mean Class Accuracy; mIoU: mean IoU

**■ Visualization of the key points
of the mixed point cloud****■ Good Scalability
(ModelNet-40 to ShapeNet-Part)****■ Reconstruction results****■ Fusion with other methods**

Methods	Pre-trained	ModelNet-40 Classification	
		OA (%)	mA (%)
Ours	-	92.74	89.88
Ours	ShapeNet-Part	92.79	90.10
Ours	ModelNet-40	93.39	90.26
PointNet++ [36] *	-	92.07	88.89
PointNet++ * + Ours	ShapeNet-Part	92.19	89.63
PointNet++ * + Ours	ModelNet-40	92.57	89.96
OGNet* [76]	-	93.23	89.82
OGNet* + Ours	ShapeNet-Part	93.35	90.51
OGNet* + Ours	ModelNet-40	93.31	90.71
PACConv* (PN) [41]	-	92.50	-
PACConv* (PN) + Ours	ShapeNet-Part	92.70	-
PACConv* (PN) + Ours	ModelNet-40	92.79	-
PT* [43]	-	91.47	89.32
PT* + Ours	ShapeNet-Part	92.03	89.50
PT* + Ours	ModelNet-40	92.07	89.58
PCT* [42]	-	92.71	89.36
PCT* + Ours	ShapeNet-Part	93.07	90.27
PCT* + Ours	ModelNet-40	93.15	90.56

PT donates Point Transformer. PCT donates Point Cloud Transformer. PACConv (PN) donates using PointNet as the backbone (without voting). OA: Overall Accuracy; mA: Mean Class Accuracy; mIoU: mean IoU; *: We re-implement the model, which achieves a slightly different performance.

■ Limited labeled data

Labeled Data Ratio (%)	Pre-trained	Average Accuracy (%)	mIoU (%)
10	-	74.02	81.59
10	✓	77.04	82.23
25	-	80.39	83.76
25	✓	81.11	84.49
50	-	81.46	84.71
50	✓	83.31	85.04

➤ Conclusion

In this paper, we propose a new self-supervised learning method, called Mixing and Disentangling (MD), for point cloud pre-training. Different from existing works, we propose to mix the original point clouds in the training set to form “new” data and then demand the model to “separate” the mixed point cloud. In this way, the model is asked to mine **the geometric knowledge, e.g., the shape-related key points for reconstruction.** (Chinese blog is here) (Code is here)

