Hyperspherical Embedding for Point Cloud Completion

CVPR 2023

Junming Zhang[1], Haomeng Zhang[1], Ram Vasudevan[1], Matthew Johnson-Roberson[2]

University of Michigan[1] Carnegie Mellon University[2]

TUE-PM-113

UNIVERSITY OF MICHIGAN
Carnegie Mellon University
FCAV
UM Ford Center for Autonomous Vehicles
Ford
NSF

Quick View - Motivation

- Traditional Encoder-Decoder architecture for point cloud completion *(e.g., FoldingNet [Yang et al. 2018], PCN [Wang et al. 2018], TopNet [Tchapmi et al. 2019], SnowFlakeNet [Xiang et al. 2021])* learns sparse embedding distribution, which leads to worse generalization results during testing.

- Different embedding distributions lead to optimization conflicts between point cloud completion and other semantic tasks.

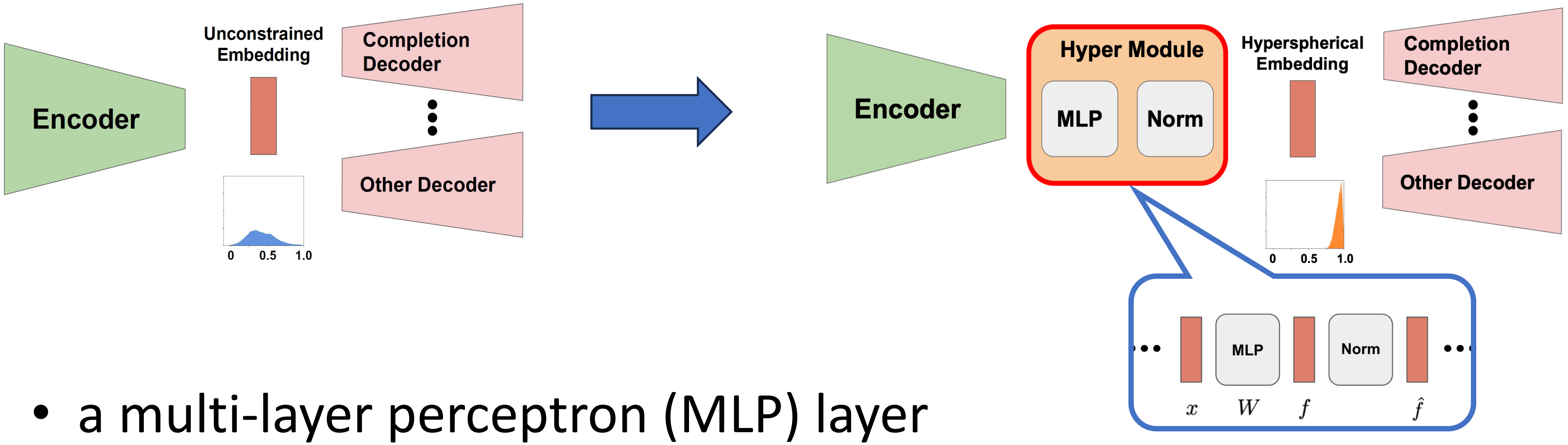
Quick View – Our Solution

✓Propose a **hyperspherical module**, which could be inserted into any existing Encoder-Decoder architecture for point cloud completion.

✓Theoretically analyze the effects of hyperspherical embedding and empirically conduct experiments on several state-of-the-art baselines and datasets.

✓Consistently improve the point cloud completion result in both single-task and multi-task learning.

Hyperspherical Module



- a multi-layer perceptron (MLP) layer
- a normalization layer

$$\hat{f} = \frac{f}{\|f\|_2}$$

$$\|f\|_2 = \sqrt{\sum_i f_i^2}$$

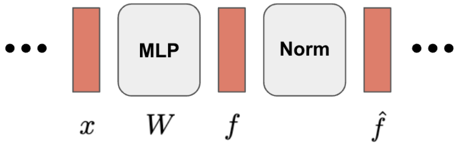
Effects of Hyperspherical Embedding

Proposition 1:

The gradient of the embedding before normalization is orthogonal to itself.

The gradient to the embedding:

$$\frac{\partial L}{\partial f} = \frac{\frac{\partial L}{\partial \hat{f}} - \hat{f}\langle \frac{\partial L}{\partial \hat{f}}, \hat{f} \rangle}{\|f\|_2}$$

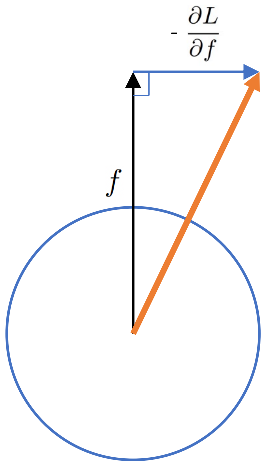


L: Loss function at optimization
f: Embedding before normalization

$$\hat{f} = \frac{f}{\|f\|_2} \qquad \|f\|_2 = \sqrt{\sum_i f_i^2}$$

Orthogonality proof:

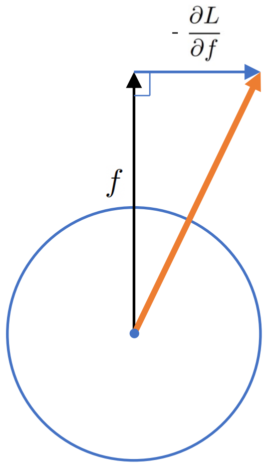
$$\begin{aligned}\langle f, \frac{\partial L}{\partial f} \rangle &= \frac{\langle f, \frac{\partial L}{\partial \hat{f}} \rangle - \langle f, \hat{f} \rangle \langle \frac{\partial L}{\partial \hat{f}}, \hat{f} \rangle}{\|f\|_2} \\ &= \frac{\langle f, \frac{\partial L}{\partial \hat{f}} \rangle - \langle \hat{f}, \hat{f} \rangle \langle \frac{\partial L}{\partial \hat{f}}, f \rangle}{\|f\|_2} \\ &= \frac{\langle f, \frac{\partial L}{\partial \hat{f}} \rangle - \langle \frac{\partial L}{\partial \hat{f}}, f \rangle}{\|f\|_2} \\ &= 0\end{aligned}$$

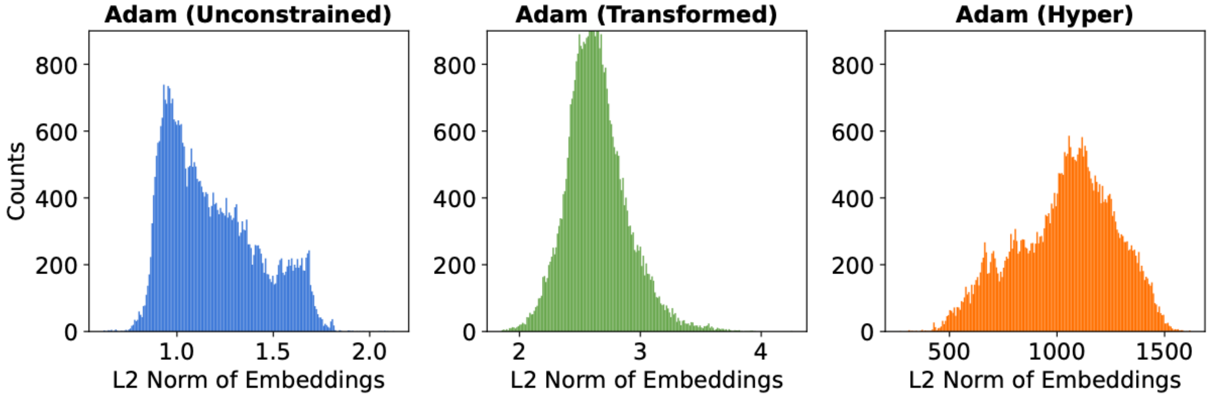


Effects of Hyperspherical Embedding

Proposition 2:

For standard stochastic gradient descent (SGD), the magnitude of the embedding before normalization increases at each update during training.





Norm distribution of embeddings

Effects of Hyperspherical Embedding

Proposition 3:

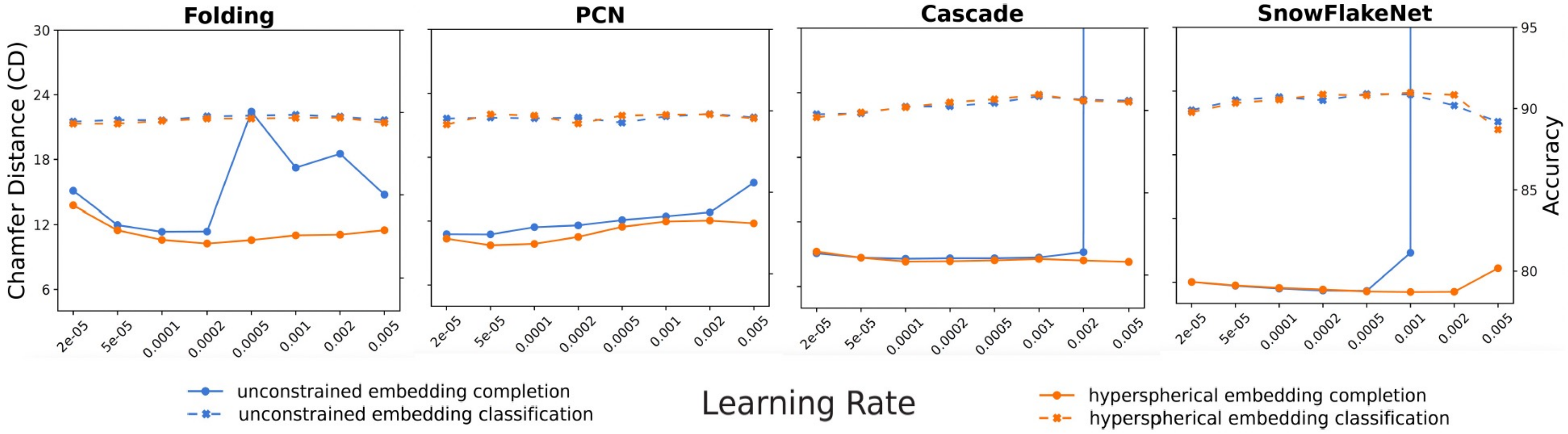
The magnitude of the gradient is inversely proportional to the norm of the embedding.

$$\frac{\partial L}{\partial f} = \frac{\frac{\partial L}{\partial \hat{f}} - \hat{f}\langle \frac{\partial L}{\partial \hat{f}}, \hat{f} \rangle}{\|f\|_2}$$

$$\Downarrow$$

$$\frac{\partial L}{\partial f} \propto \frac{1}{\|f\|_2}$$

Wider range of learning rates:



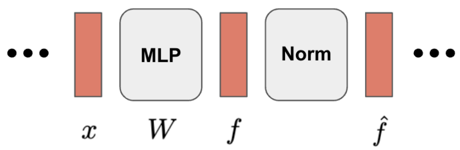
Multi-task learning on MVP dataset with different learning rates

Effects of Hyperspherical Embedding

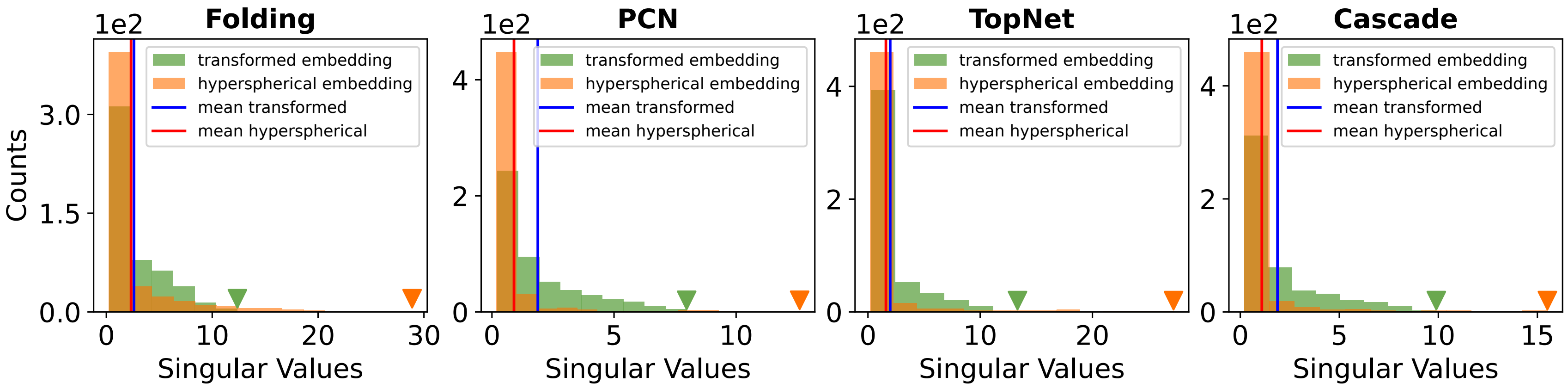
Proposition 4:

During optimization, the increased norm of f requires a poorly conditioned weight matrix.

$$W = U\Sigma V^T$$



W: Weight matrix


Folding

PCN

TopNet

Cascade

Distributions of singular values

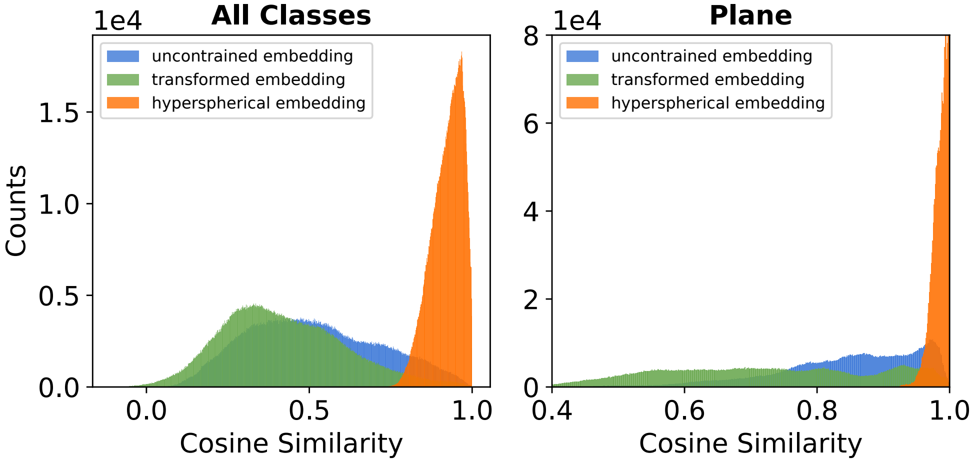
- Large singular values become larger, and small singular values become smaller.

Effects of Hyperspherical Embedding

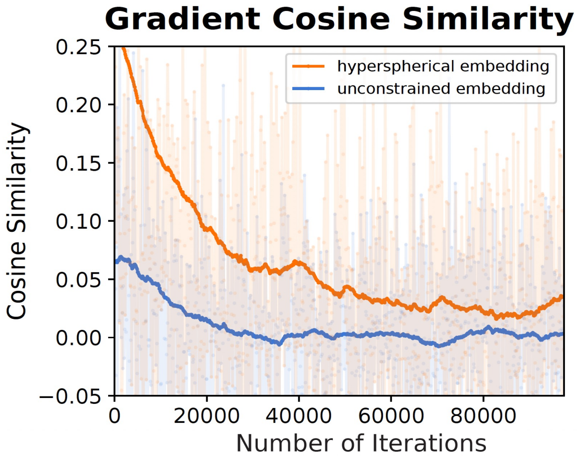
Proposition 4:

During optimization, the increased norm of f requires a poorly conditioned weight matrix.

More compact distribution:

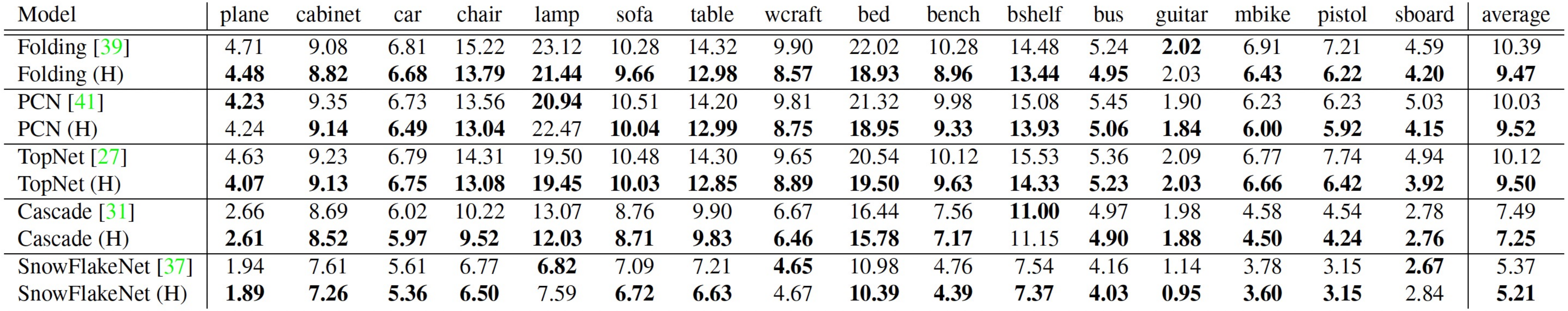


Distributions of cosine similarity in single-task



Gradient conflicts between tasks in multitask learning during training

Single-task Completion

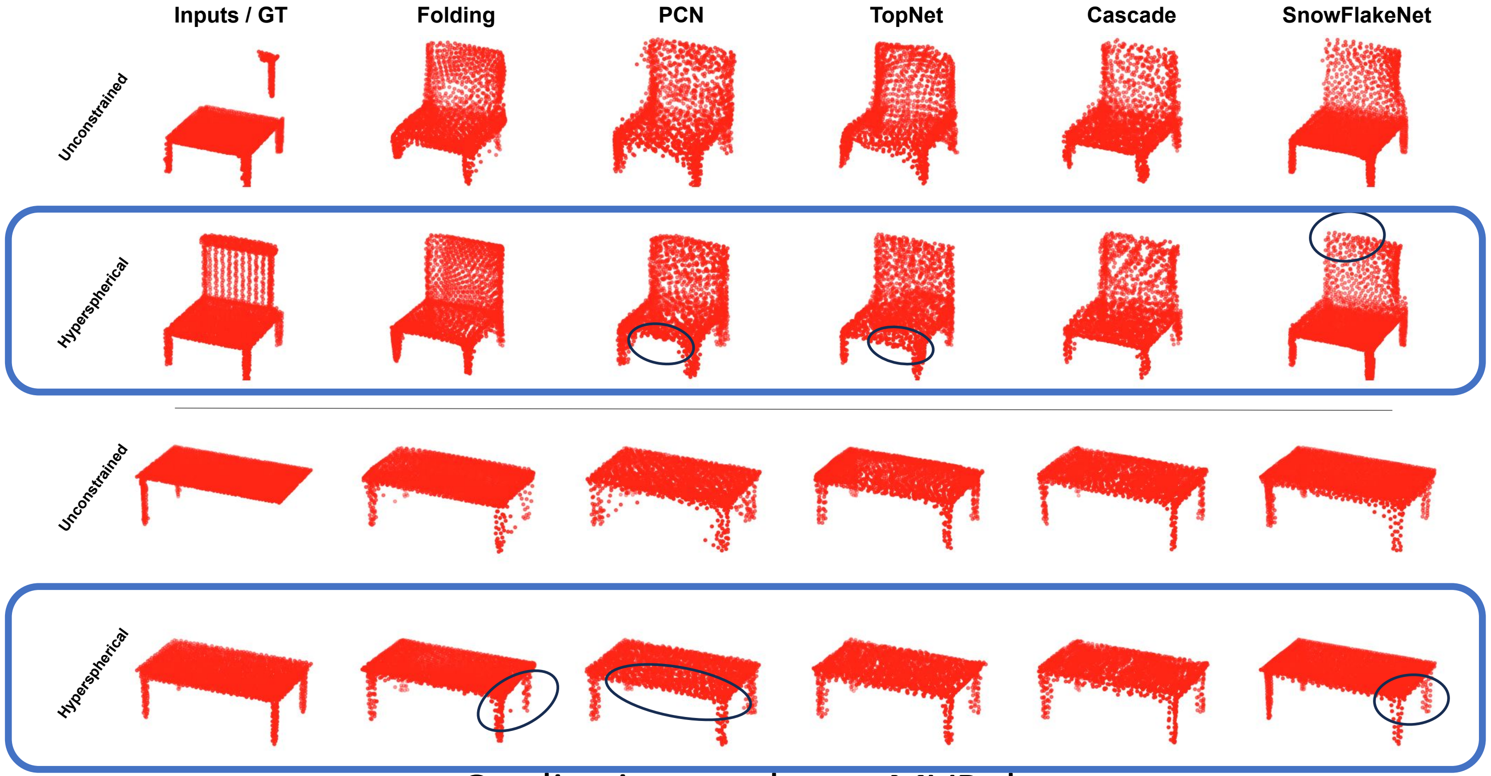


Model	plane	cabinet	car	chair	lamp	sofa	table	wcraft	bed	bench	bshelf	bus	guitar	mbike	pistol	sboard	average
Folding [39]	4.71	9.08	6.81	15.22	23.12	10.28	14.32	9.90	22.02	10.28	14.48	5.24	**2.02**	6.91	7.21	4.59	10.39
Folding (H)	**4.48**	**8.82**	**6.68**	**13.79**	**21.44**	**9.66**	**12.98**	**8.57**	**18.93**	**8.96**	**13.44**	**4.95**	2.03	**6.43**	**6.22**	**4.20**	**9.47**
PCN [41]	**4.23**	9.35	6.73	13.56	**20.94**	10.51	14.20	9.81	21.32	9.98	15.08	5.45	1.90	6.23	6.23	5.03	10.03
PCN (H)	4.24	**9.14**	**6.49**	**13.04**	22.47	**10.04**	**12.99**	**8.75**	**18.95**	**9.33**	**13.93**	**5.06**	**1.84**	**6.00**	**5.92**	**4.15**	**9.52**
TopNet [27]	4.63	9.23	6.79	14.31	19.50	10.48	14.30	9.65	20.54	10.12	15.53	5.36	2.09	6.77	7.74	4.94	10.12
TopNet (H)	**4.07**	**9.13**	**6.75**	**13.08**	**19.45**	**10.03**	**12.85**	**8.89**	**19.50**	**9.63**	**14.33**	**5.23**	**2.03**	**6.66**	**6.42**	**3.92**	**9.50**
Cascade [31]	2.66	8.69	6.02	10.22	13.07	8.76	9.90	6.67	16.44	7.56	**11.00**	4.97	1.98	4.58	4.54	2.78	7.49
Cascade (H)	**2.61**	**8.52**	**5.97**	**9.52**	**12.03**	**8.71**	**9.83**	**6.46**	**15.78**	**7.17**	11.15	**4.90**	**1.88**	**4.50**	**4.24**	**2.76**	**7.25**
SnowFlakeNet [37]	1.94	7.61	5.61	6.77	**6.82**	7.09	7.21	**4.65**	10.98	4.76	7.54	4.16	1.14	3.78	3.15	**2.67**	5.37
SnowFlakeNet (H)	**1.89**	**7.26**	**5.36**	**6.50**	7.59	**6.72**	**6.63**	4.67	**10.39**	**4.39**	**7.37**	**4.03**	**0.95**	**3.60**	**3.15**	2.84	**5.21**

Quantitative Results on MVP dataset (Single-task)

- Consistent improvement for all baseline models with our hyperspherical module.

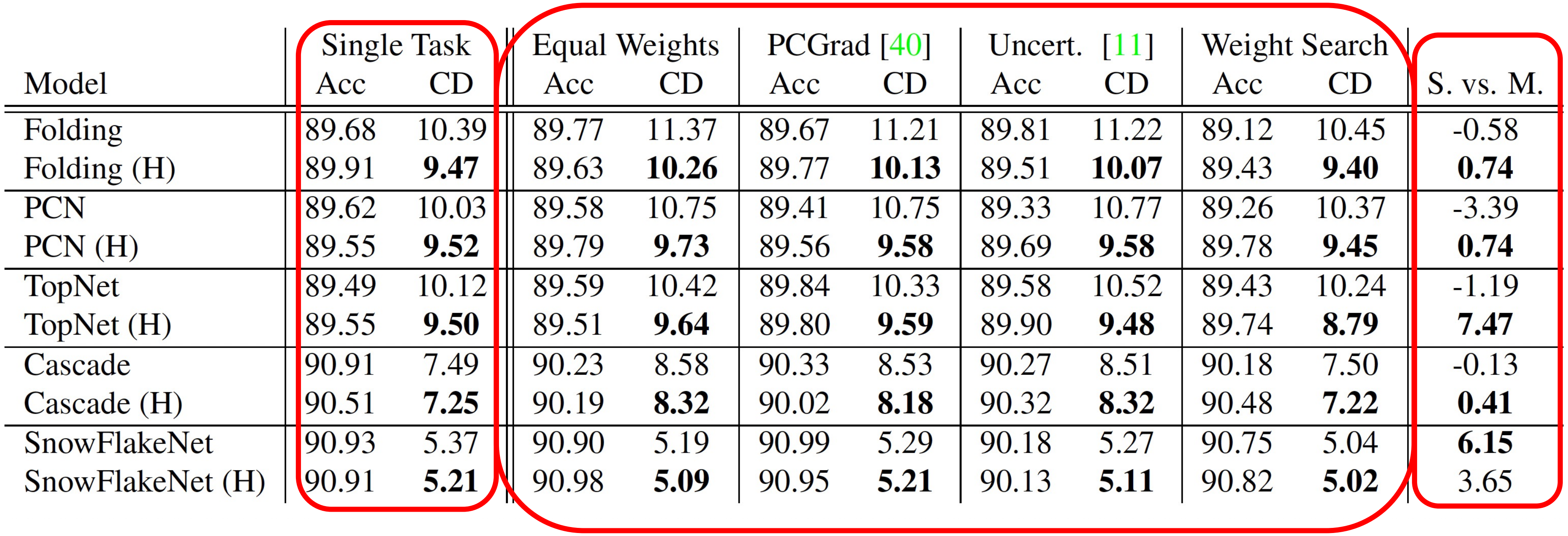
Single-task Completion



Qualitative results on MVP dataset

- With our hyperspherical module, the completion result has less noise.

Multi-task Learning

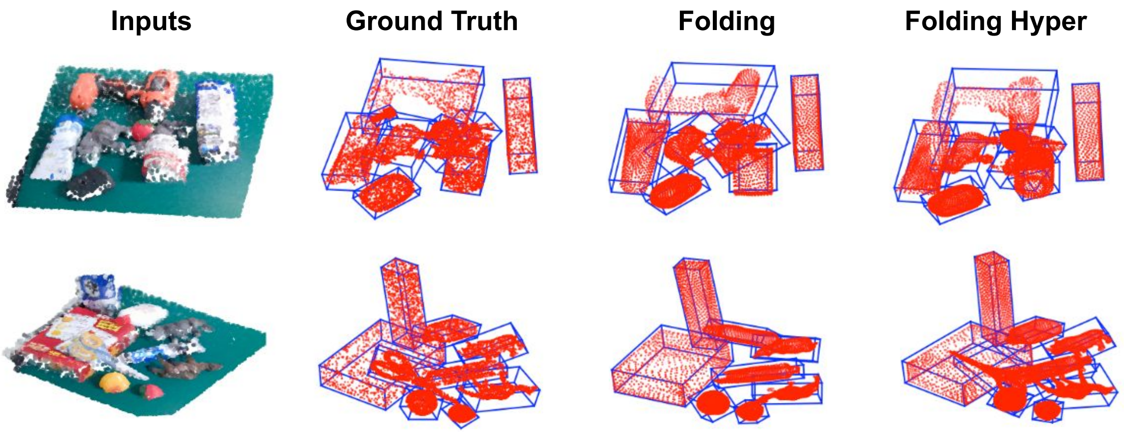


Model	Single Task		Equal Weights		PCGrad [40]		Uncert. [11]		Weight Search		
	Acc	CD	Acc	CD	Acc	CD	Acc	CD	Acc	CD	S. vs. M.
Folding	89.68	10.39	89.77	11.37	89.67	11.21	89.81	11.22	89.12	10.45	-0.58
Folding (H)	89.91	**9.47**	89.63	**10.26**	89.77	**10.13**	89.51	**10.07**	89.43	**9.40**	**0.74**
PCN	89.62	10.03	89.58	10.75	89.41	10.75	89.33	10.77	89.26	10.37	-3.39
PCN (H)	89.55	**9.52**	89.79	**9.73**	89.56	**9.58**	89.69	**9.58**	89.78	**9.45**	**0.74**
TopNet	89.49	10.12	89.59	10.42	89.84	10.33	89.58	10.52	89.43	10.24	-1.19
TopNet (H)	89.55	**9.50**	89.51	**9.64**	89.80	**9.59**	89.90	**9.48**	89.74	**8.79**	**7.47**
Cascade	90.91	7.49	90.23	8.58	90.33	8.53	90.27	8.51	90.18	7.50	-0.13
Cascade (H)	90.51	**7.25**	90.19	**8.32**	90.02	**8.18**	90.32	**8.32**	90.48	**7.22**	**0.41**
SnowFlakeNet	90.93	5.37	90.90	5.19	90.99	5.29	90.18	5.27	90.75	5.04	**6.15**
SnowFlakeNet (H)	90.91	**5.21**	90.98	**5.09**	90.95	**5.21**	90.13	**5.11**	90.82	**5.02**	3.65

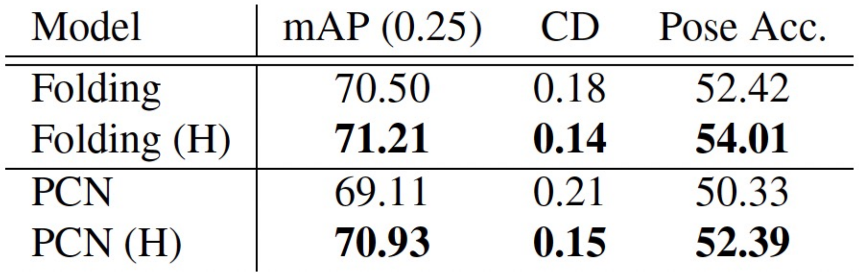
Quantitative results on MVP dataset (Multi-task)

- Consistent improvement on completion task over all multi-task methods.
- Better completion result in multi-task learning than in single-task setting.

Real World Scenarios



Qualitative results on GraspNet dataset

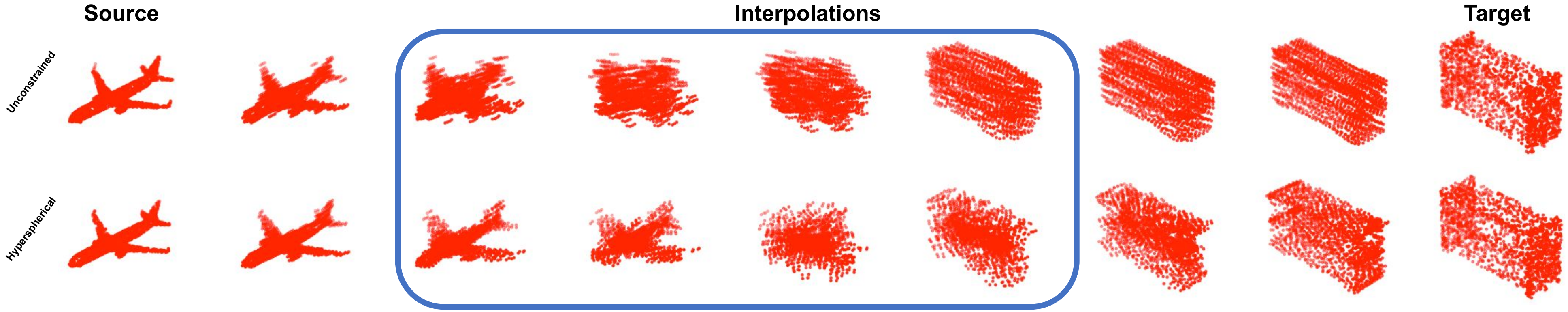


Model	mAP (0.25)	CD	Pose Acc.
Folding	70.50	0.18	52.42
Folding (H)	**71.21**	**0.14**	**54.01**
PCN	69.11	0.21	50.33
PCN (H)	**70.93**	**0.15**	**52.39**

Quantitative results on GraspNet dataset

- The proposed hyperspherical module bring consistent improvement over object detection, pose estimation and point cloud completion.

Visualize the Embedding Space



Point cloud interpolation samples in the embedding space

- The generated point clouds with hyperspherical embeddings have more clear clues from source to target shapes.

Conclusion

- We propose a **hyperspherical module** that outputs hyperspherical embeddings, which improves the performance of point cloud completion.
- We theoretically investigate the effects of hyperspherical embeddings and demonstrate that the point cloud completion benefits from them by stable training and learning a compact embedding distribution.
- We analyze training point cloud completion with other tasks and observe conflicts between them, which can be reconciled by the hyperspherical embedding.