109-2 3DCVDL final project

Template is all you need: 2D to 3D reconstruction with template learned by contrastive learning

R08945027 M.Y. HO (何明洋)
R09942089 Y.S. Haung (黃郁珊)
B06201018 C.J. Chang (張芷榕)

Code available at: https://github.com/Kaminyou/Template-is-all-you-need

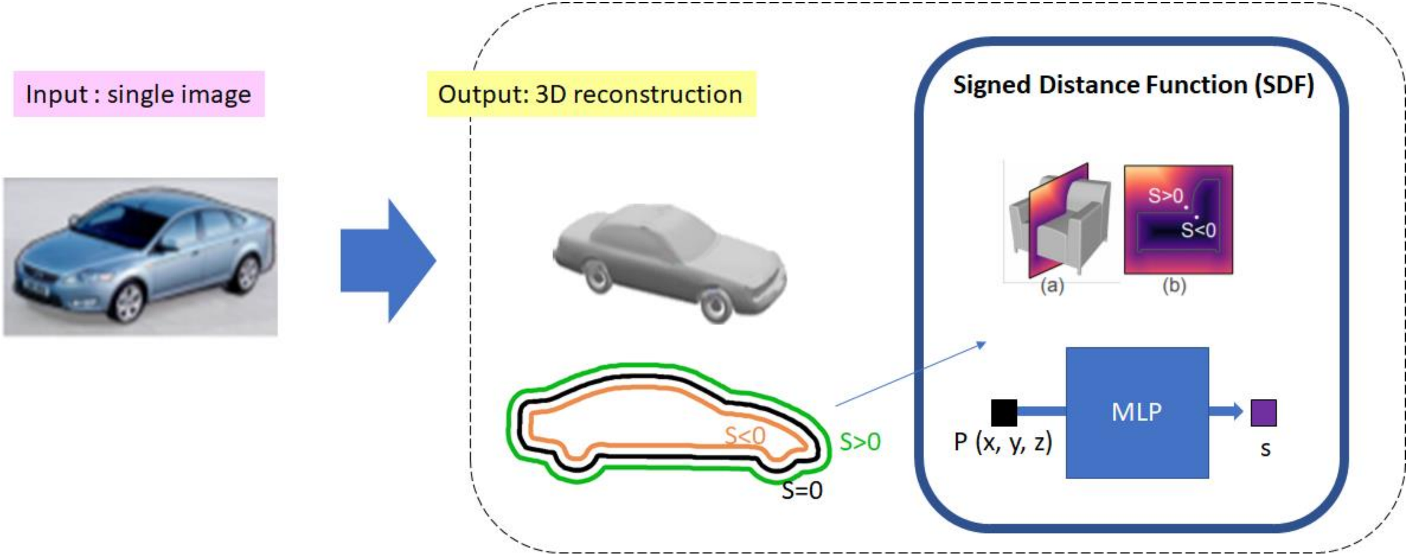
Outline

- Introduction
- Method
  - Dataset and Preprocessing
  - Training scheme and architecture
- Result
- Discussion
- Conclusion
- Work Distribution

Introduction

Goal 1: 2D to 3D reconstruction

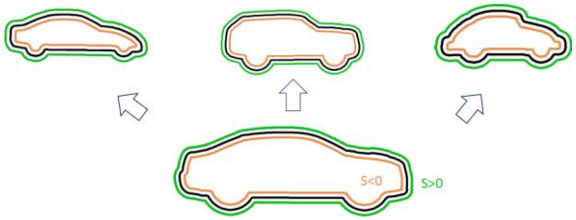
We mainly focus on testing set and other unseen data.

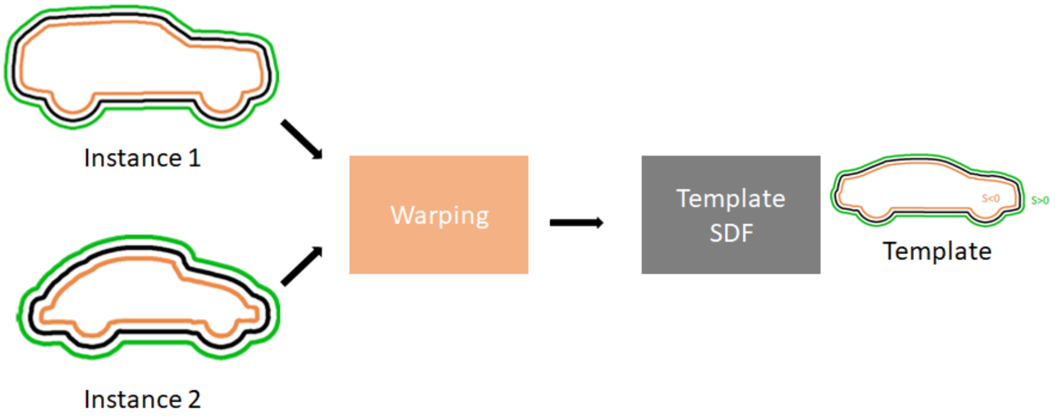


Introduction

Goal 2: Find out the template in each category.

Relationship between template and instances

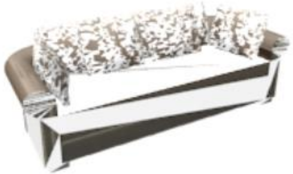




Template can help find common characteristics from instances belong to the same category.

Method - Dataset and Preprocessing

Plane



Sofa

Chair

Method - Dataset and Preprocessing

ShapeNet provides 3D object in mesh **OBJ** format

→ **2D images** in PNG format w/ 50 easy and 50 hard views (training input)
224*224*3 (RGB)

→ **3D object** in SDF format (training ground truth)
N*4 (x,y,z, sdf)

→ **3D object** in PLY format (for evaluation)

Method - training scheme and architecture

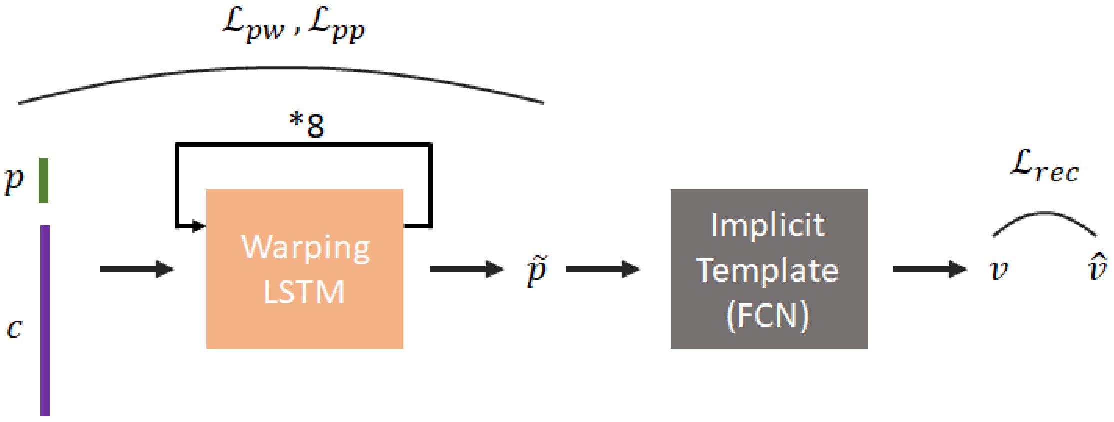
Phase 1: Train for embedding **Get an unique embedding for each instance**

$$\mathcal{L}_{rec}^{(s)} = \sum_{k=1}^{K} \sum_{i=1}^{N} L_{\epsilon_s, \lambda_s}\left(\mathcal{T}\left(\boldsymbol{p}^{(s)}\right), v_{k,i}\right)$$

$$\mathcal{L}_{rec} = \sum_{s \in \{2,4,6,8\}} \mathcal{L}_{rec}^{(s)}$$

$$\mathcal{L}_{pw} = \sum_{k=1}^{K} \sum_{i=1}^{N} h\left(\|\mathcal{W}(\boldsymbol{p}_i, \boldsymbol{c}_k) - \boldsymbol{p}_i\|_2\right)$$

$$\mathcal{L}_{pp} = \sum_{k=1}^{K} \sum_{i \neq j} \max\left(\frac{\|\Delta \boldsymbol{p}_i - \Delta \boldsymbol{p}_j\|_2}{\|\boldsymbol{p}_i - \boldsymbol{p}_j\|_2} - \epsilon, 0\right)$$



p denotes sdf samples' x, y, z coordinates in R^3

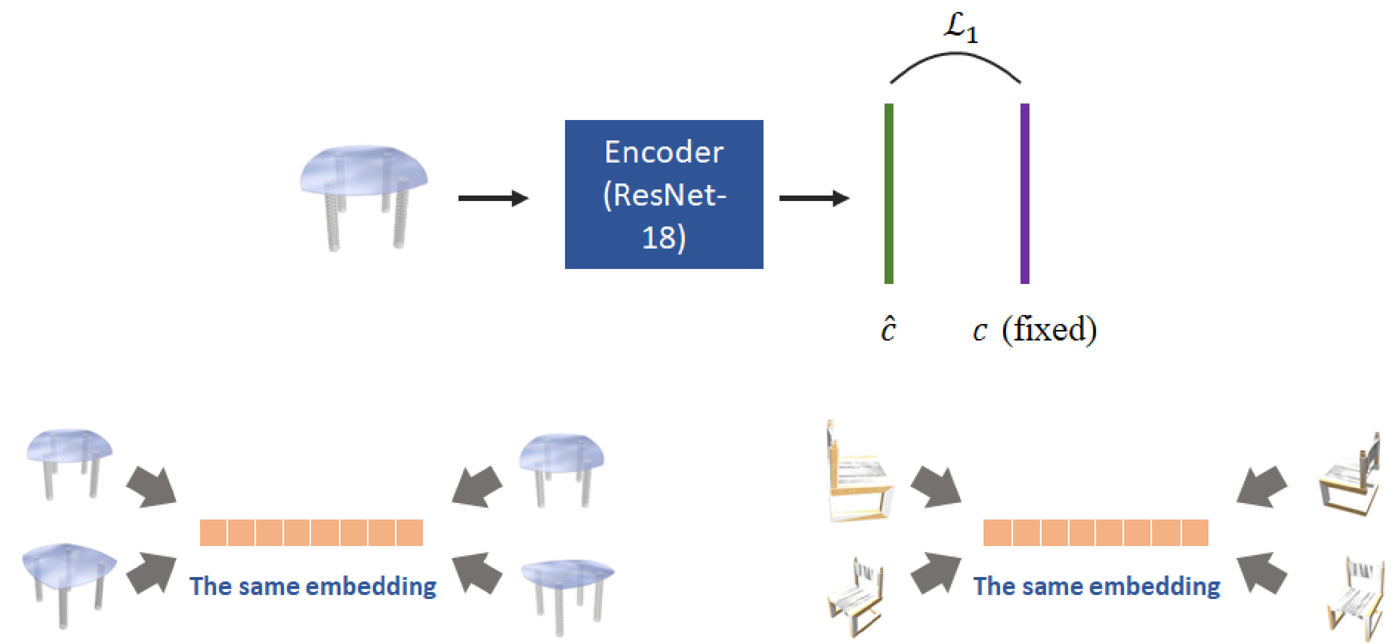
v denotes sdf samples' corresponding sdf value in R^1

c denotes trainable embedding in R^{256}

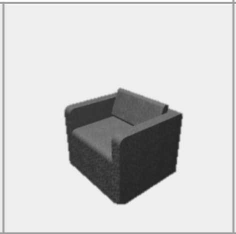
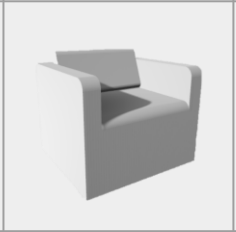
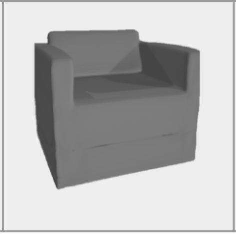
Method - training scheme and architecture

Phase 2: Train for encoder

Encode the images to corresponding embeddings



Result - chairs

	Train		Test		Real world
rgb					
gt					NA
3d					

Chamfer distance (1000x)

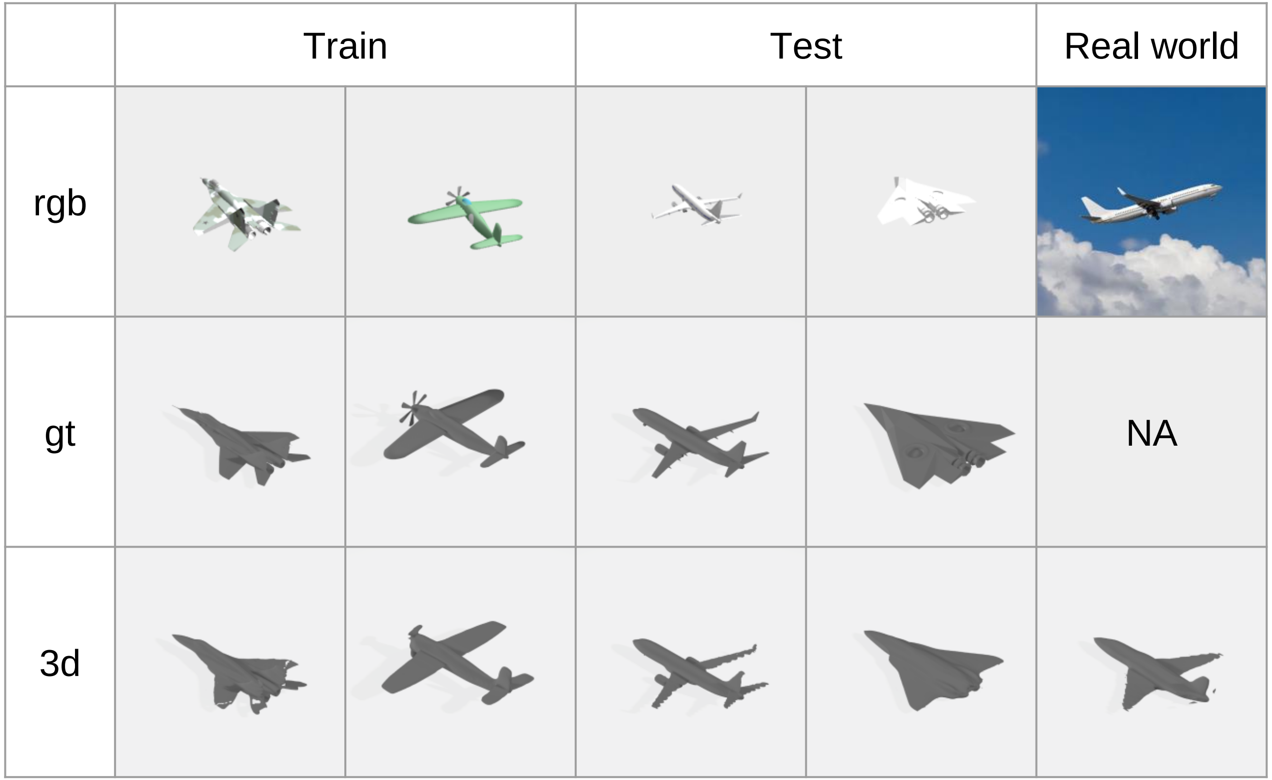
	Mean	Median
3D CNN	11.15	NA
DISN	7.54	NA
Pixel2mesh	11.13	NA
Ours (fix)	1.61	1.02
Ours (rnd)	1.82	1.16

Result - sofas

Chamfer distance (1000x)

	Mean	Median
3D CNN	9.76	NA
DISN	8.71	NA
Pixel2mesh	6.54	NA
Ours (fix)	1.50	0.72
Ours (rnd)	1.49	0.63

Result - planes



Chamfer distance (1000x)

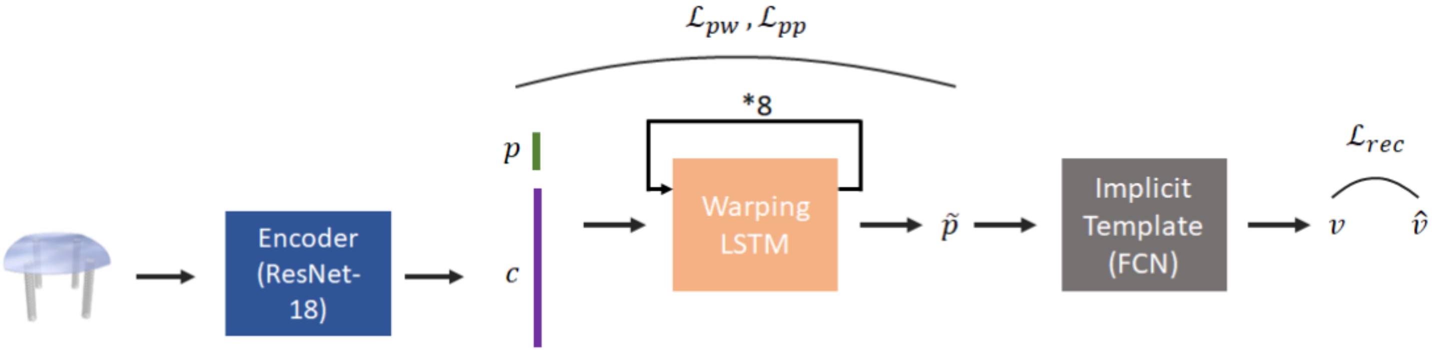
	Mean	Median
3D CNN	10.47	NA
DISN	9.01	NA
Pixel2mesh	6.10	NA
Ours (fix)	4.19	0.16
Ours (rnd)	4.52	0.18

Discussion - what about end-to-end training?

The previous method relies on **intermediary embeddings** in the first phase.

- Final results hugely depend on the quality of the embeddings.
- The **number of parameters scales linearly** with the number of training data.

→ Try training in an **end-to-end** manner.



Discussion - what about end-to-end training?

How does the encoder learn embeddings that are view-invariant?

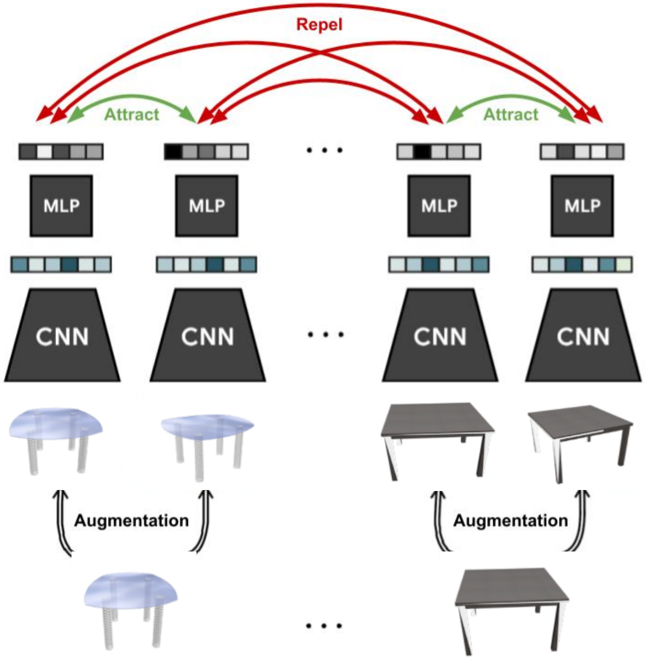
Sol 1	Sol 2
L1 loss to match the pretrained embeddings’ distribution.	Implicitly learned from the supervised loss function
strong	weak

→ Contrastive learning may come in handy!

Discussion - what about end-to-end training?

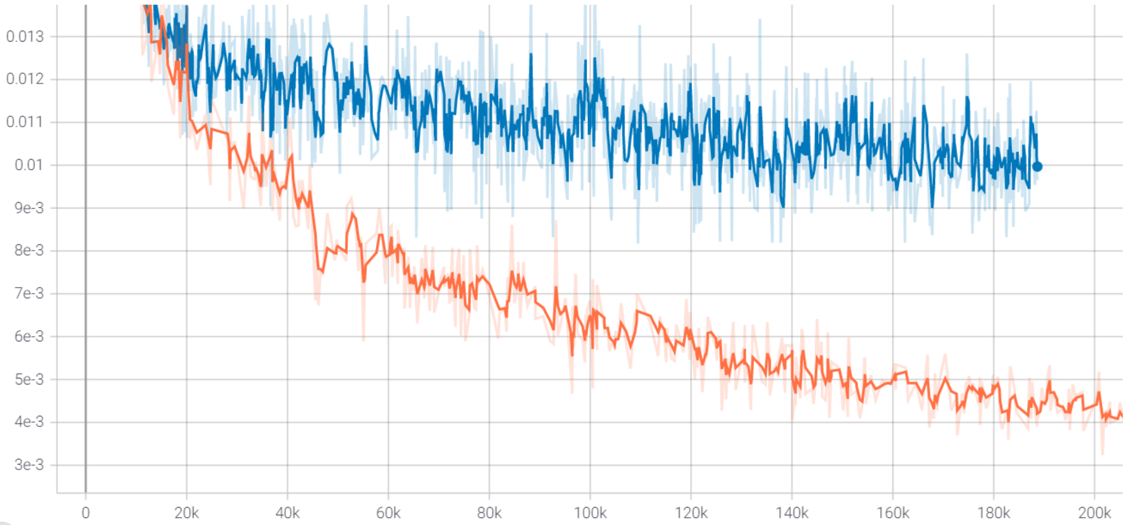
- Force the **different views** of the same instance to be **closer in latent space**.

- Pretrain the encoder with **contrastive learning** and then use it **as the initial weights** for the later end-to-end training



Discussion - what about end-to-end training?

Training loss

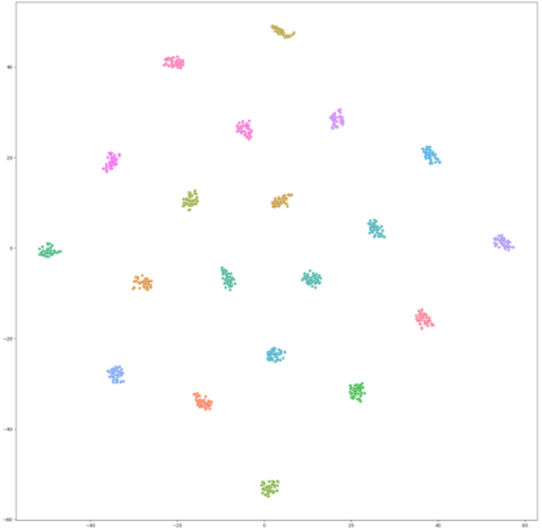


ImageNet pretrained

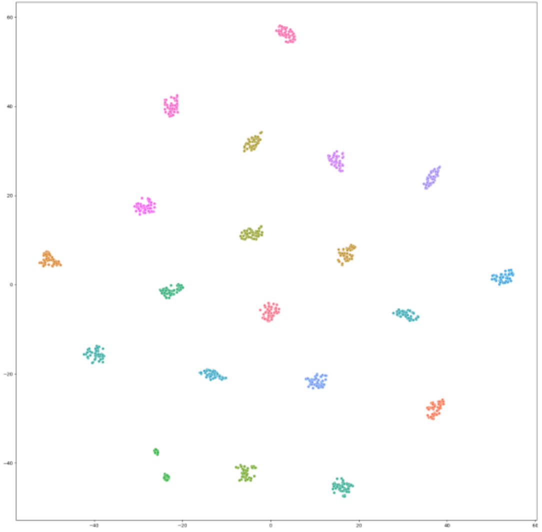
contrastive learning

Discussion - what about end-to-end training?

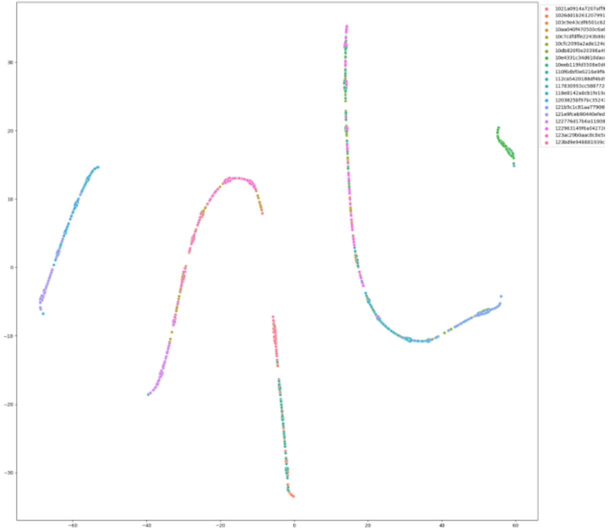
Sol 1



Sol 2
w/ contrastive learning



Sol 2
w/o contrastive learning



Discussion - what about end-to-end training?

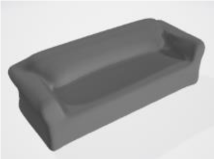
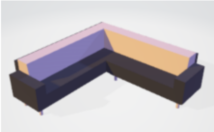
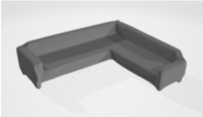
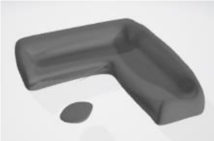
	rgb	gt	Sol 1	Sol 2
train				
test				

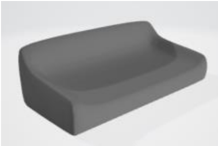
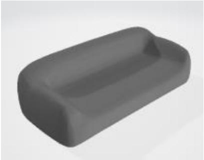
Sol 1 template	Sol 2 template
	

Chamfer distance (1000x)

	Mean	Median
Sol 1 (fix)	1.61	1.02
Sol 1 (rnd)	1.82	1.16
Sol 2 (fix)	1.53	0.93
Sol 2 (rnd)	1.59	1.00

Discussion - what about end-to-end training?

	rgb	gt	Sol 1	Sol 2
train				
test				

Sol 1 template	Sol 2 template
	

Chamfer distance (1000x)

	Mean	Median
Sol 1 (fix)	1.50	0.72
Sol 1 (rnd)	1.49	0.63
Sol 2 (fix)	2.01	1.40
Sol 2 (rnd)	2.54	1.26

Discussion - what about end-to-end training?

Sol 1 template	Sol 2 template
	

	rgb	gt	Sol 1	Sol 2
train				
test				

Chamfer distance (1000x)

	Mean	Median
Sol 1 (fix)	4.19	0.16
Sol 1 (rnd)	4.52	0.18
Sol 2 (fix)	1.76	0.29
Sol 2 (rnd)	1.75	0.28

Conclusion

What we have done:

- Fulfill the 2D to 3D reconstruction task.
- Find out the template in each category.
- The unseen images can be applied to our network and return good results.

Novelty:

- Leverage the concept of template to achieve 2D to 3D reconstruction.
- Improve the feature extraction from 2D image using contrastive learning.

Work Distribution

何明洋

- Be in charge of “sofa” class
- Data preprocessing
- Propose solution 1
- Embedding analysis and result evaluation

黃郁珊

- Be in charge of “chair” class
- Propose the main ideas
- Implement warping and SDF submodules
- Implement solution 1

張芷榕

- Be in charge of “plane” class
- Propose solution 2
- Implement contrastive learning
- Implement result generation function

Q & A