Category-Level 6D Object Pose Estimation in the Wild: A Semi-Supervised Learning Approach and A New Dataset

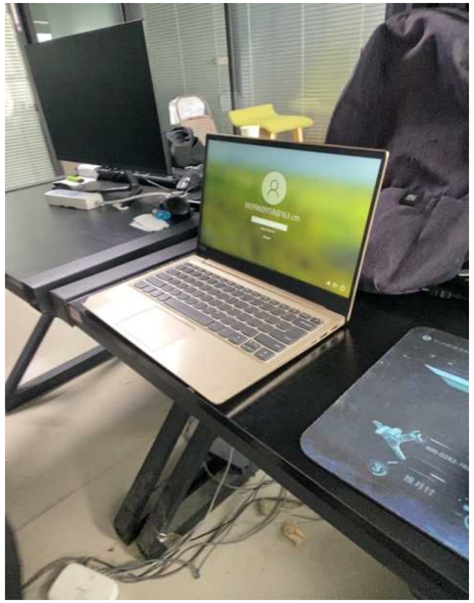
Yang Fu and Xiaolong Wang

UC San Diego

Task: Category-level Pose Estimation

Input

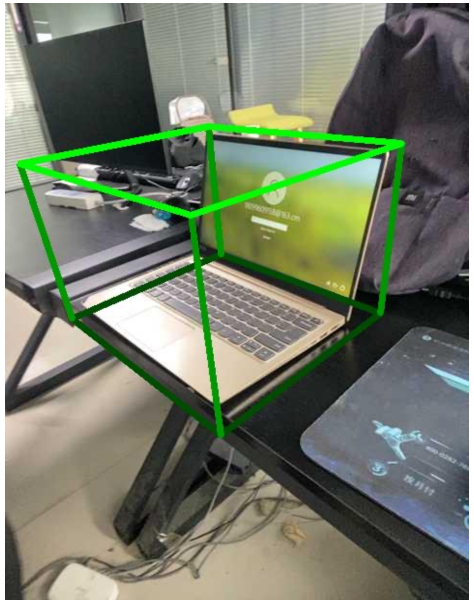
RGB Image



Depth Point Cloud

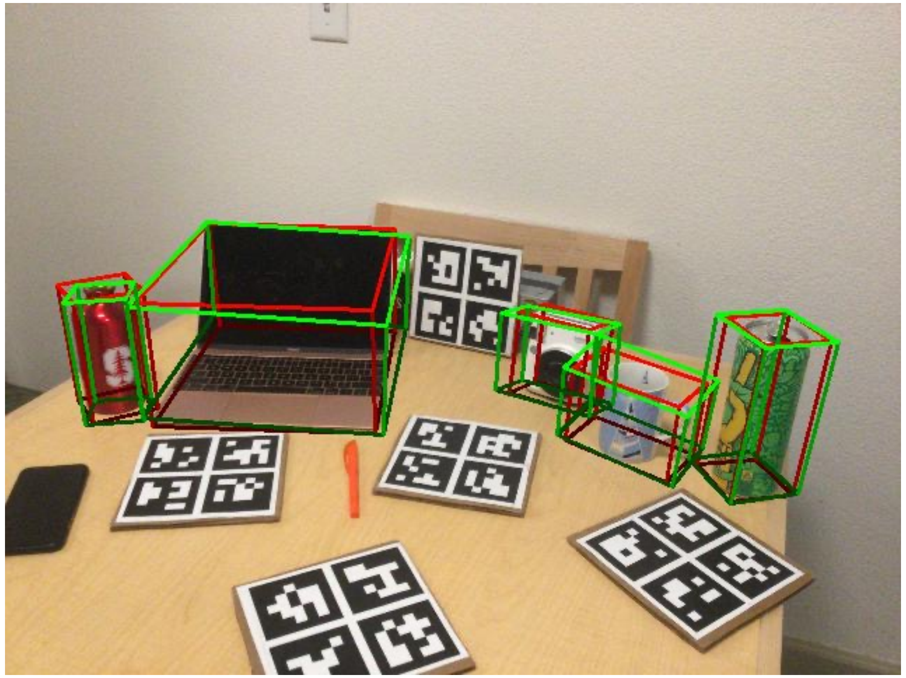
Output:

3D Rotation + 3D Translation + 3D Size



Generalize to **different object instances** within the **same category**

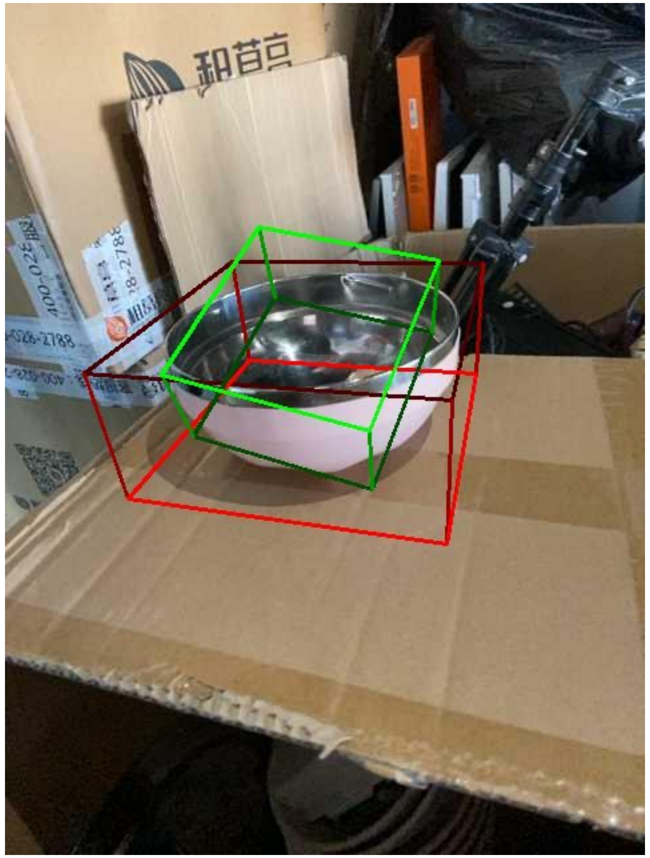
Results: Pose Estimation on NOCS-REAL275

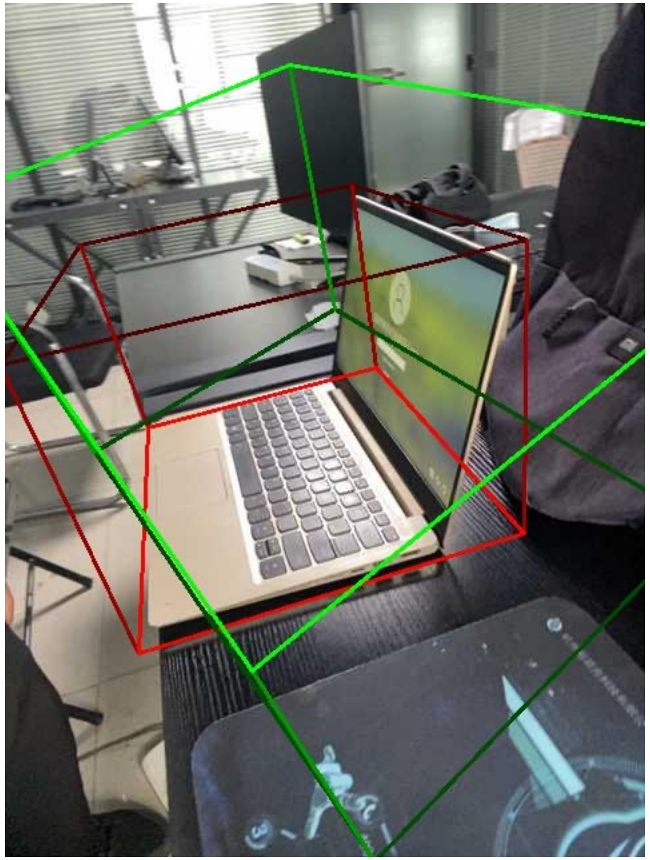


Red box indicates the ground-truth pose, Green indicates the predicted one

Testing on Real-world scenes…

Poor Generalization Ability

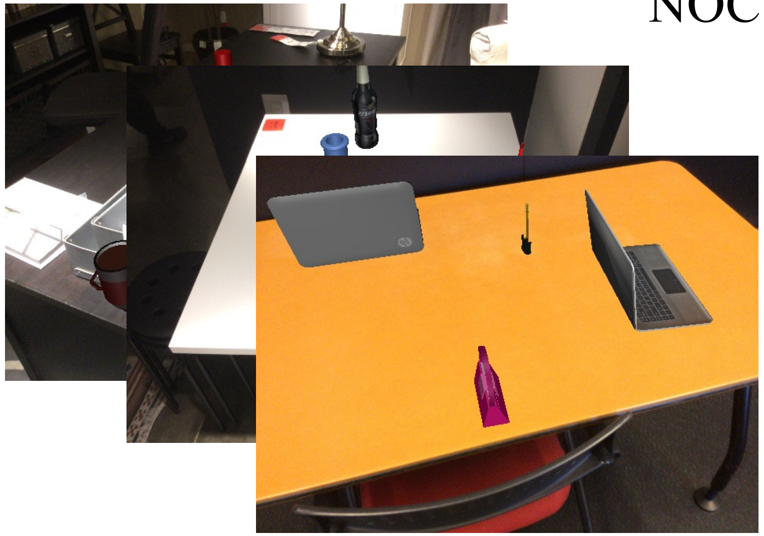




Adopting the NOCS pre-trained model on real-world images always leads poor results

Red box indicates the ground-truth pose, Green indicates the predicted one

Few Real Data



NOCS-CAMER75

NOCS-REAL275

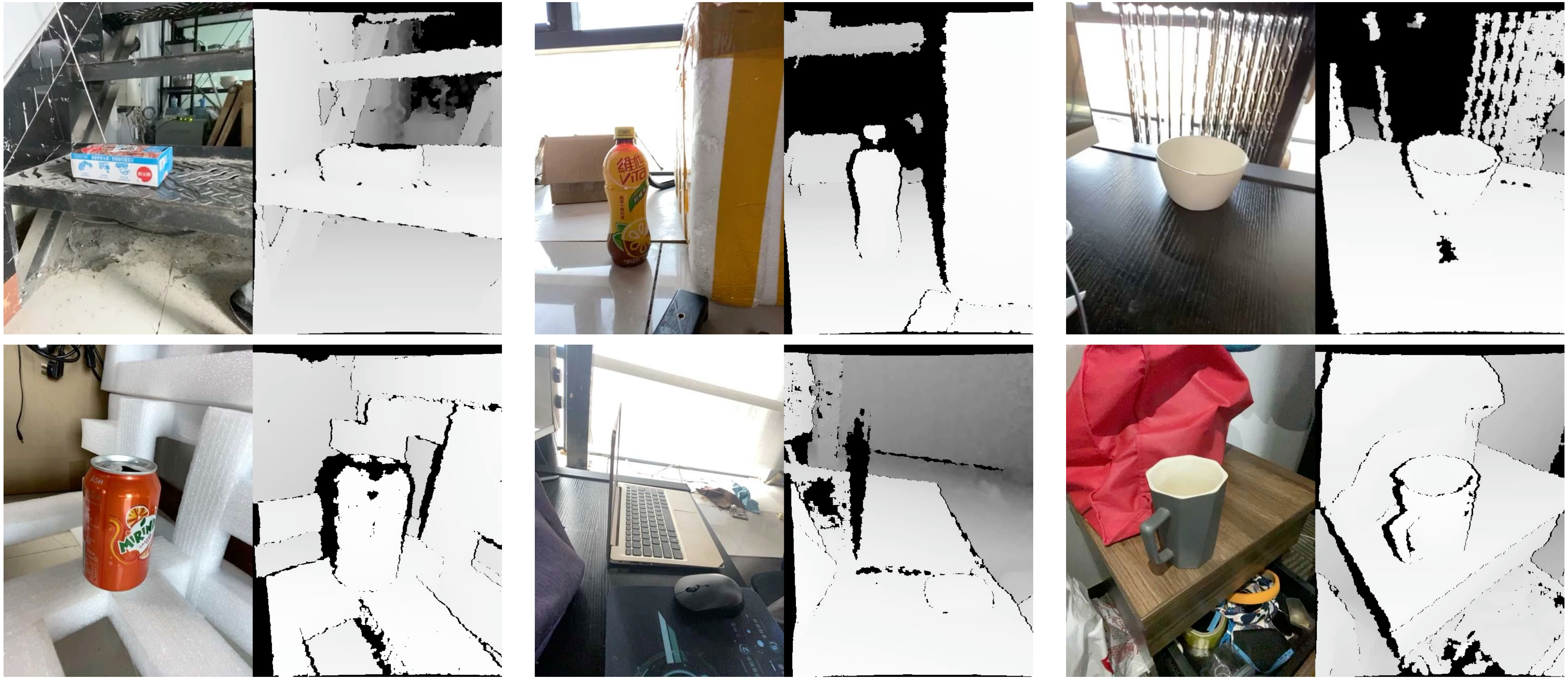
NOCS-REAL275 only contains 7,000 images under 13 scenes.

New RGBD Video Dataset --- Wild6D

Wild6D Data Collection

- Recording with iPhone or iPad.
- More than 5,000 RGBD videos across 1,700 objects (>1.1 million images).
- Only annotate few key frames and track for the remaining frames.
- No annotations for training videos.

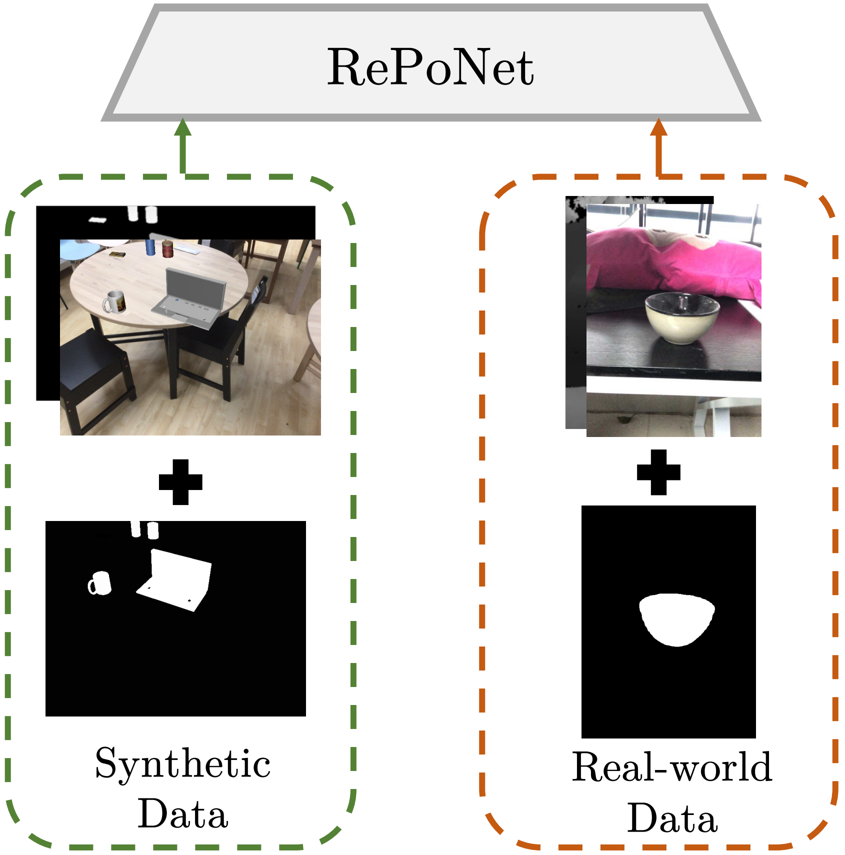
Wild6D Data Samples



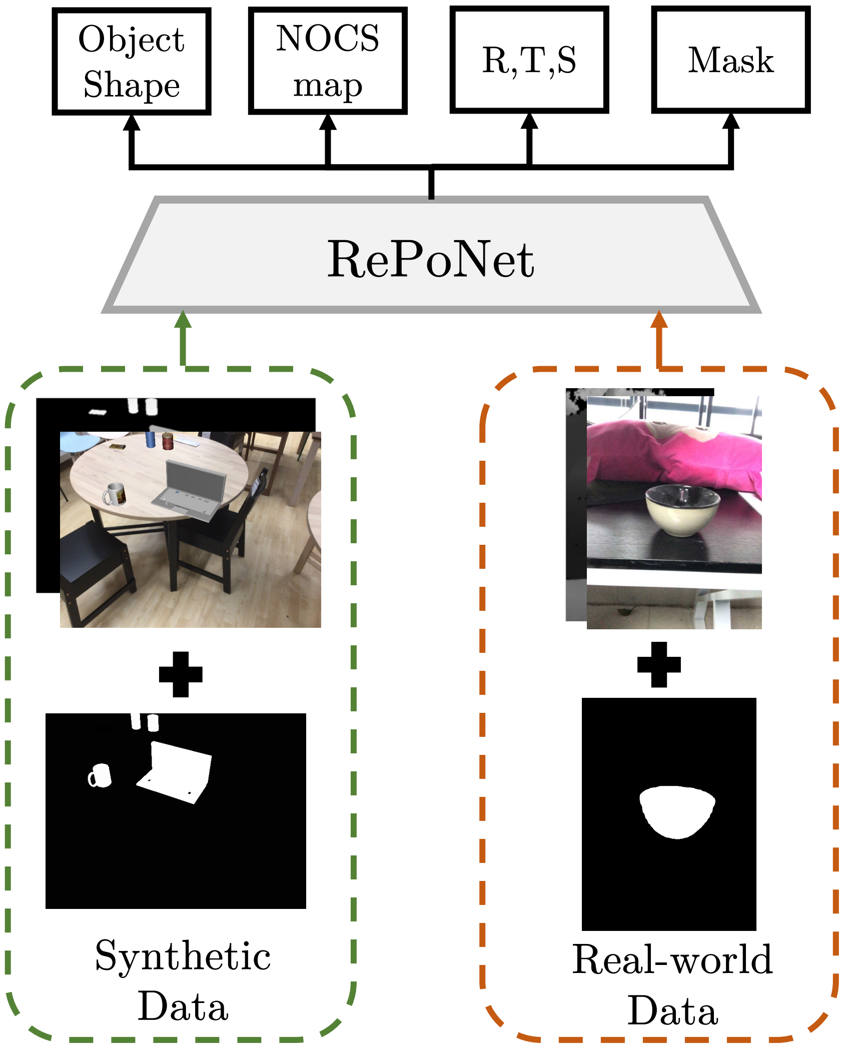
Comparison with existing data

Datasets	RGBD	Real	#Categories	#instances	#images
Objectron [1]		✓	9	18K	4M
CAMERA25 [47]	✓		6	184	300K
REAL275 [47]	✓	✓	6	24	8k
Wild6D	✓	✓	5	1.8K	1M

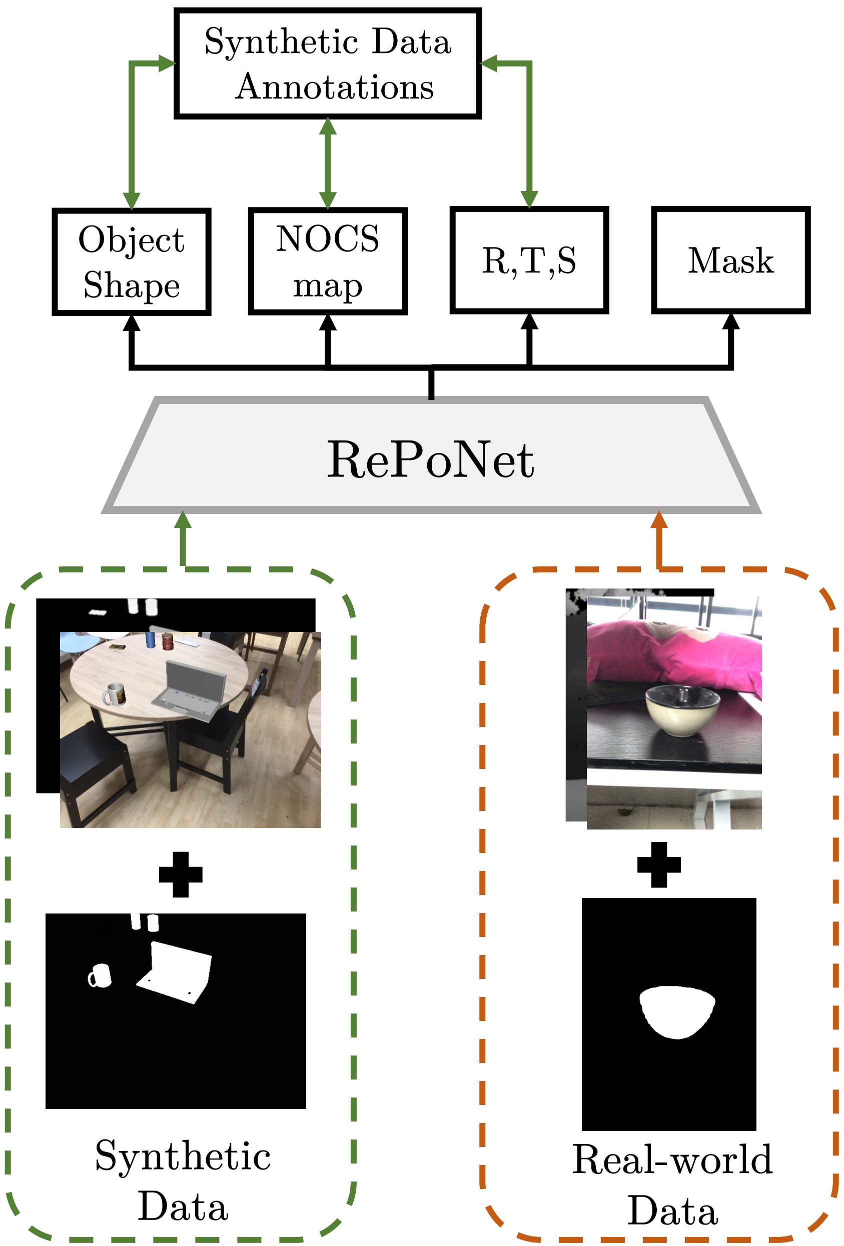
Leveraging Wild6D Data…



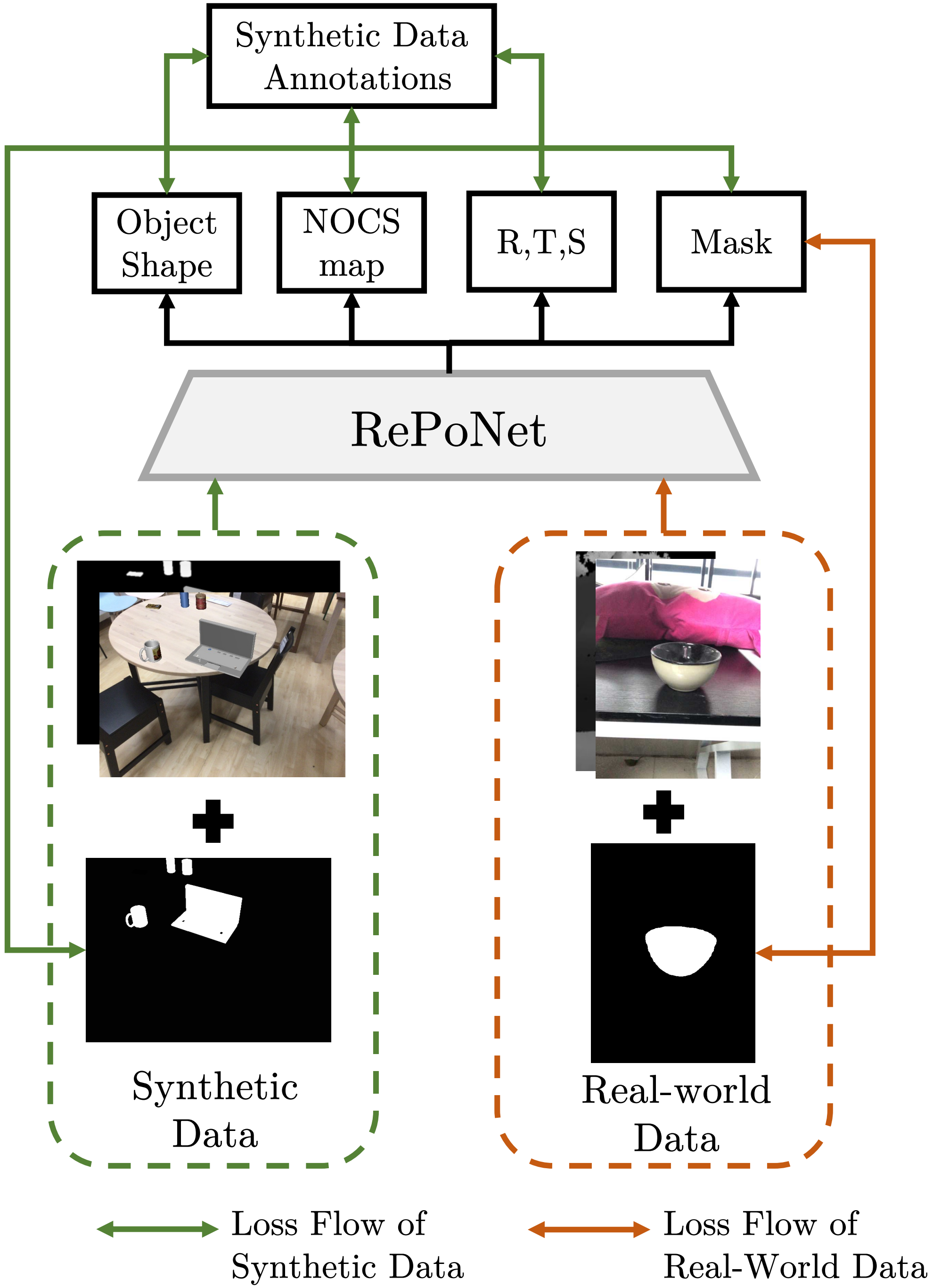
1. Input two sets of data: synthetic data and real-world data



1. Input two sets of data: synthetic data and real-world data
2. Estimate the object shape, object pose, NOCS map and rendered mask.



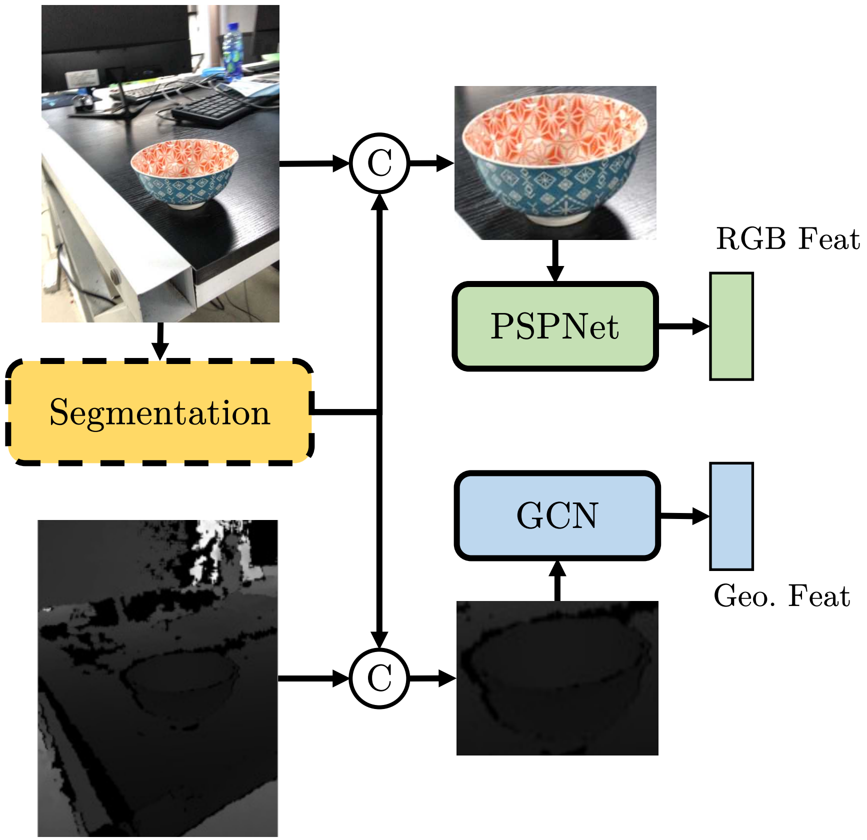
1. Input two sets of data: synthetic data and real-world data
2. Estimate the object shape, object pose, NOCS map and rendered mask.
3. Supervise the prediction of synthetic data by its annotations.



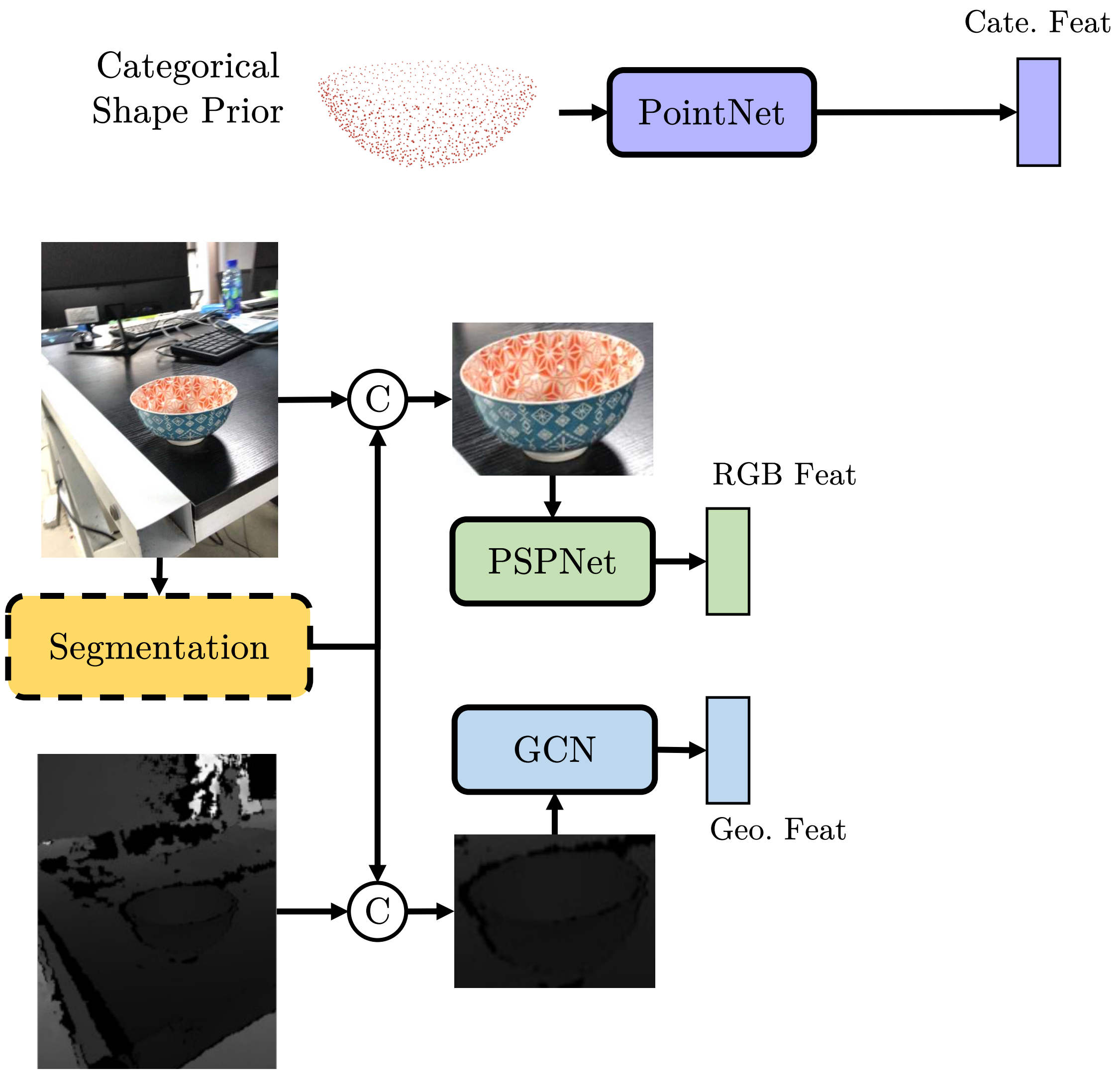
1. Input two sets of data: synthetic data and real-world data
2. Estimate the object shape, object pose, NOCS map and rendered mask.
3. Supervise the prediction of synthetic data by its annotations.
4. Optimize the rendered mask by the input foreground mask for both synthetic data and real-world data

The **estimated pose** and **reconstructed shape** can be jointly optimized in a **self-supervised manner**.

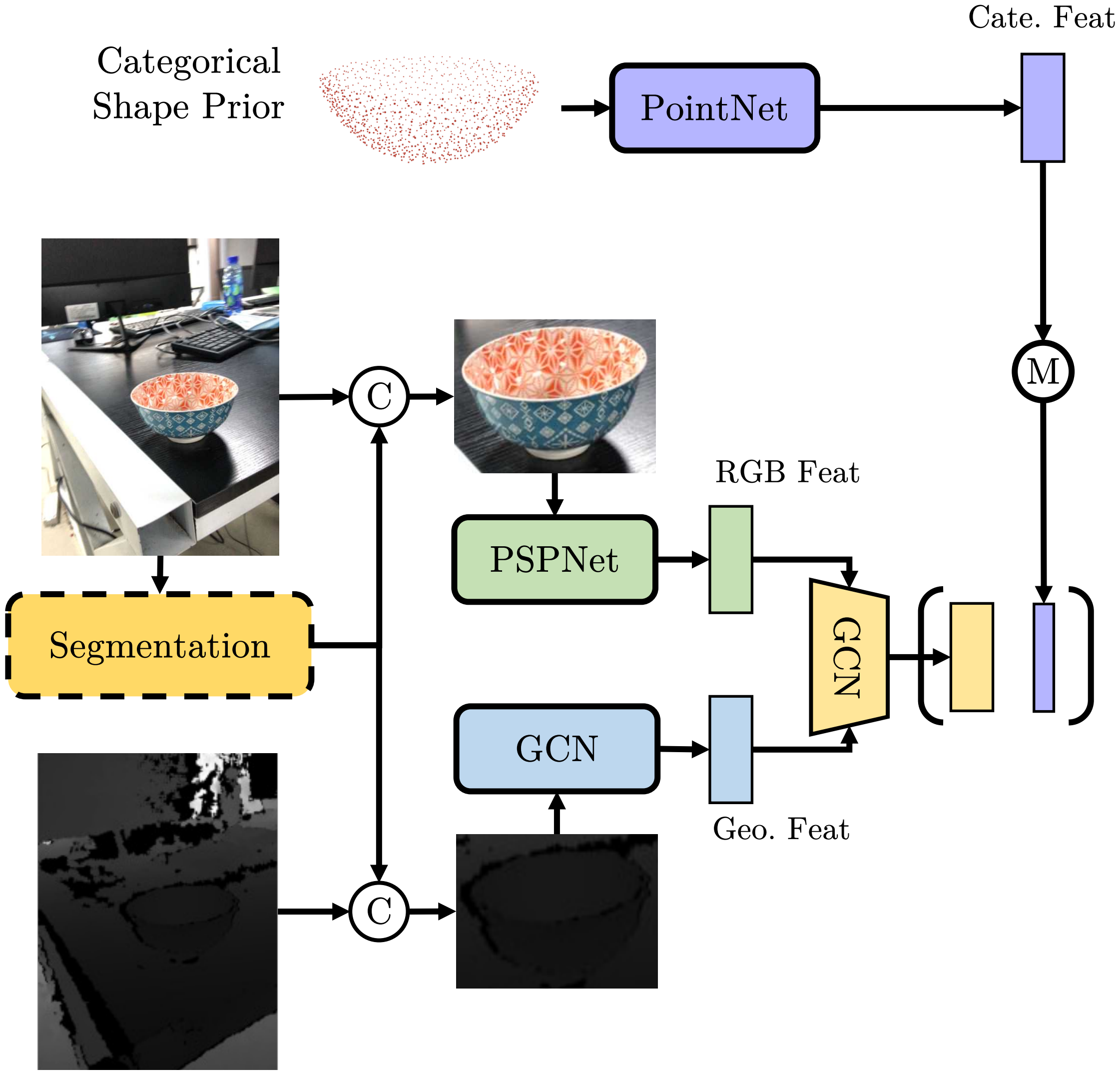
The proposed approach -- RePoNet



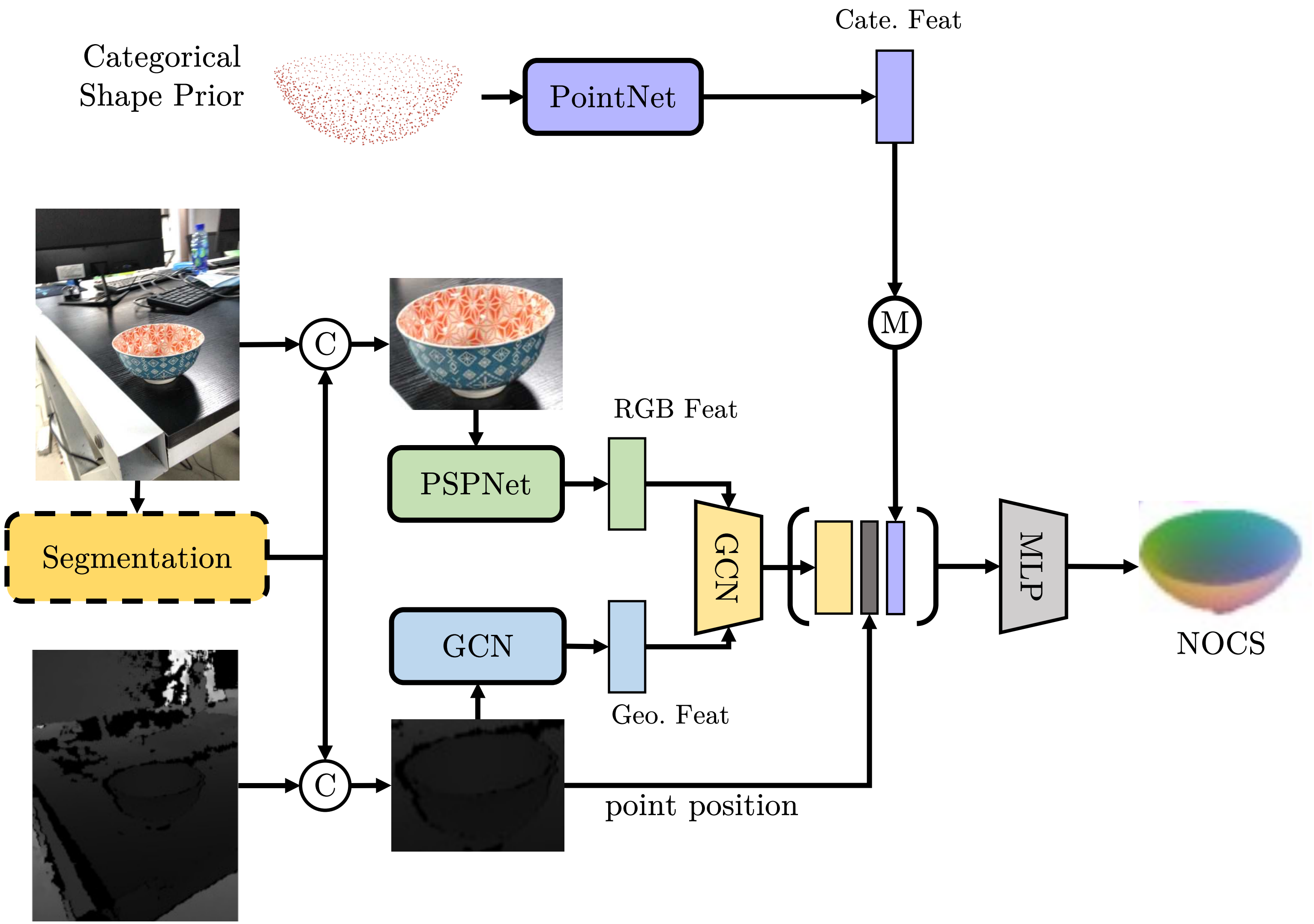
1. Crop RGBD images according to the foreground mask and extract the RGB feature and geometry feature.



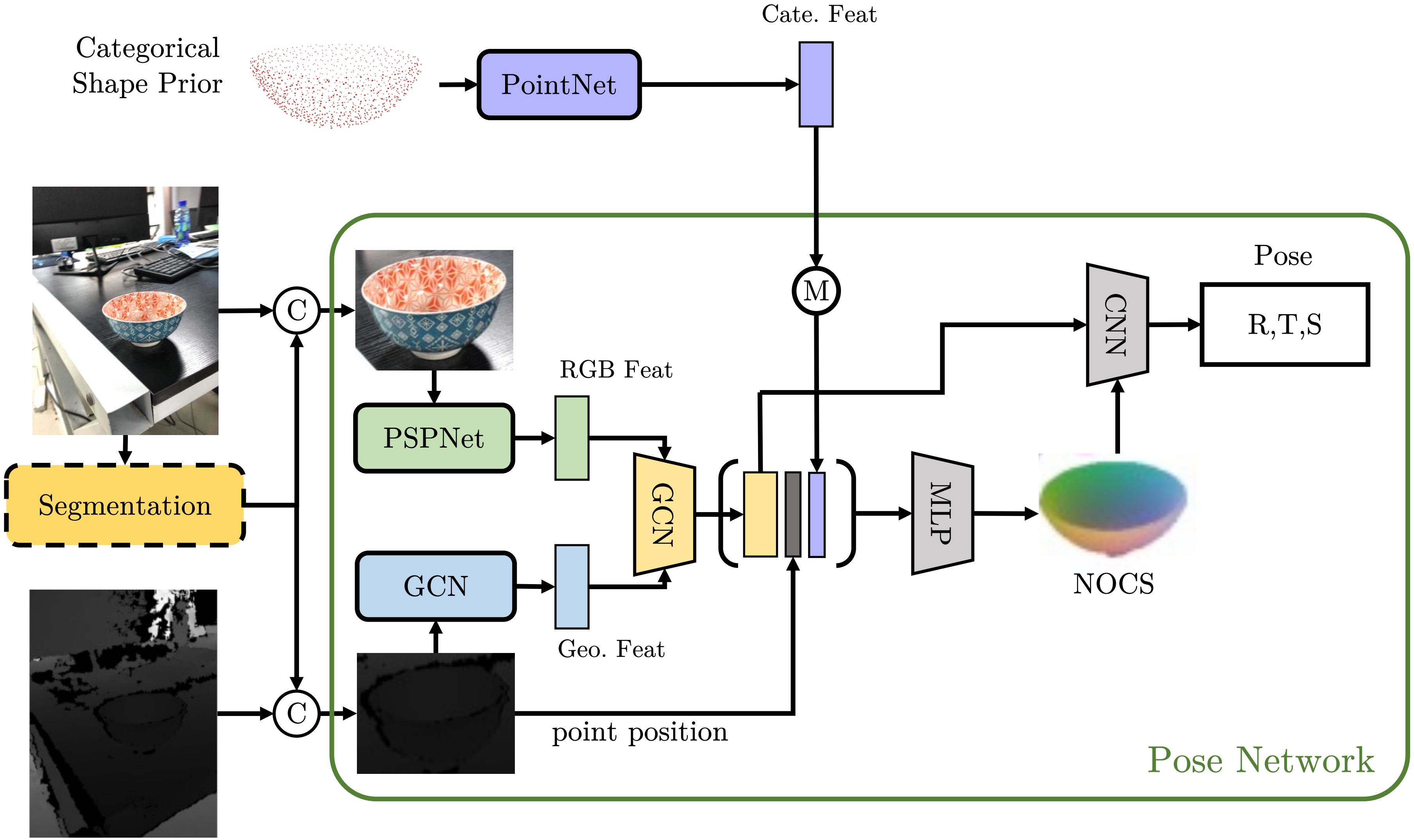
2. Define a categorical shape prior and extract its feature via PointNet.



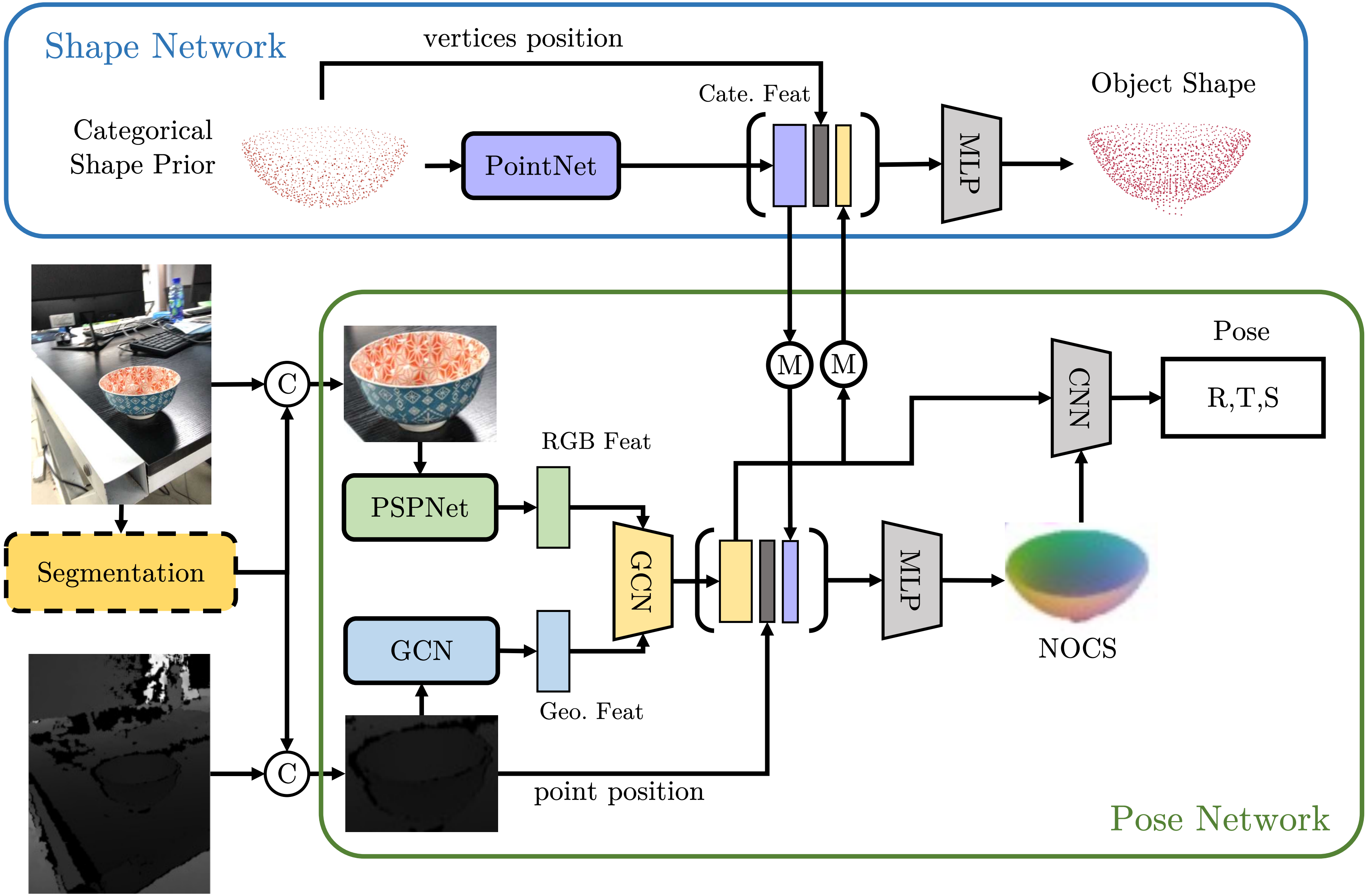
3. For each RGBD image, aggregate its RGB feature and geometry feature via GCNs as the RGBD feature and concatenate it with the categorical feature.



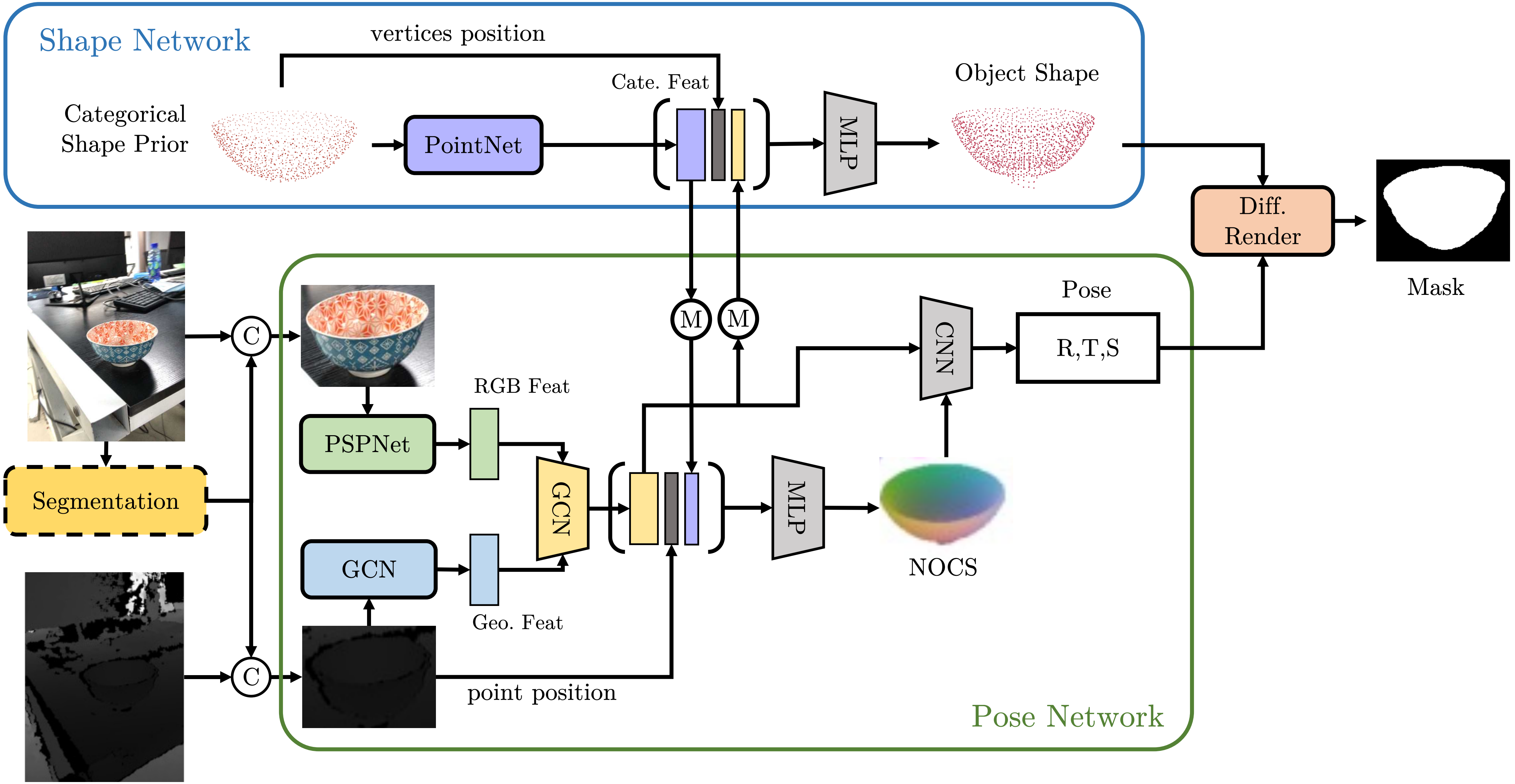
4. Utilize the concatenated feature and input point positions to predict the NOCS coordinate



5. Input the predicted NOCS coordinate and its corresponding RGBD feature to the pose regression network to estimate the 3D Rotation, 3D Translation and 3D Size.



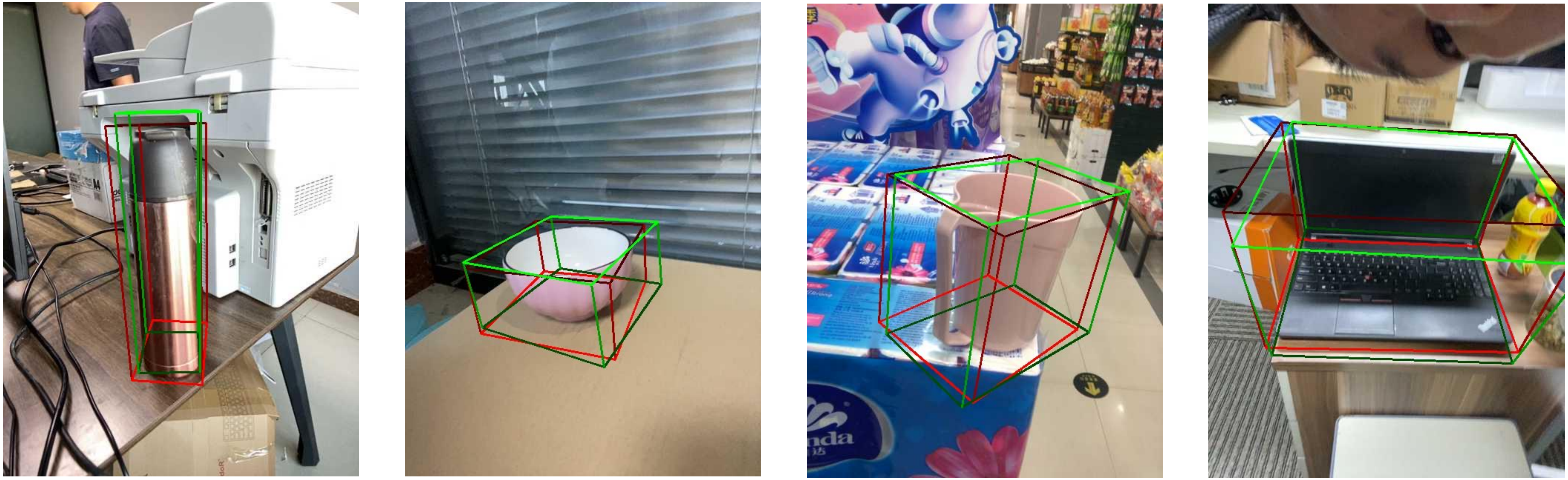
6. Concatenate the categorical feature with RGBD feature and the mesh vertex position to reconstruct the object shape.



7. Predict the object mask given the estimated object pose and reconstructed object shape by the differentiable rendering.

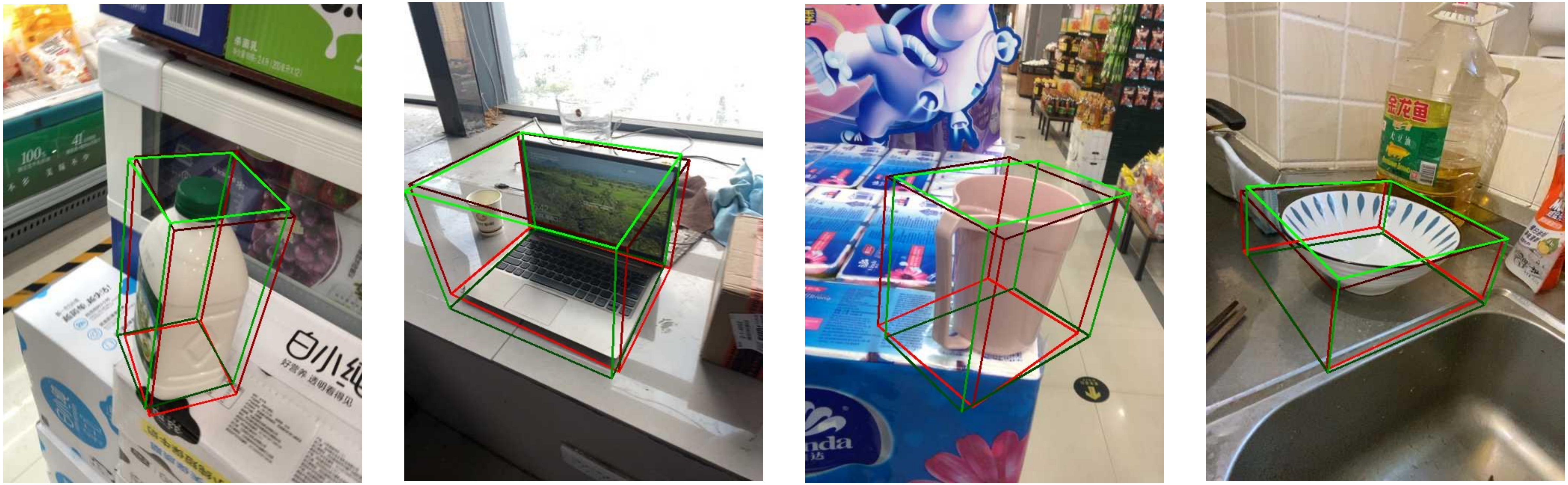
Results on Wild6D

Results: Pose Estimation on Wild6D



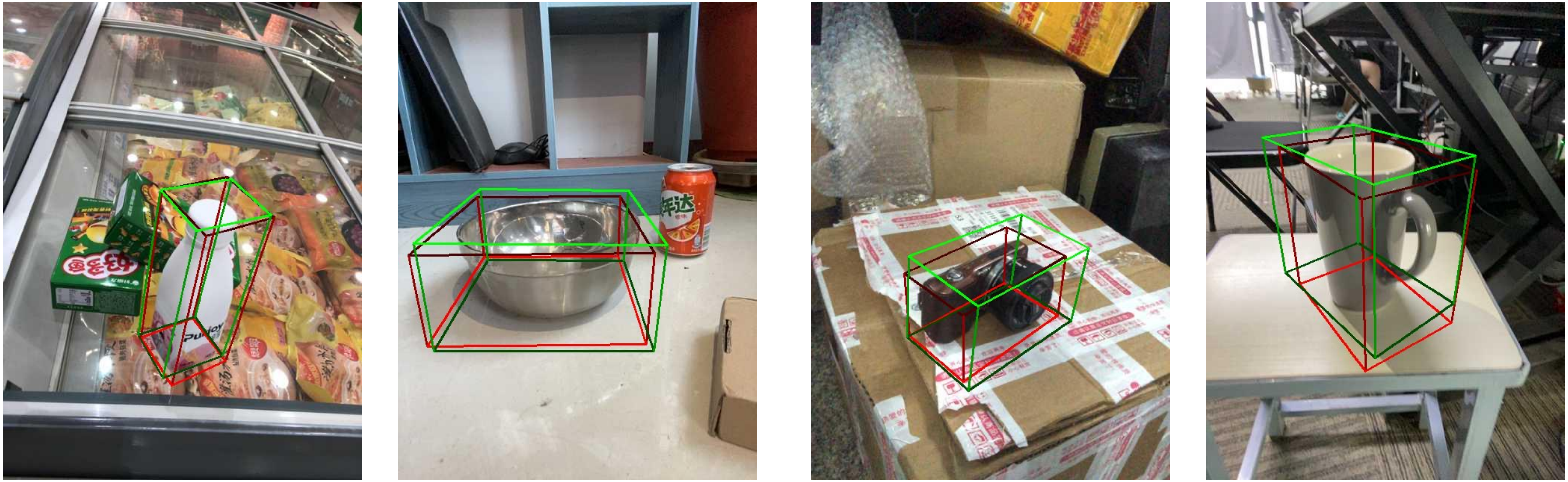
Red box indicates the ground-truth pose, Green indicates the predicted one

Results: Pose Estimation on Wild6D



Red box indicates the ground-truth pose, Green indicates the predicted one

Results: Pose Estimation on Wild6D



Red box indicates the ground-truth pose, Green indicates the predicted one

Thank you