



Introduction to Embodied AI

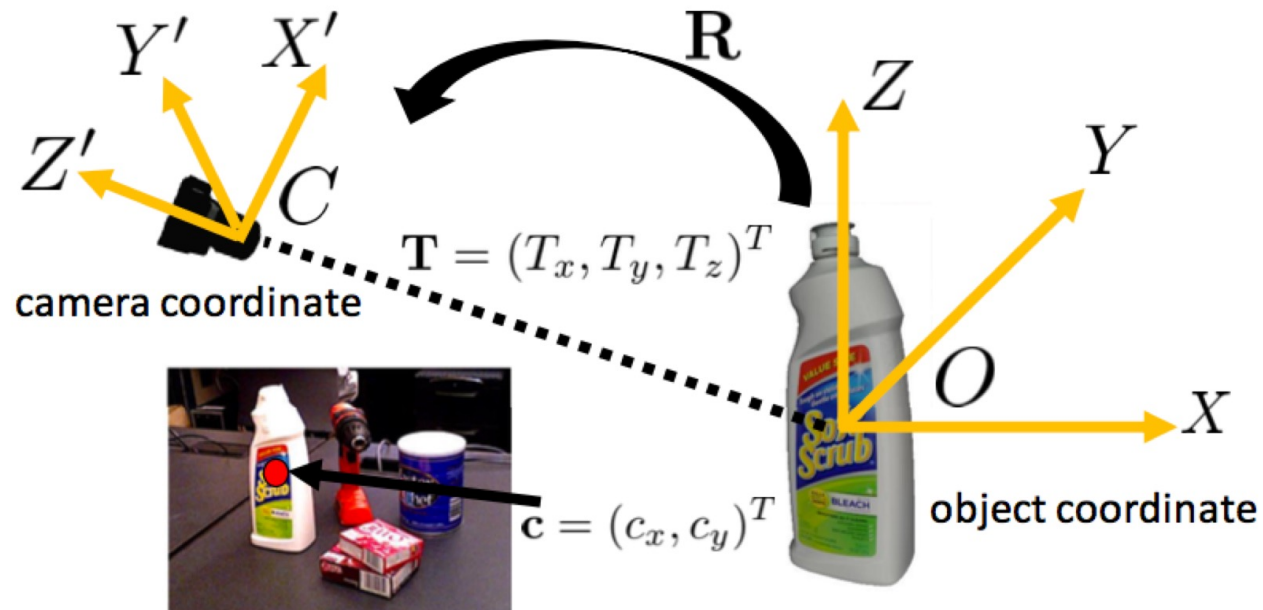
Lecture 6 - Vision and Grasping II

Prof. He Wang



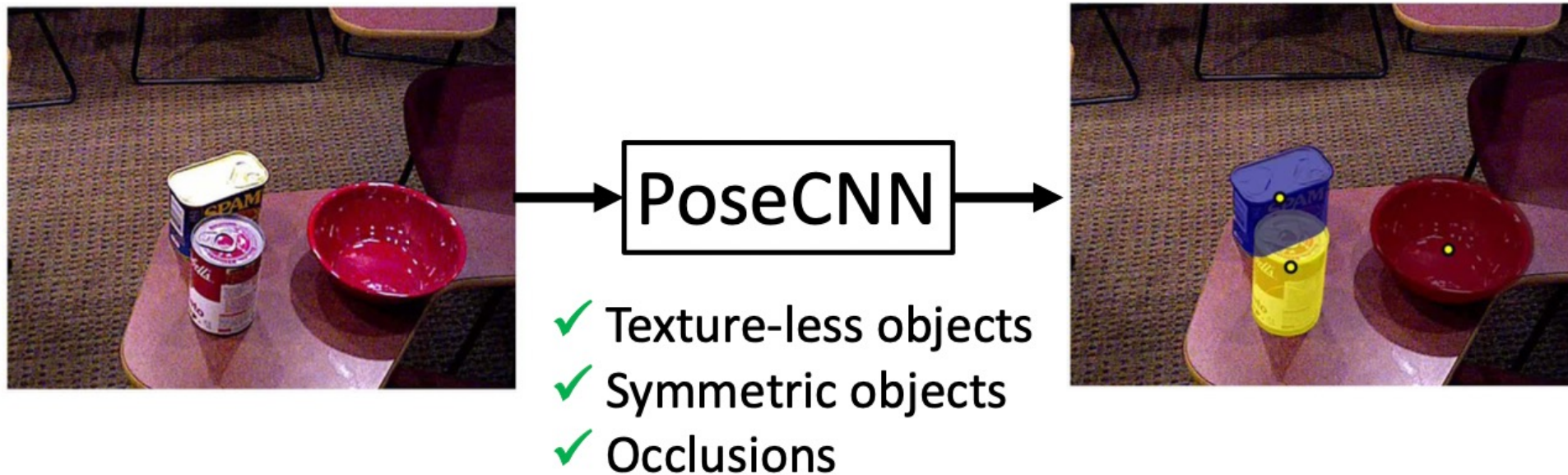
Instance-Level 6D Object Pose Estimation

Instance-Level 6D Pose Estimation



- Instance-level: a small set of known instances
- Pose is defined for each instance according to their CAD model.
- Input: RGB/RGBD
- Output: 6D pose(s) of an object or all known objects

PoseCNN



PoseCNN

- 3D Translation $\mathbf{T} = (T_x, T_y, T_z)^T$



2D center

$$\mathbf{c} = (c_x, c_y)^T$$

Distance T_z

2D Center Localization

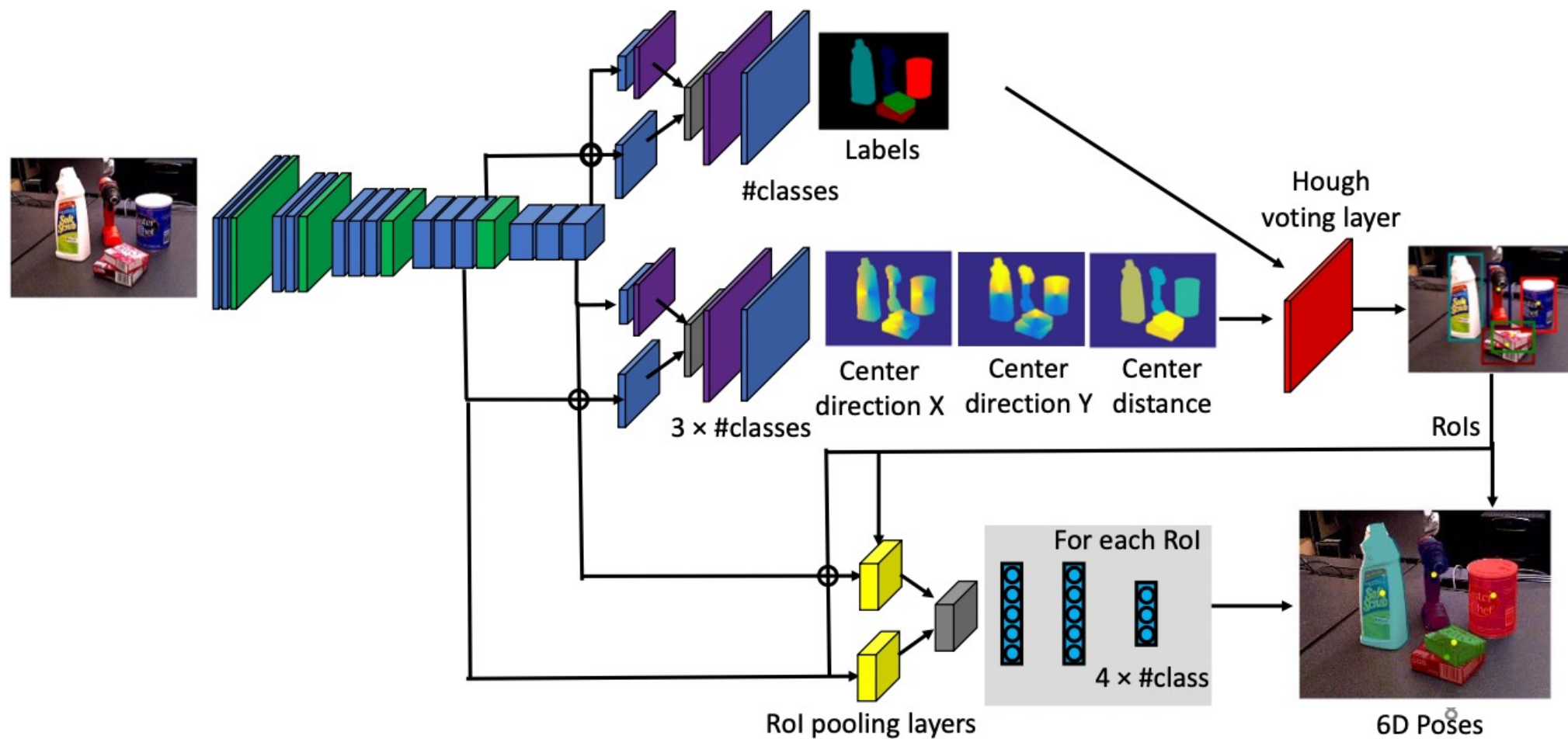
- 3D Rotation $\mathbf{R}$



$\mathbf{R}$

3D Rotation Regression

PoseCNN



Regress quaternion.

PoseCNN: Result

Our YCB-Video Dataset

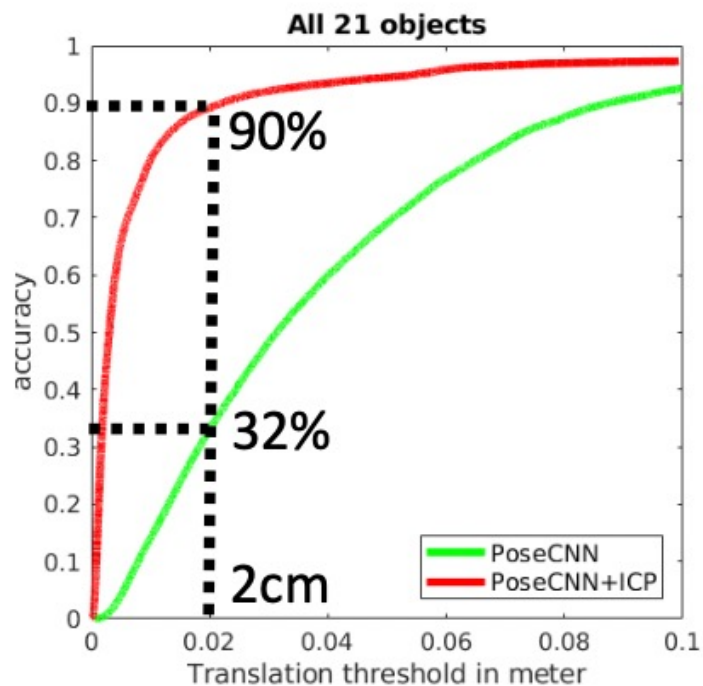


21 YCB Objects

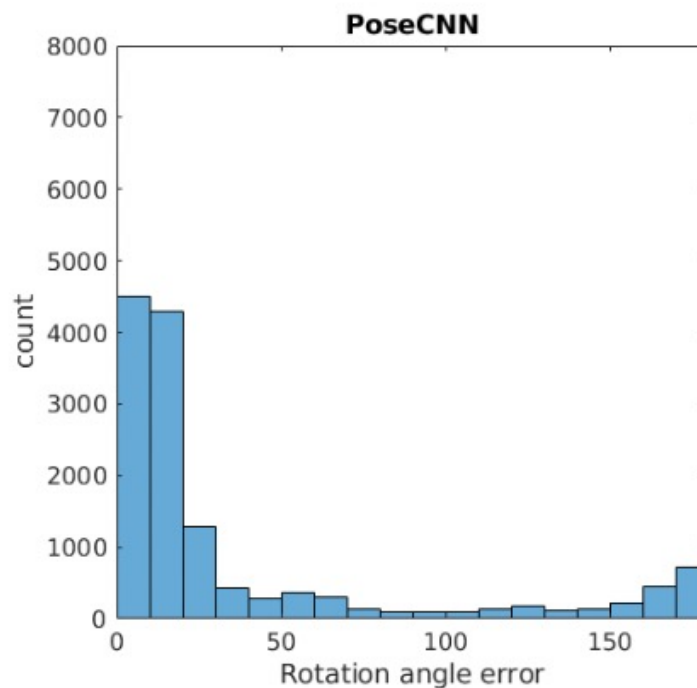


92 Videos, 133,827 frames₀

PoseCNN: Result



3D Translation Error Analysis



3D Rotation Error Analysis

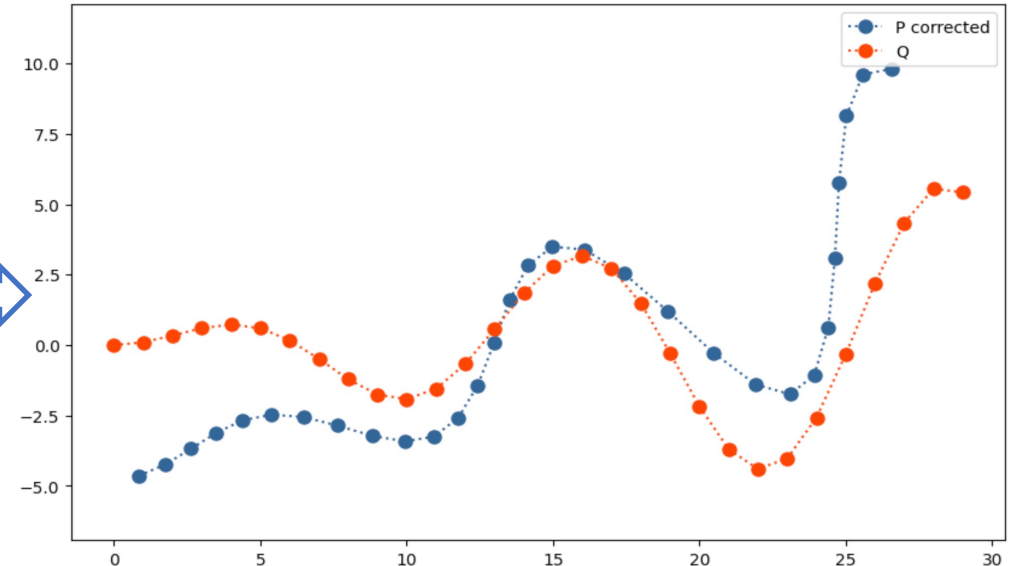
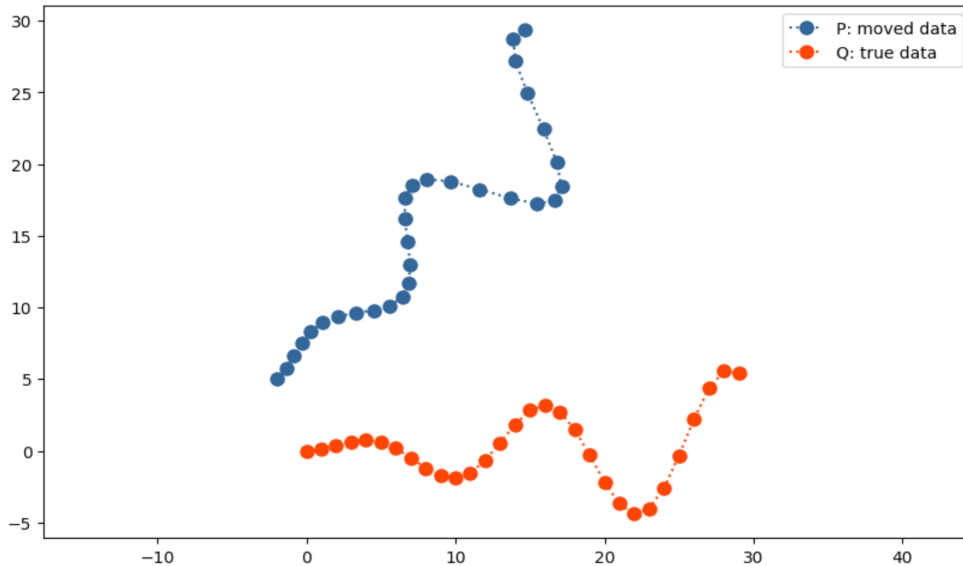
Iterative Closet Point

Iterative Closet Point (ICP)

- ICP is a method for point cloud registration. It takes two point clouds as input and computes an **R & T matrix** that transforms one point cloud to align with the other as closely as possible.

$$\hat{R}, \hat{T} = \operatorname{argmin}_{R \in SO(3), T \in \mathbb{R}^{1 \times 3}} \|Q - (RP + T)\|_F^2$$

where $R \in SO(3), T \in \mathbb{R}^3, Q \in \mathbb{R}^{3 \times n}, P \in \mathbb{R}^{3 \times n}$.

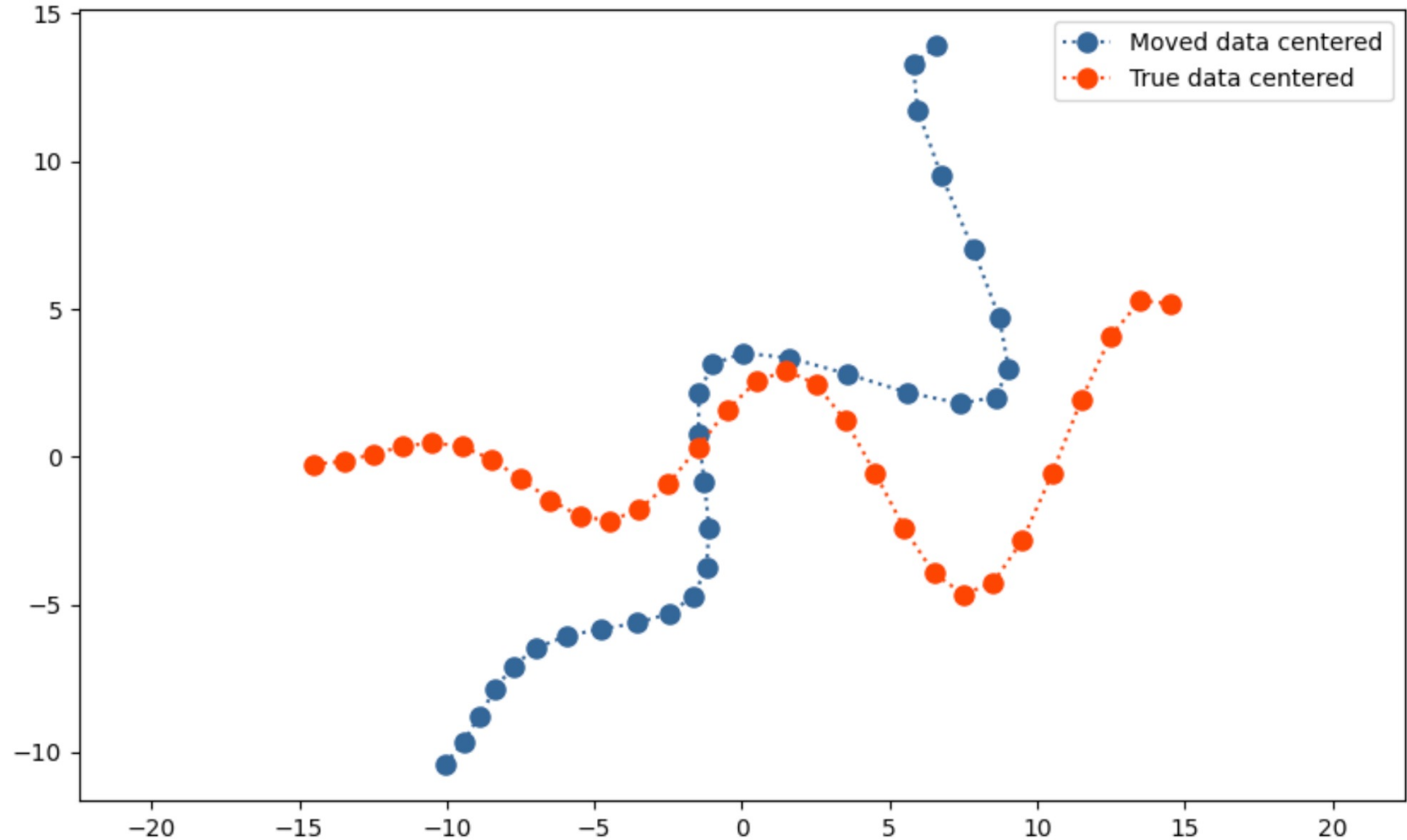


Iterative Closet Point (ICP)

➤ Make data centered

$$\tilde{P} = P - \bar{P}$$

$$\tilde{Q} = Q - \bar{Q}$$

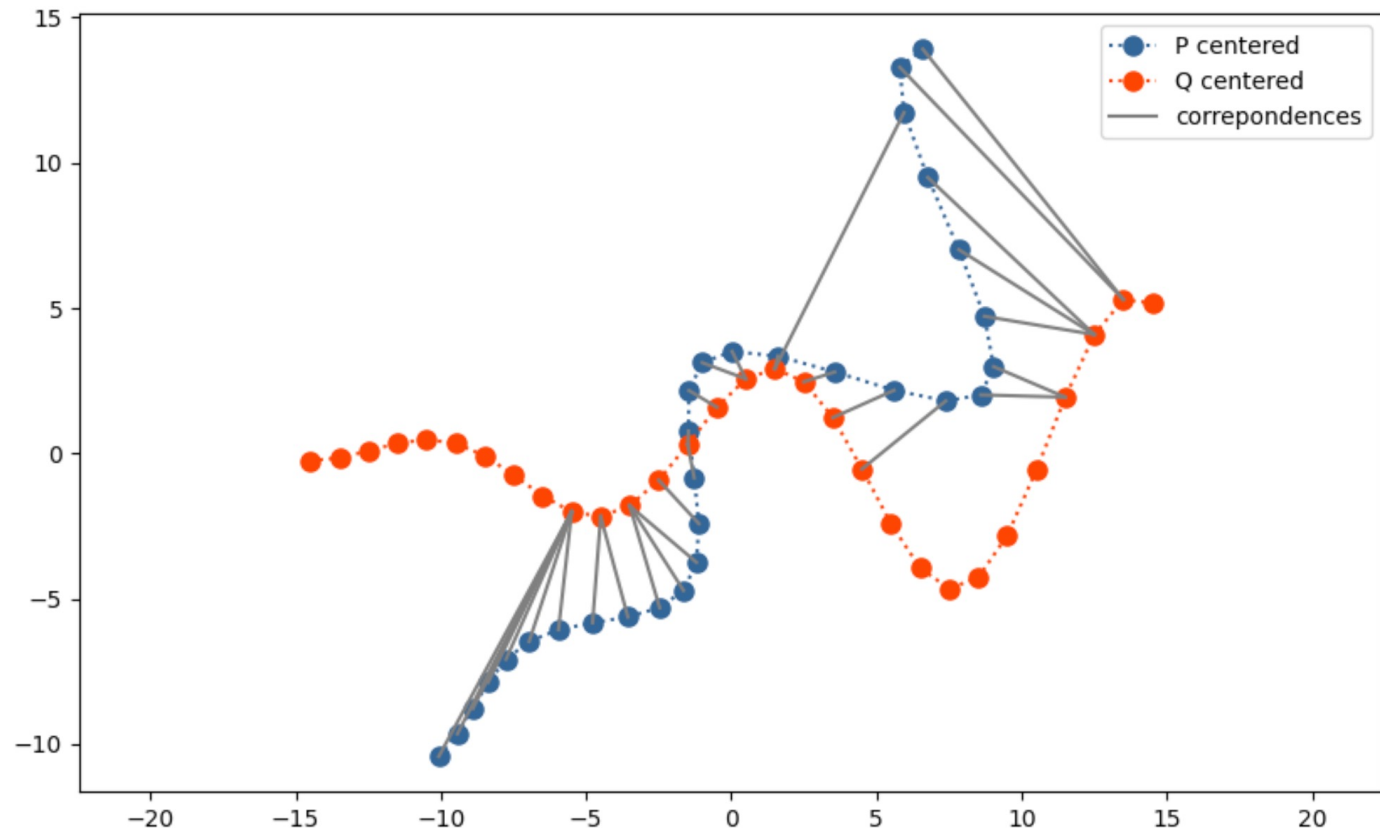


Iterative Closet Point (ICP)

- Compute chamfer distance between two point clouds (for every p_i , search the closest q_j to it) and get correspondences $\tilde{P}_{corr}$ from $\tilde{P}$ to $\tilde{Q}$:

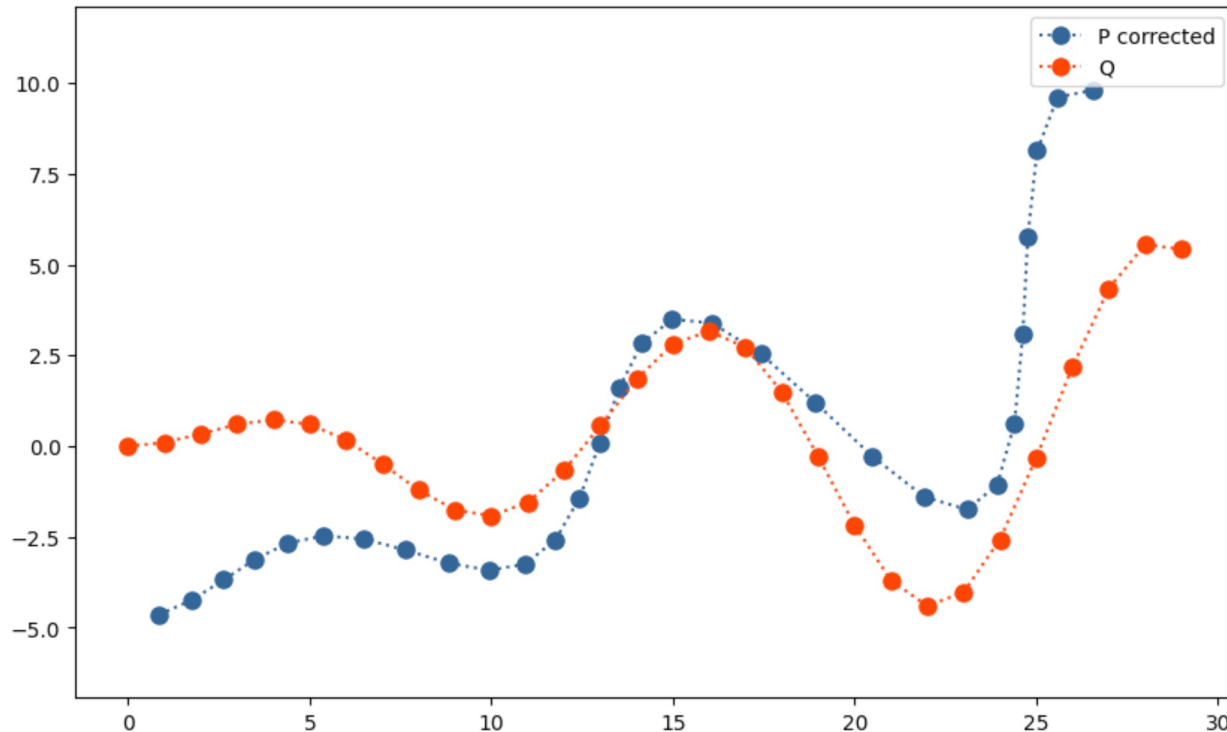
$$\tilde{Q}: \quad \tilde{p}_{i,corr} = \operatorname{argmin}_{\tilde{q}_j \in \tilde{Q}} \|\tilde{q}_j - \tilde{p}_i\|_2^2$$

- Computational complexity is $O(|P| * |Q|)$
- Use KD-tree to accelerate search.



Iterative Closet Point (ICP)

- Problem becomes $\hat{R} = \operatorname{argmin}_{R \in SO(3)} \|\tilde{P}_{corr} - R\tilde{P}\|_F^2$
- Find $\hat{R}$ using constrained orthogonal Procrustes
- Then obtain T from the residual
- Update P

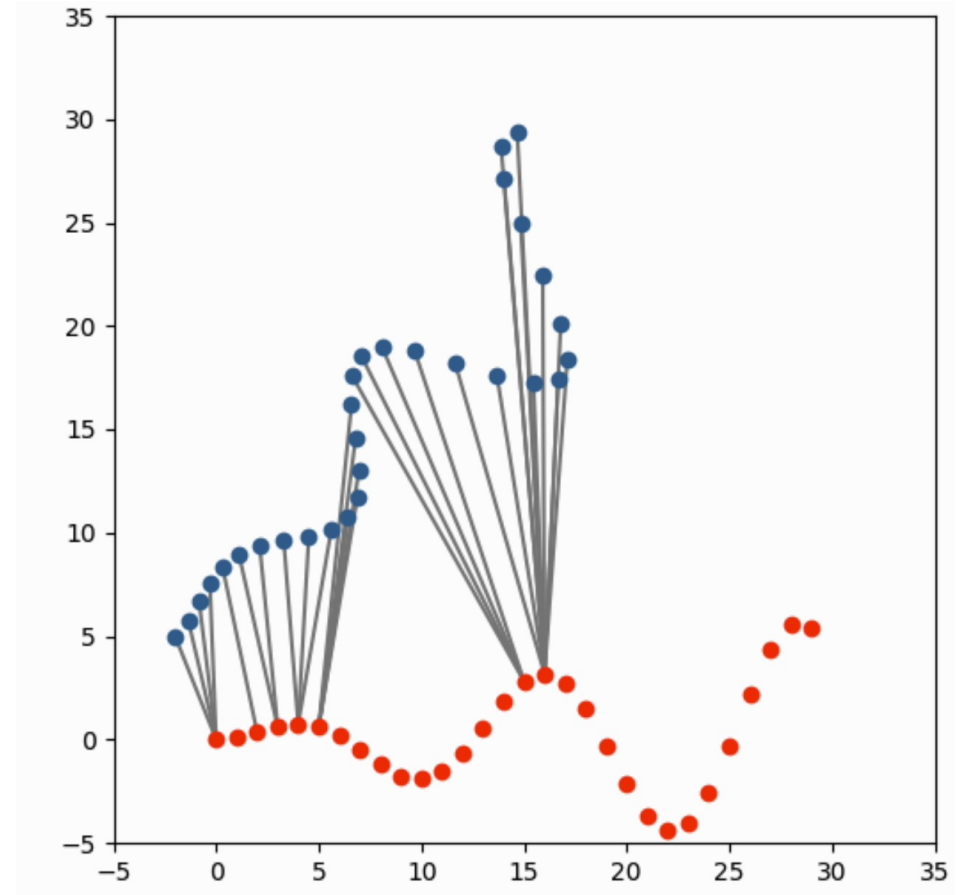


$$U, D, V^T = SVD(\tilde{P}_{corr}\tilde{P}^T)$$
$$\hat{R} = U \operatorname{diag}\{1, 1, \det UV^T\} V^T$$
$$\hat{T} = \bar{Q} - \hat{R}\bar{P}$$
$$P_{new} = \hat{R}P + \hat{T}$$

Iterative Closet Point (ICP)

➤ Make it iterative

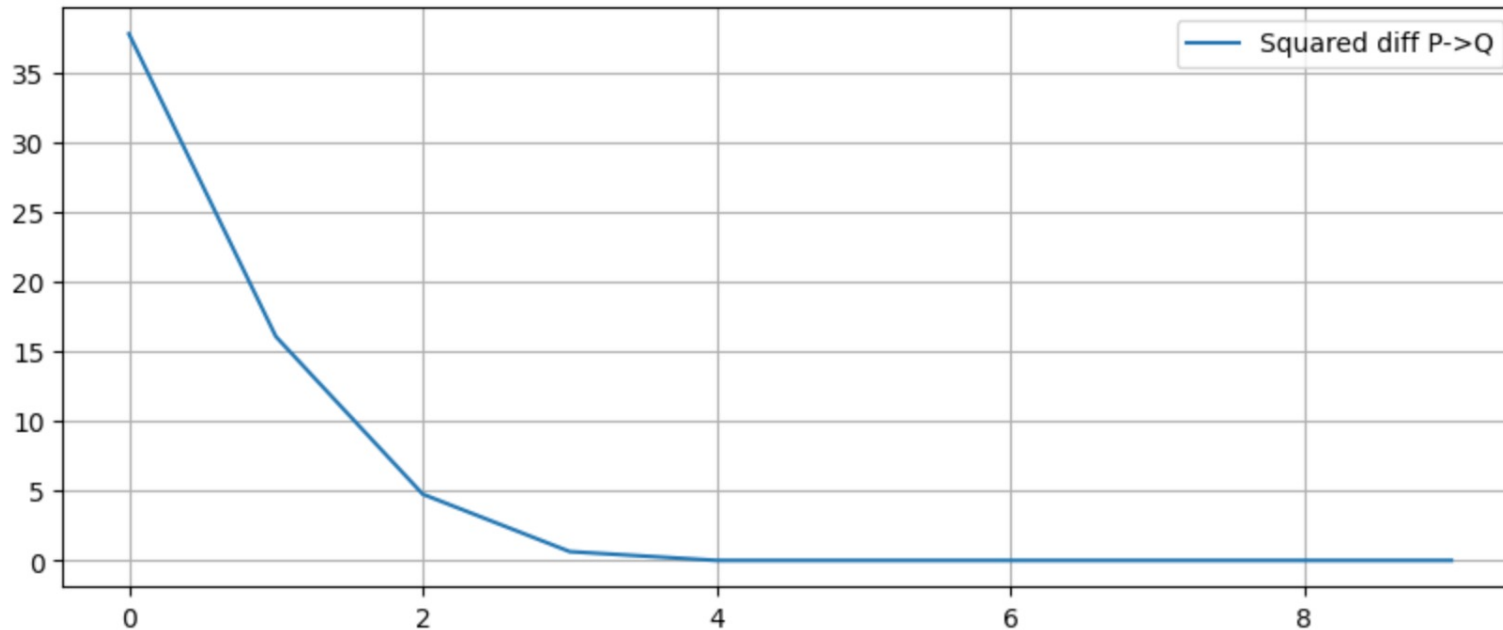
1. Make data centered by subtracting the mean
2. Find correspondences for each point in $\tilde{P}$
3. Get the rotation $\hat{R}$ by constrained orthogonal Procrustes
4. Apply the $\hat{R}$ to the mean vector of P and get $\hat{T}$
5. Update P using $\hat{R}$ and $\hat{T}$



Iterative Closet Point (ICP)

➤ Iteration termination condition.

1. $\hat{R}$, $\hat{T}$ is smaller than a certain threshold.
2. The change in loss is smaller than a certain threshold.
3. The maximum number of iterations is reached.



Iterative Closet Point (ICP)

➤ Advantages

- Simple, no need for point cloud segmentation or feature extraction.
- When the initial estimate is good, it has decent accuracy and convergence.

➤ Disadvantages

- High computational cost for finding the nearest corresponding points (can be reduced by downsampling dense point clouds or using small-sample matching for faster iterations).
- Only considers point-to-point distances, lacking utilization of point cloud structural information.
- Highly dependent on the accuracy of the initial estimate.

Iterative Closet Point (ICP)

➤ ICP Variants

- Point-to-Plane ICP

Takes the local structure of the point cloud into account by considering the distance from the source point to the plane of the target point.

- Plane-to-Plane ICP

Considers the local structure of the source point cloud and computes the plane-to-plane distance.

- Fast and Robust ICP

Improve the speed and robustness of ICP

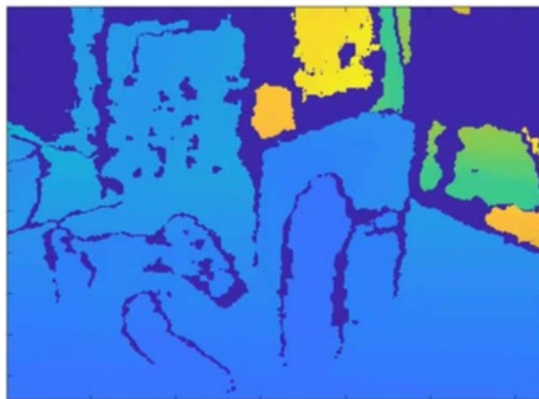
<https://arxiv.org/abs/2007.07627>

Video: <https://www.youtube.com/watch?v=LcghboLgTiA>

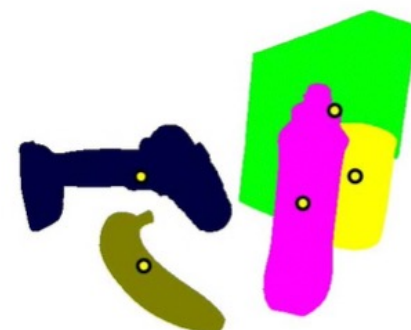
PoseCNN: Result Visualization



RGB



Depth



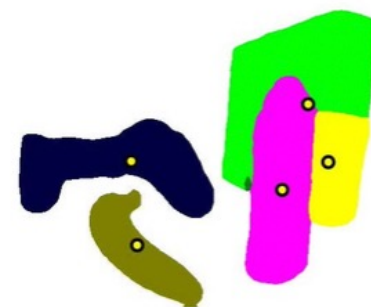
Groundtruth Labels



PoseCNN (RGB only)

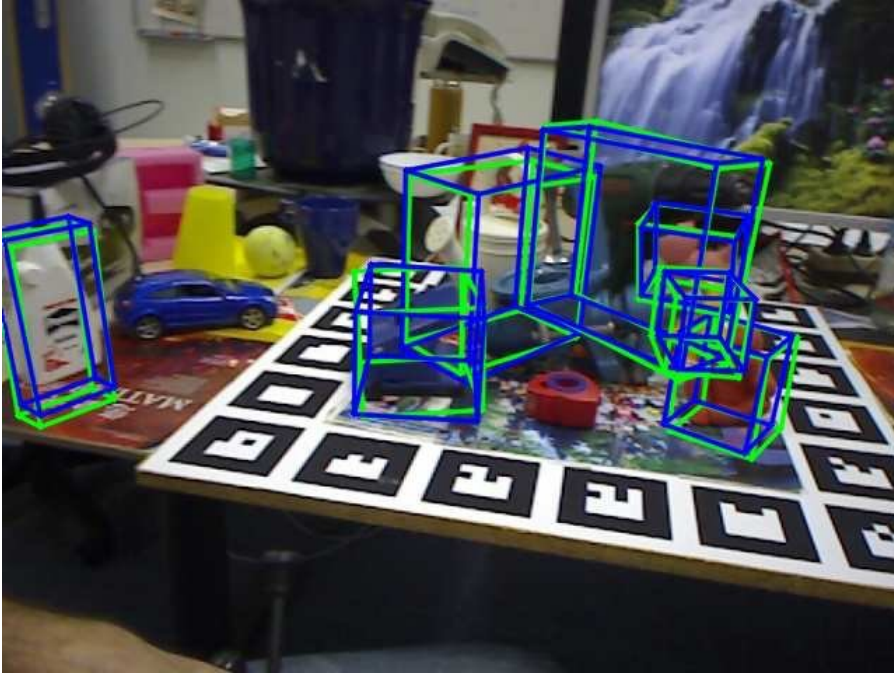


PoseCNN + ICP



Predicted Labels

Instance-Level 6D Pose Estimation

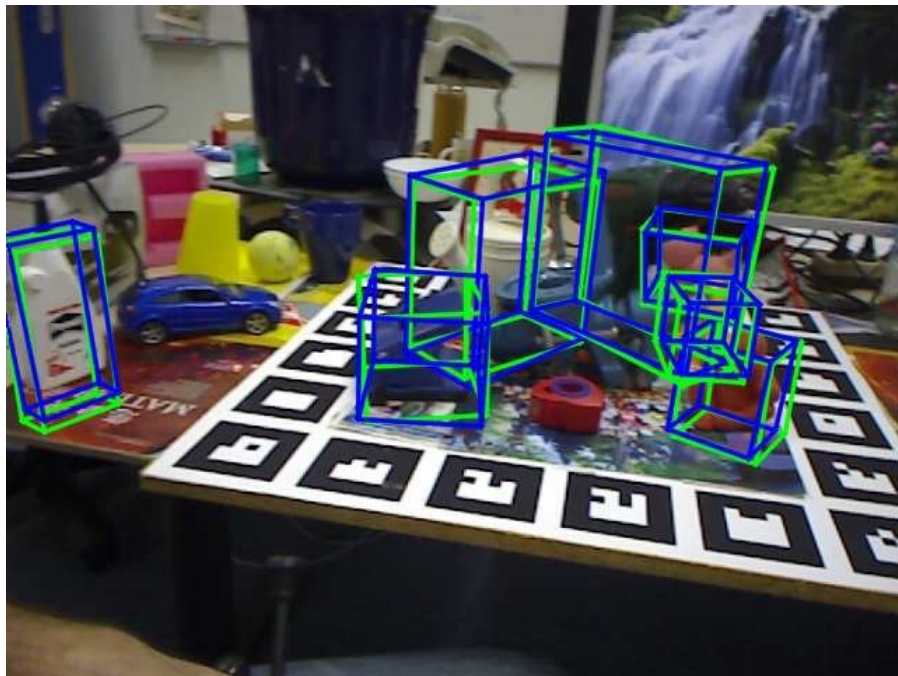


LINEMOD dataset

- A small set of known instances
- Ground truth CAD model available
- Useful in controlled environments

Hinterstoisser, Stefan, et al. "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes." *Asian conference on computer vision*. Springer, Berlin, Heidelberg, 2012.

Limitations of Instance-Level 6D Pose Est.



LINEMOD dataset

Limitations

- Instance-specific
- Can't generalize to novel objects
- Needs CAD model

Hinterstoisser, Stefan, et al. "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes." *Asian conference on computer vision*. Springer, Berlin, Heidelberg, 2012.

Category-Level 6D Object Pose Estimation

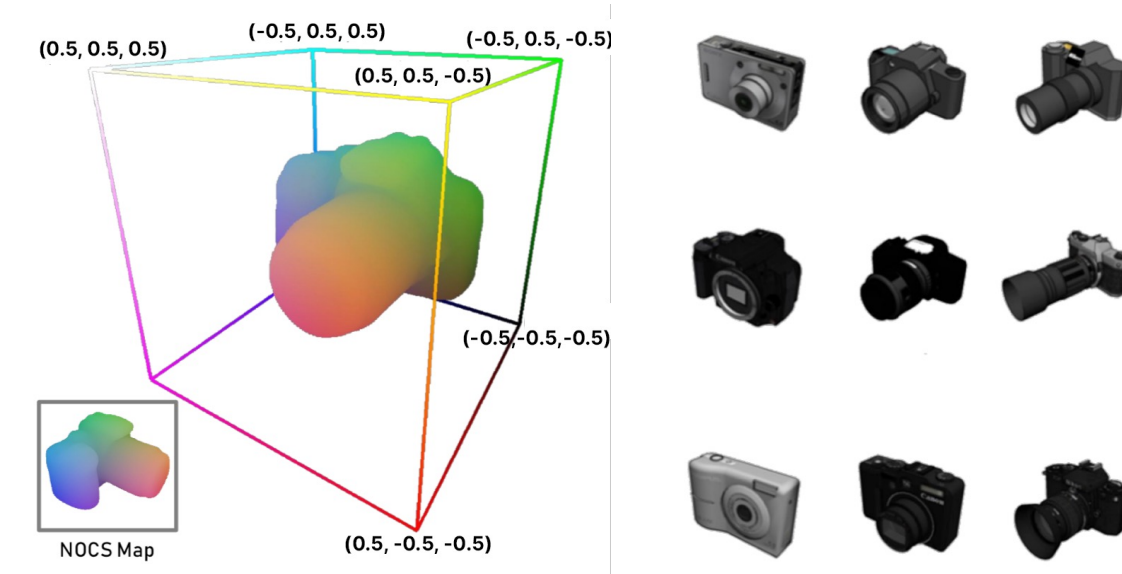
Category-Level 6D Pose



Our goal

- Generalize to novel object instances
- **3D input, full 6D pose output**
- Without the need to use CAD model

Normalized Object Coordinate Space for Category-Level 6D Object Pose and Size Estimation



He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, Leonidas Guibas

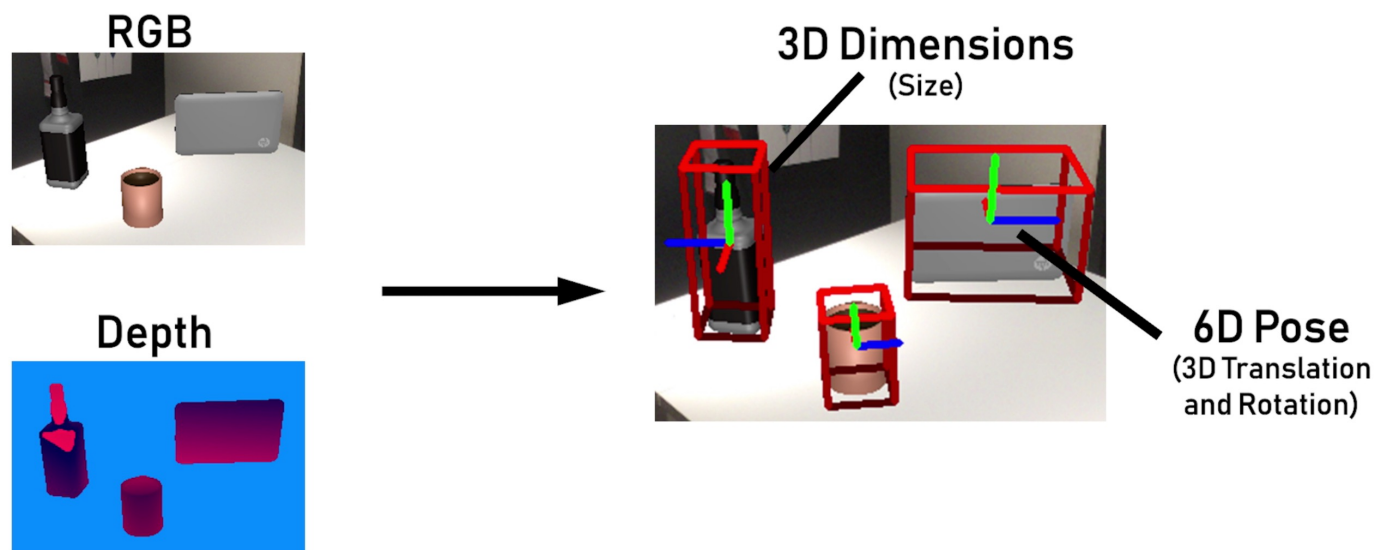


LONG BEACH
CALIFORNIA
June 16-20, 2019

Oral Presentation

Category-Level Object Pose and Size Estimation

Detecting and estimating **6D pose** and **3D size** of **previously unseen** objects from certain categories from RGBD images.



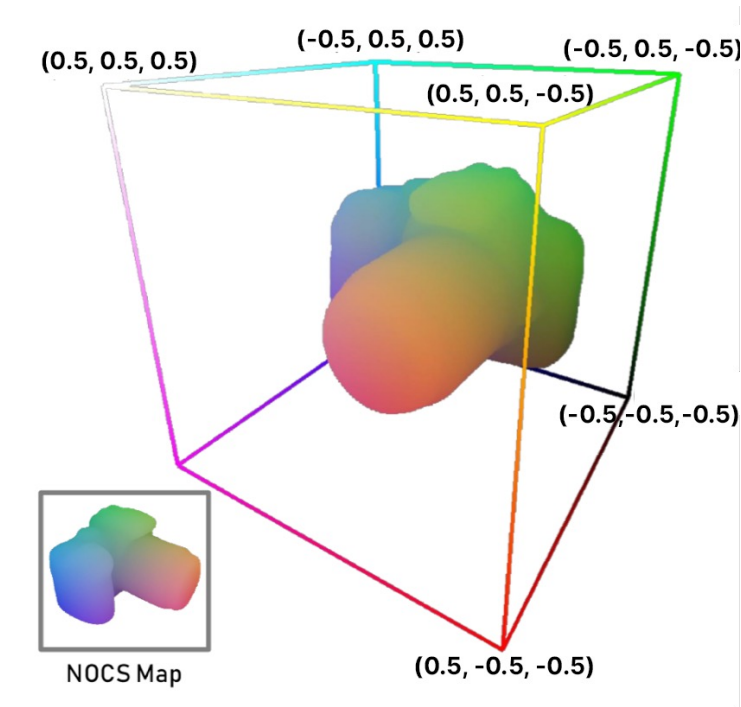
without knowing CAD models

Challenges

- Intra-category shape variations



Representation: Normalized Object Coordinate Space



RGB colors represent XYZ coordinates of shape.

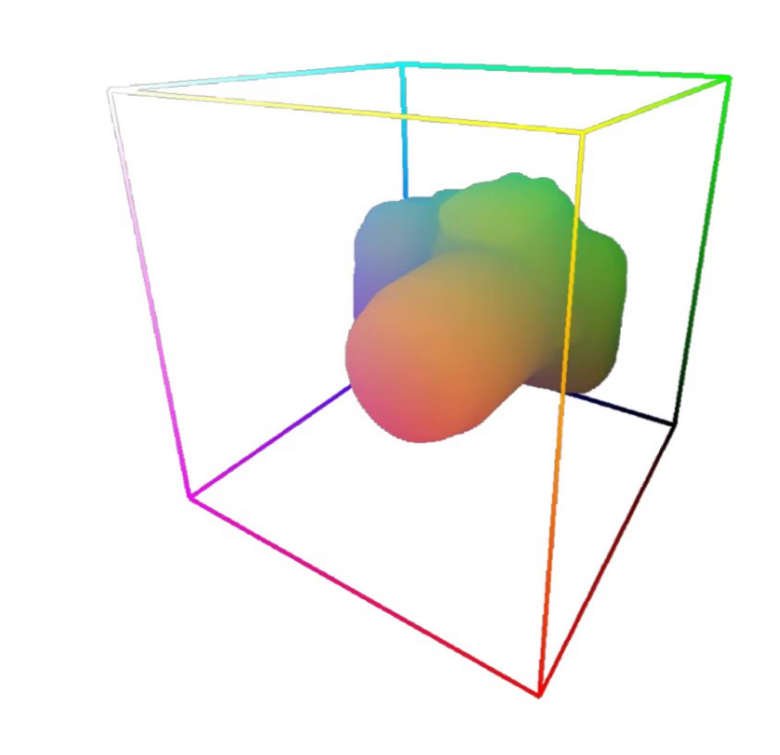
Representation: Normalized Object Coordinate Space

Step 1 (rotation normalization):
align object orientations



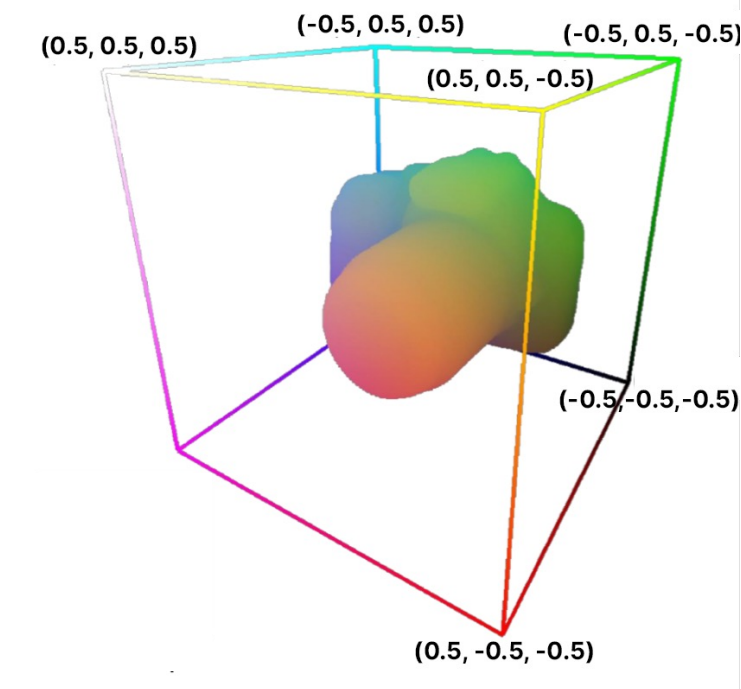
Representation: Normalized Object Coordinate Space

Step 2 (translation normalization):
zero-center the objects



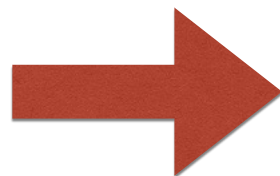
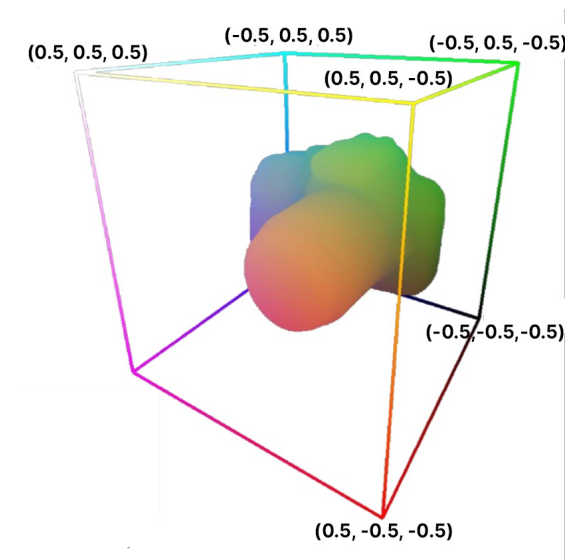
Representation: Normalized Object Coordinate Space

Step 3 (scale normalization):
uniformly normalize the scales

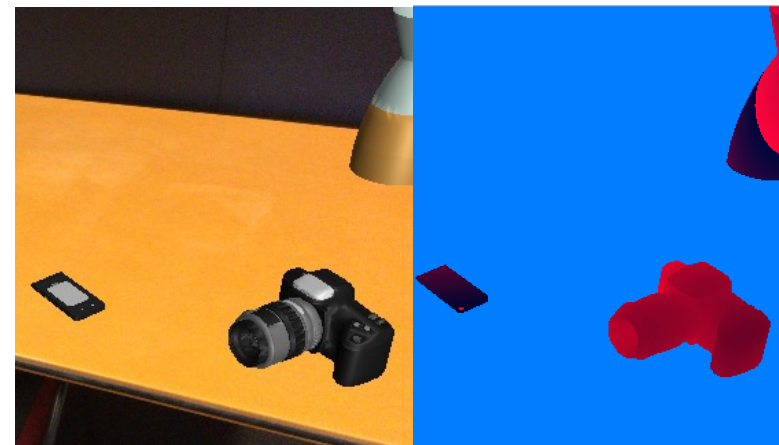


Definition of Category-Level 6D Pose

NOCS = Reference frame



RGBD



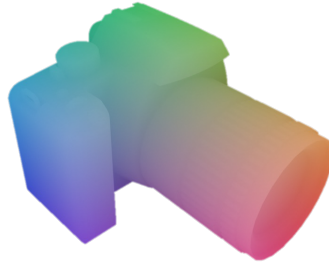
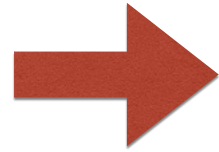
Category-level pose = transformation from NOCS to camera space

From Image to NOCS map to Pose



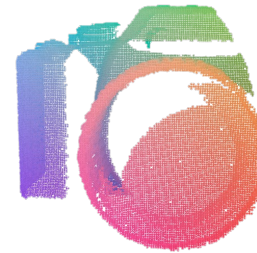
RGB Image

Per-pixel
predict



NOCS map

NOCS to point

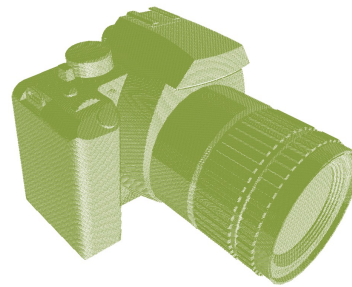
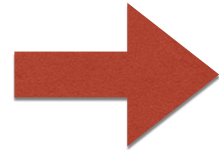


NOCS pts



Depth

Backproject

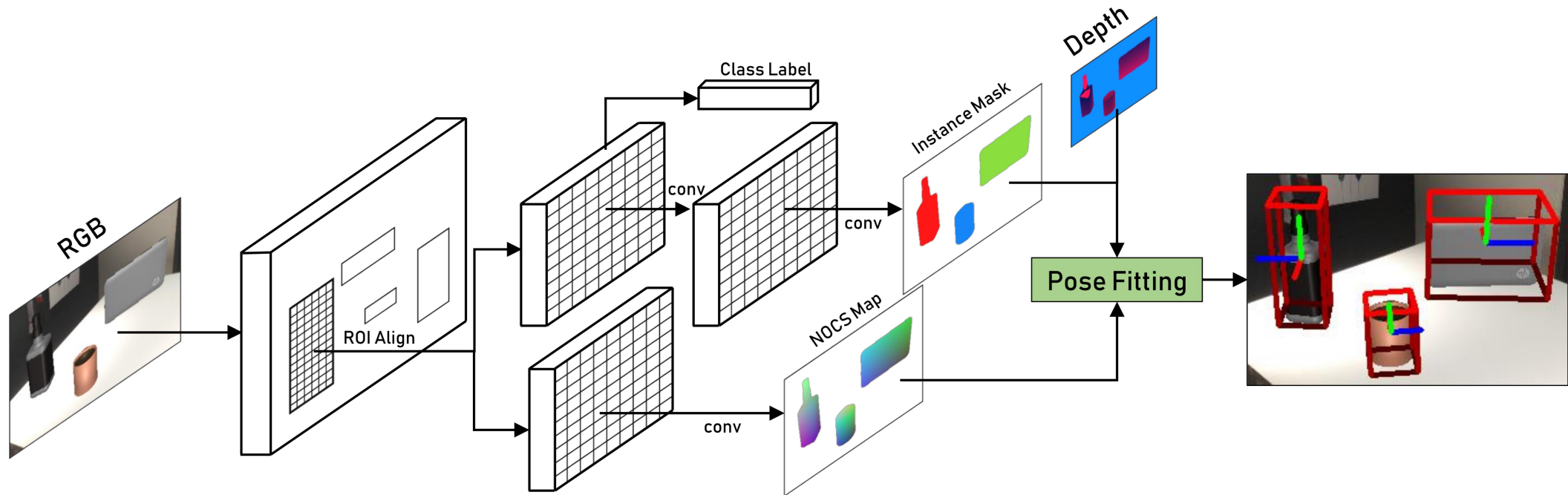


Depth pts



Pose fitting
w/ RANSAC

Network



- Network: a Mask-RCNN-based detector
- Outputs: 9 DoF amodal bounding boxes

Challenges

- Intra-category shape variations
- **Data and annotations**



Challenges

- Intra-category shape variations
- **Data and annotations**

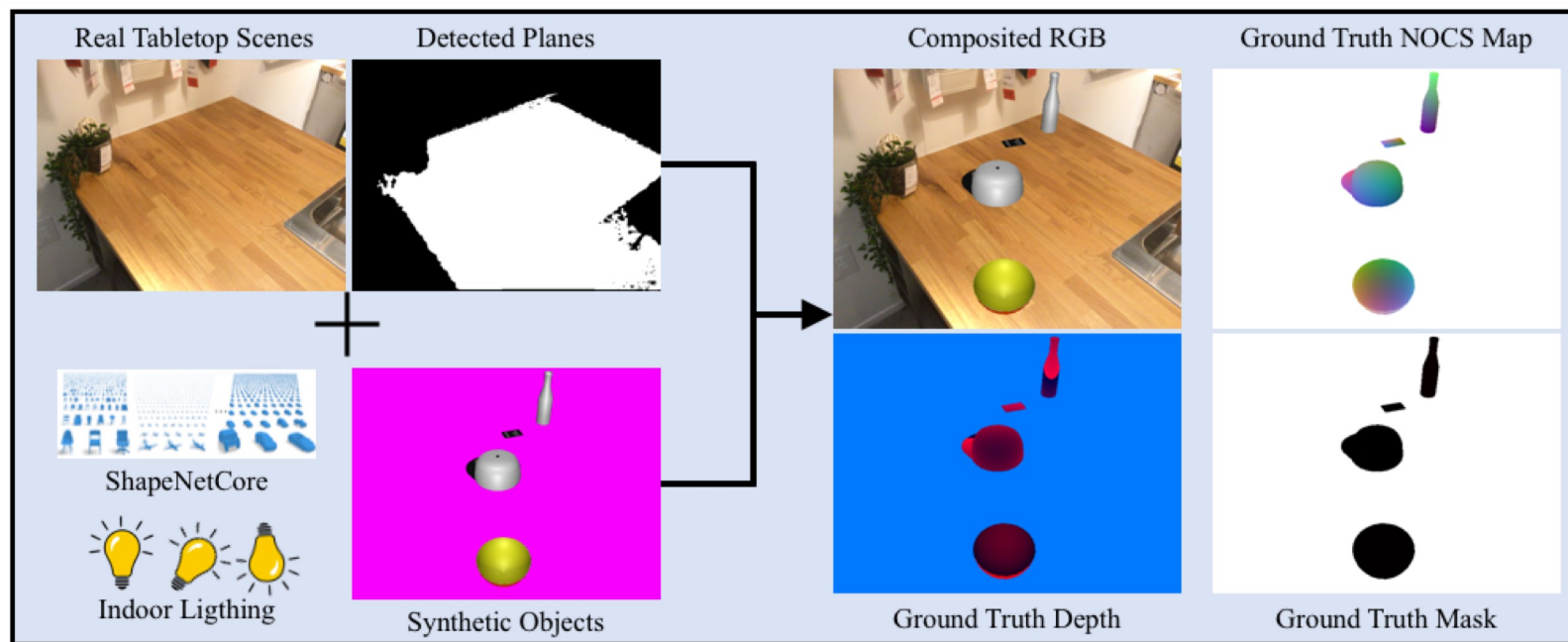


Challenges

- Intra-category shape variations
- **Data and annotations**
 - covering different instances
 - covering different environments
 - accurate 6D pose annotations



Context-Aware MixEd ReAlity (CAMERA)



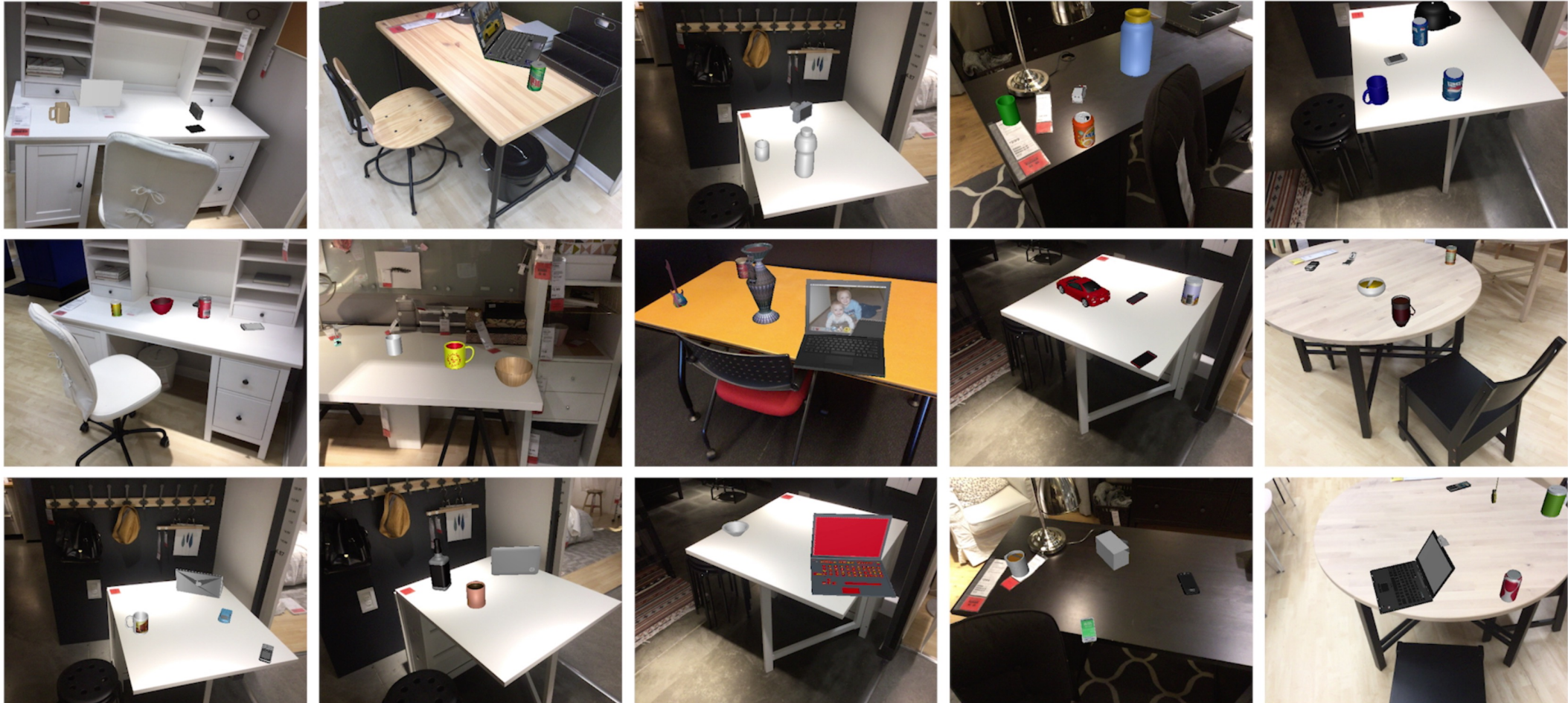
Mixed-reality data generation pipeline

Dataset Statistics

- **Context-Aware MixEd ReAlity**
 - 300K images (275K/25K)
 - 31 scenes from IKEA
 - 1085 ShapeNet models



Mixed Reality Data



Real-World Dataset

* markers not used for prediction



Dataset Statistics

Real Dataset (Train/Val/Test)

- 8K RGB-D (4K/0.95K/3.75K)
- 18 real scenes (6/6/6)
- 42 unique instances (3/1/3)

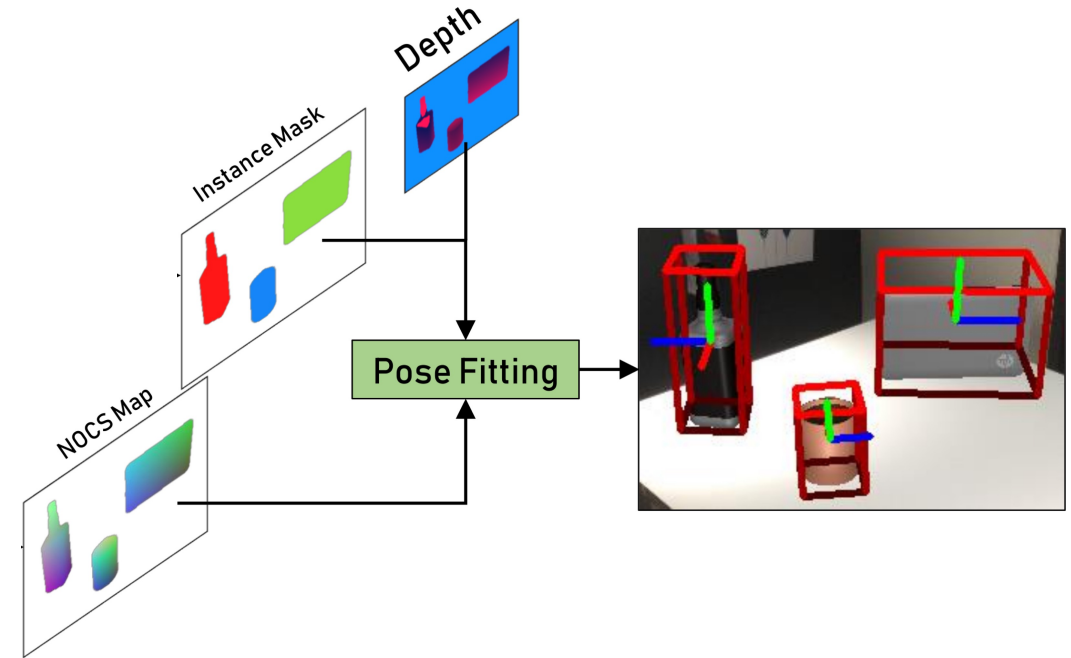
6 Object Categories

- bottle, bowl, camera, laptop, mug
- symmetric vs. asymmetric



Pose and Size Fitting

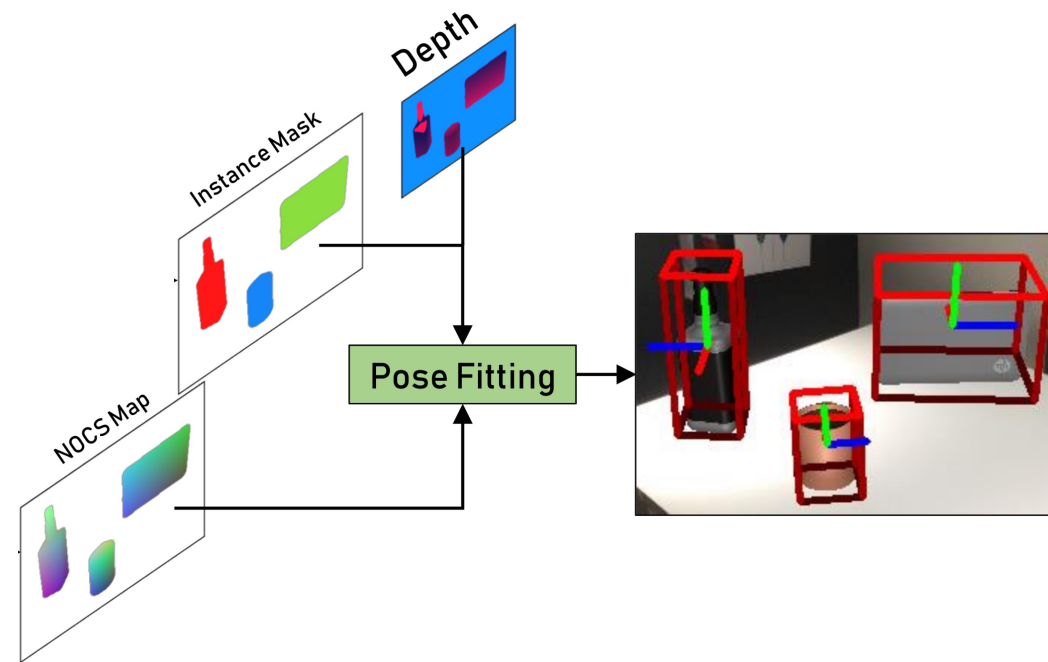
- Umeyama algorithm* for estimating 7D similarity transformation
- RANSAC for outlier removal



<https://zpl.fi/aligning-point-patterns-with-kabsch-umeyama-algorithm/>

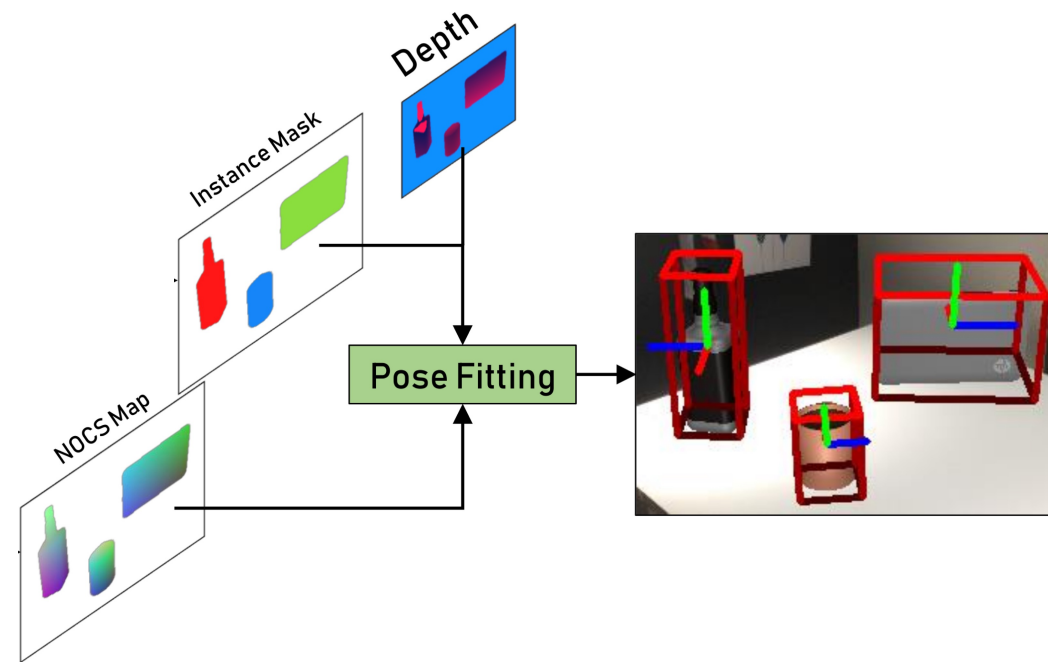
Pose and Size Fitting

- Umeyama algorithm
- RANSAC for outlier removal
- Use predicted NOCS aspect ratio to get 3 sizes



Pose and Size Fitting

- Umeyama algorithm
- RANSAC for outlier removal
- NOCS aspect ratio to get 3 sizes
- Outputs: 9 DoF amodal bounding boxes



Handling Symmetry

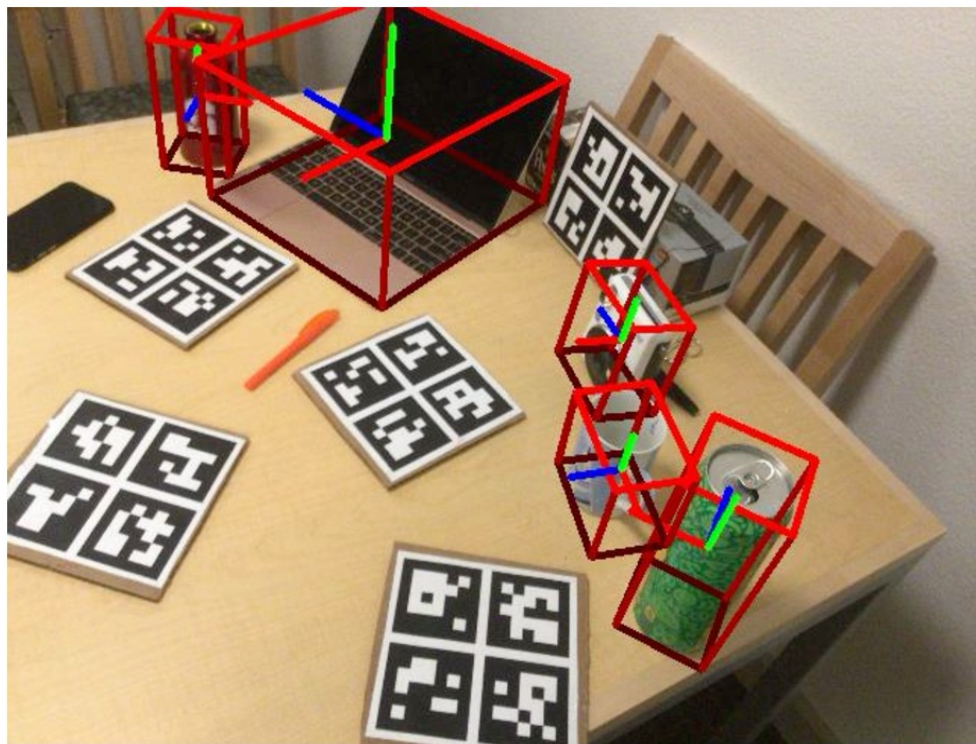
Symmetry-aware loss function: $L_s = \min_{i=1,\dots,|\theta|} L(\tilde{\mathbf{y}}_i, \mathbf{y}^*)$

min-of-N loss

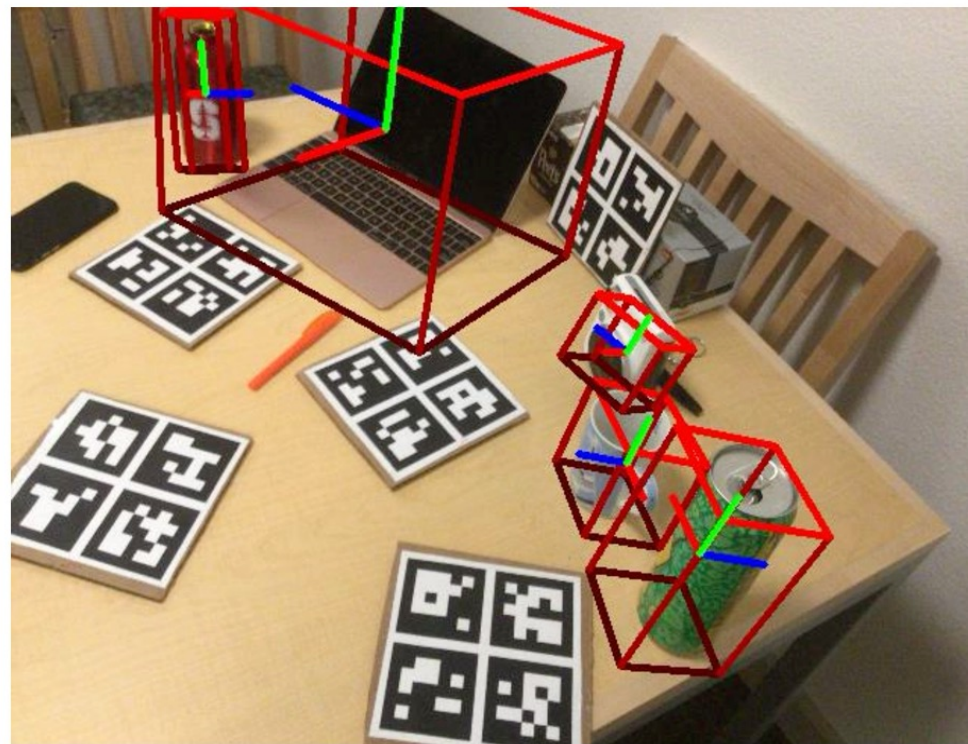


Result Visualization

Ground Truth



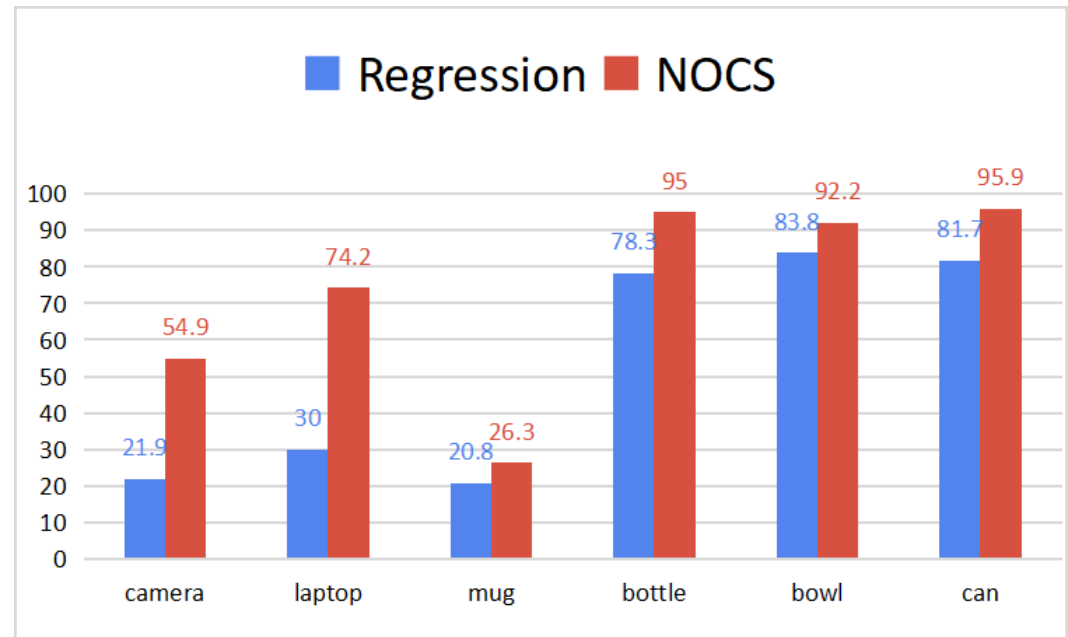
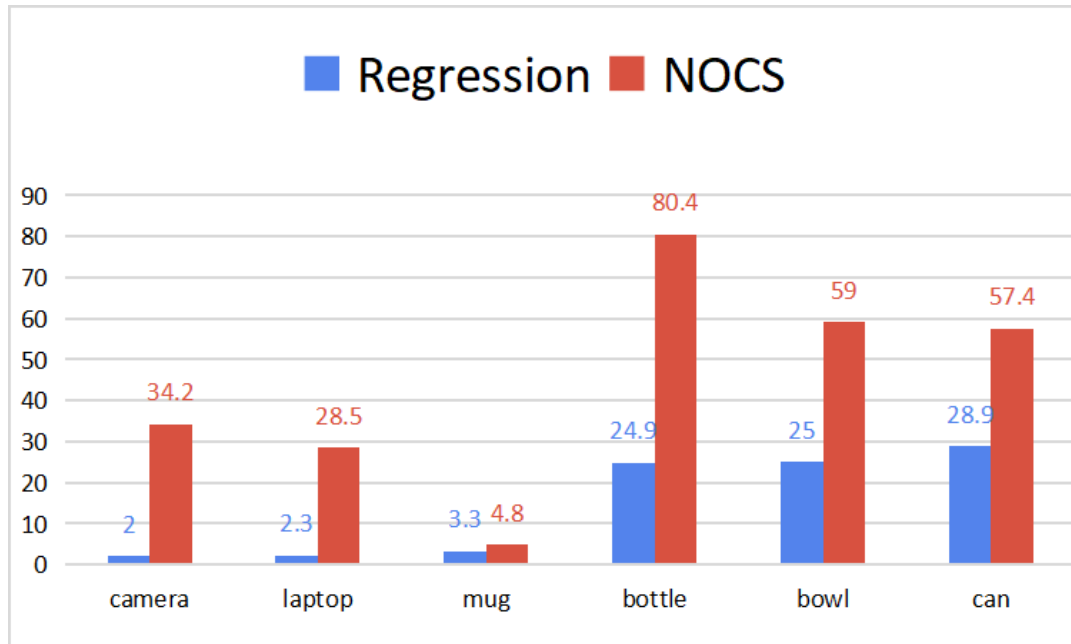
Prediction (Ours)



Ablation Study: NOCS vs. Pose Regression

5 degree acc.

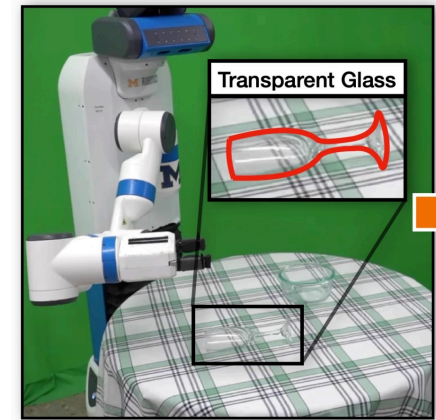
10 degree acc.



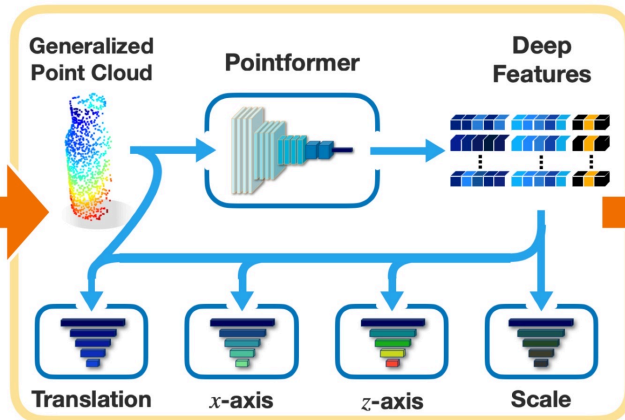
The higher, the better.

Applications of Category-Level Pose Estimation

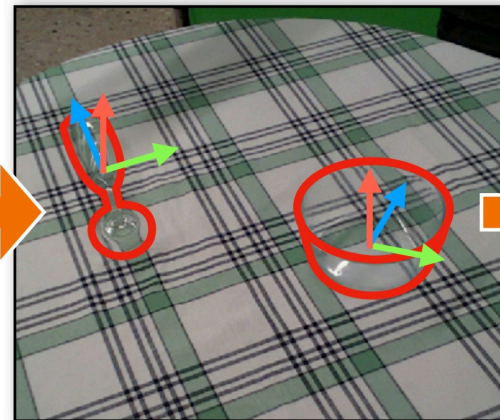
Robot Perceiving Transparent Object



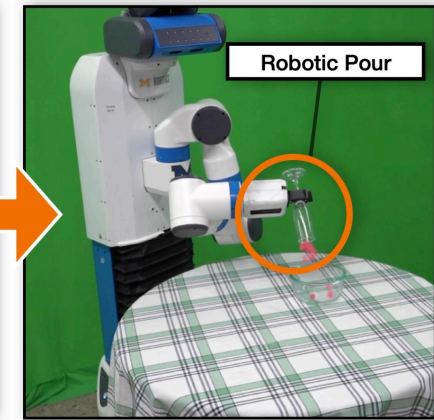
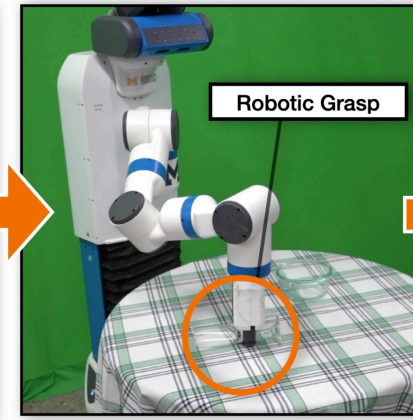
TransNet



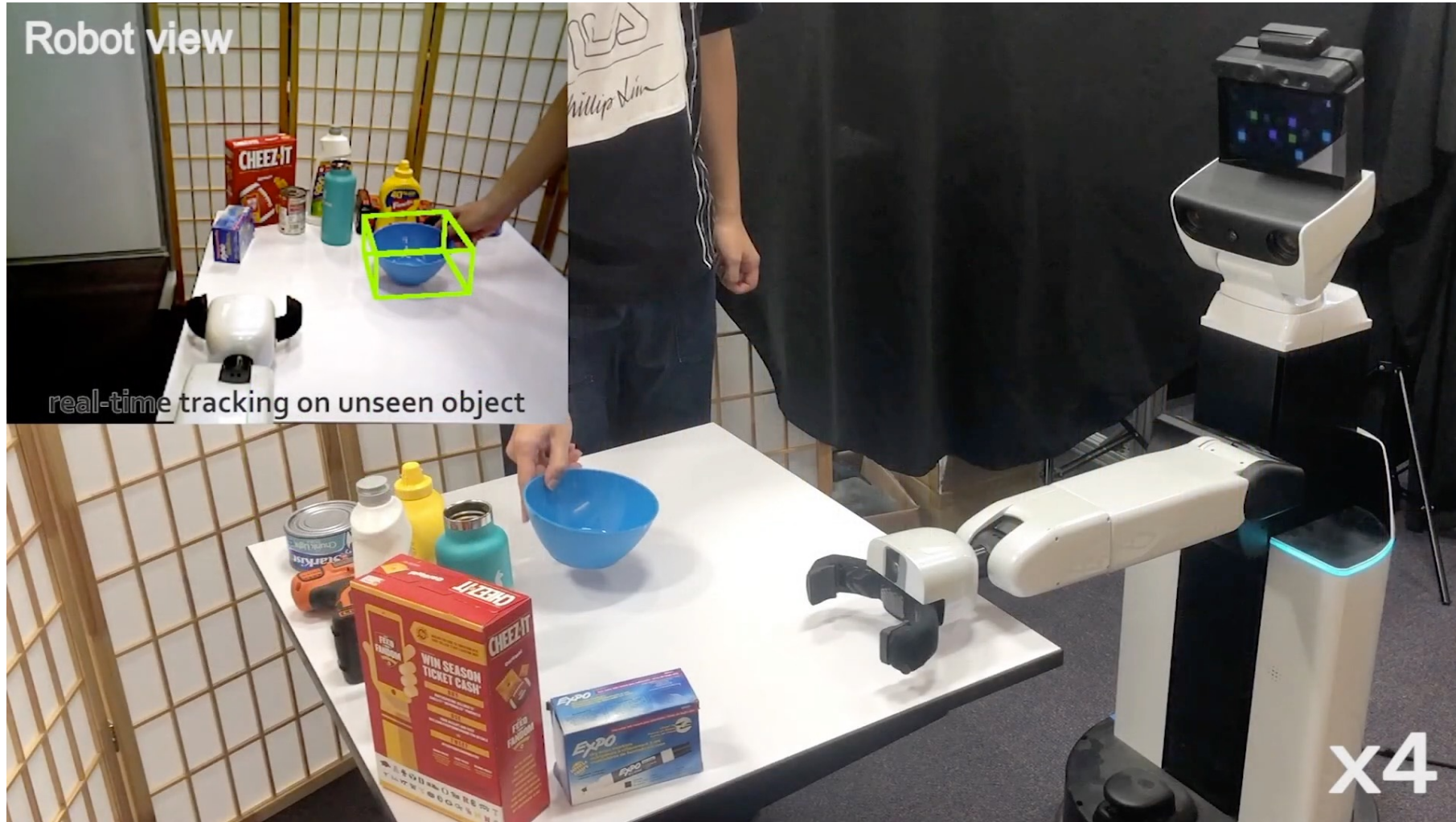
Category-Level Transparent Pose



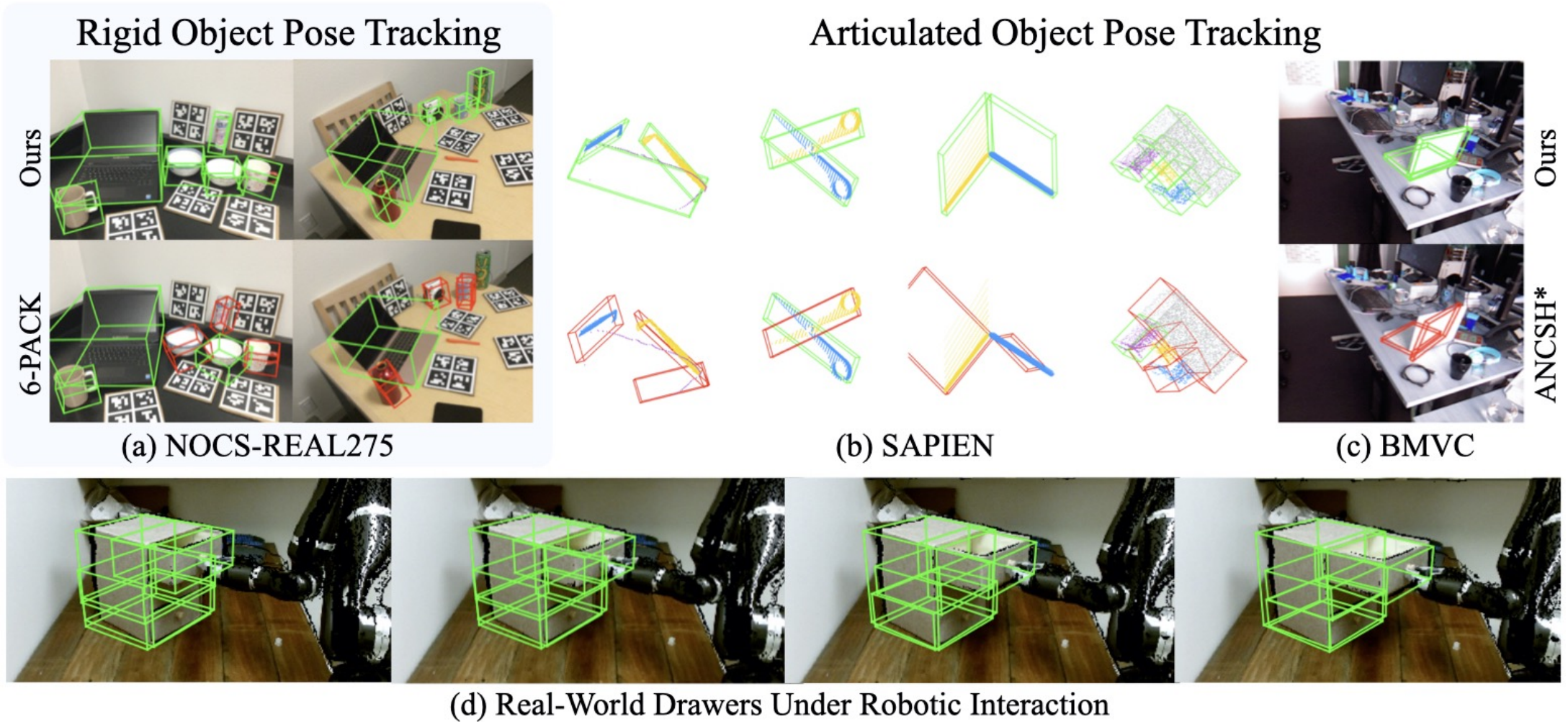
Robot Manipulation of Transparent Objects



Applications of Category-Level Pose Tracking



Pose Estimation for Rigid Object and Articulated Parts



GAPartNet:

Cross-Category Domain-Generalizable Object Perception and Manipulation via Generalizable and Actionable Parts

CVPR 2023 Highlight

Haoran Geng*, Helin Xu*, Chengyang Zhao*,
Chao Xu, Li Yi, Siyuan Huang, He Wang⁺



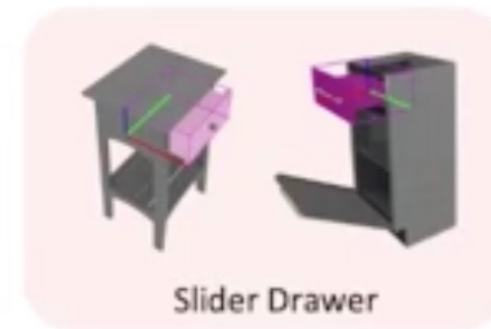
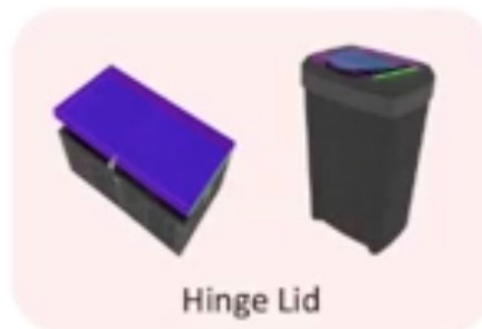
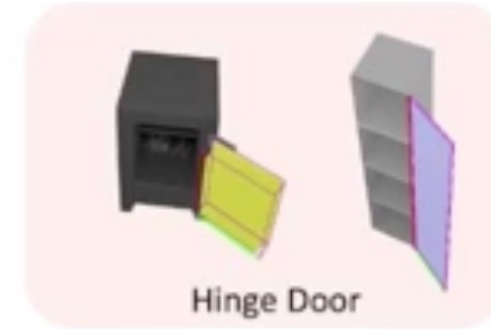
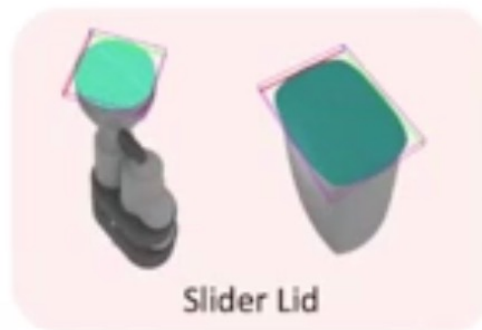
PartNet-Mobility
Object-Centric

GAPartNet
Part-Centric

9 part classes
8,489 parts

from

1166 objects
27 object
categories



Real-World Experiment

Inputs

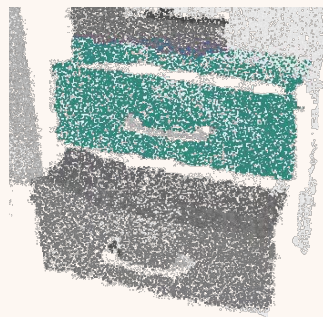


point cloud



RGB image

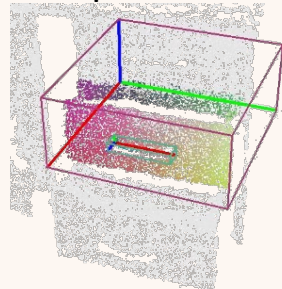
Predictions



segmentation



NPCS prediction

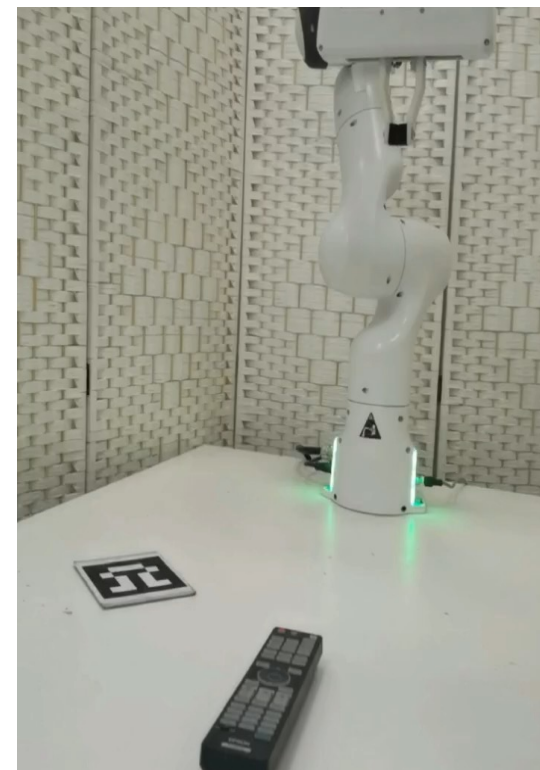
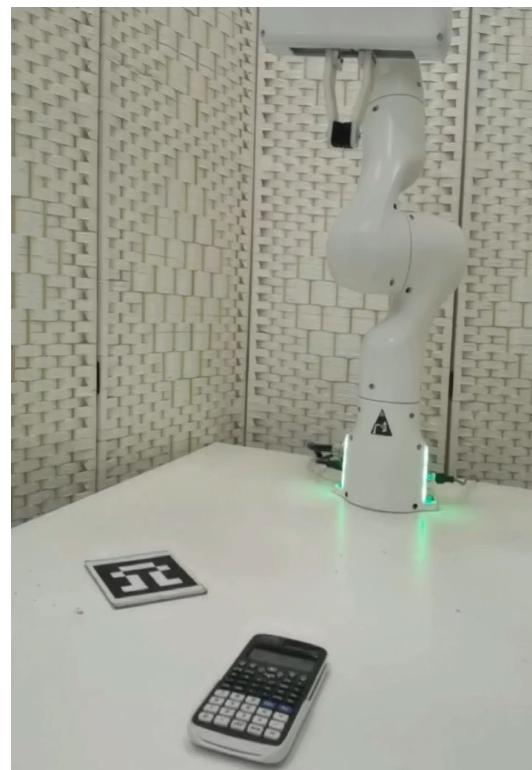
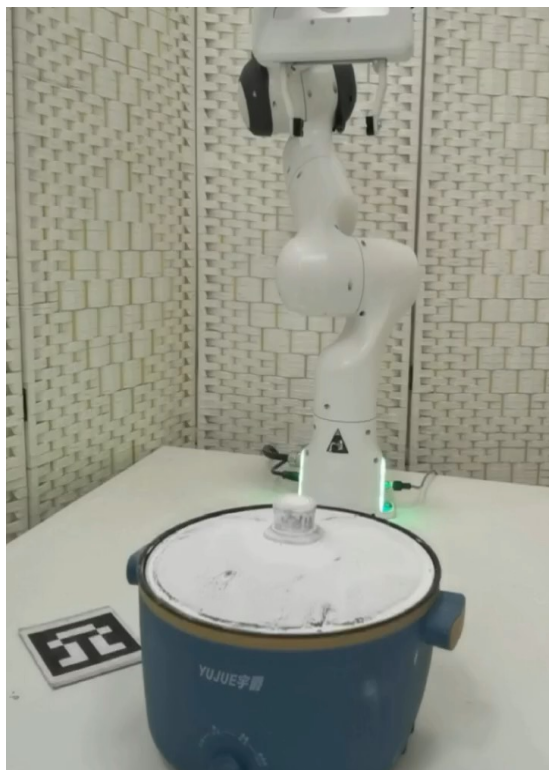
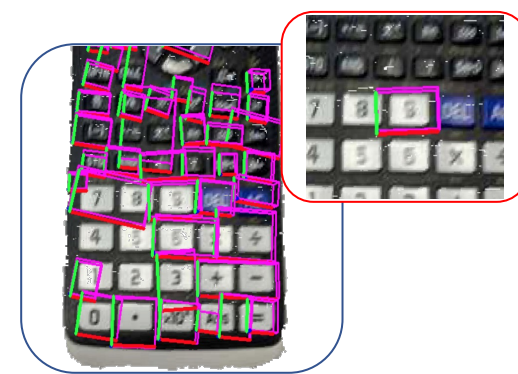
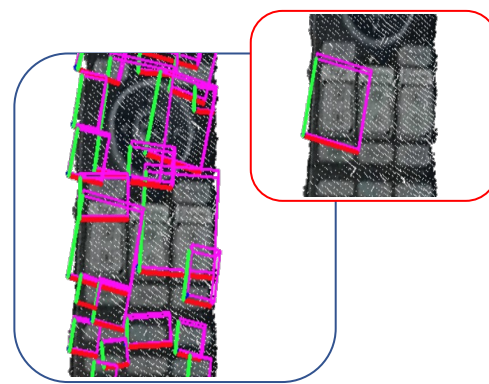
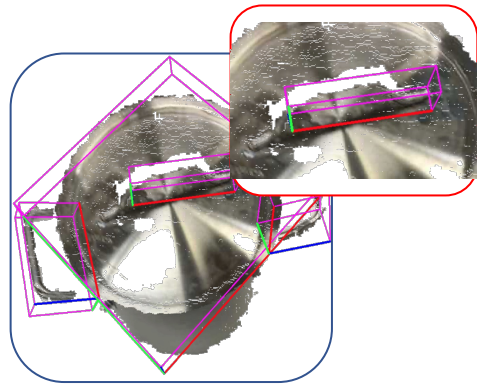
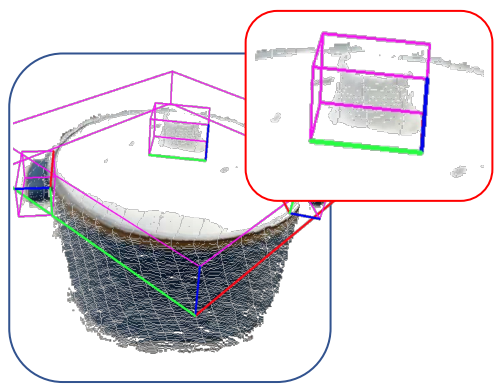


bounding box

Interaction Video

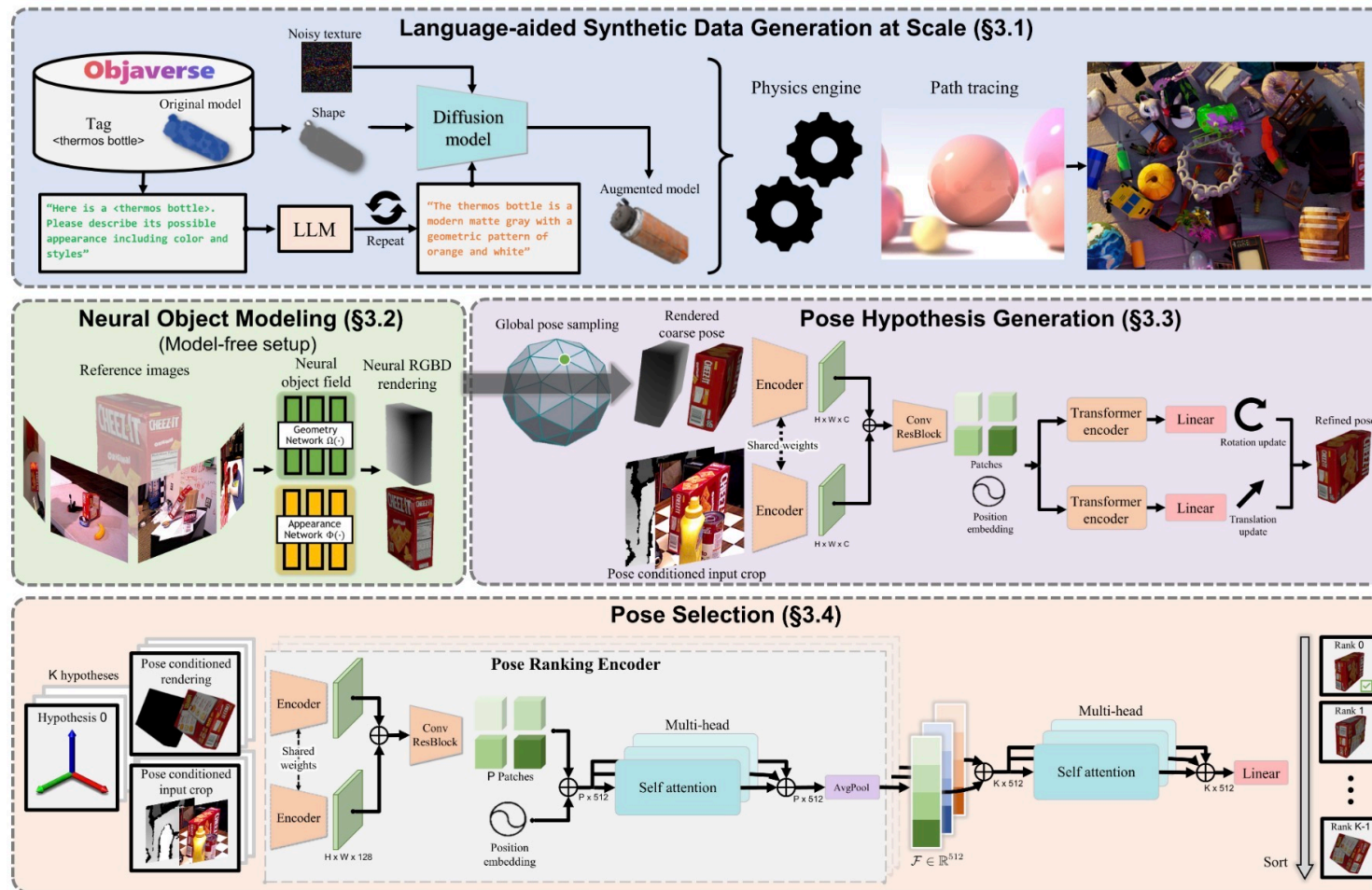






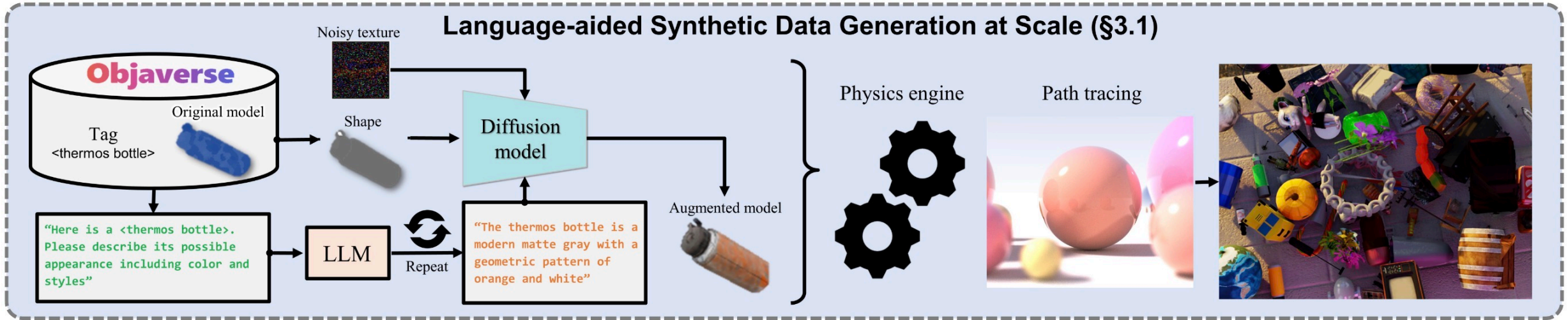
Foundation Model for 6D Object Pose Estimation

Foundation Model for 6D Pose Estimation



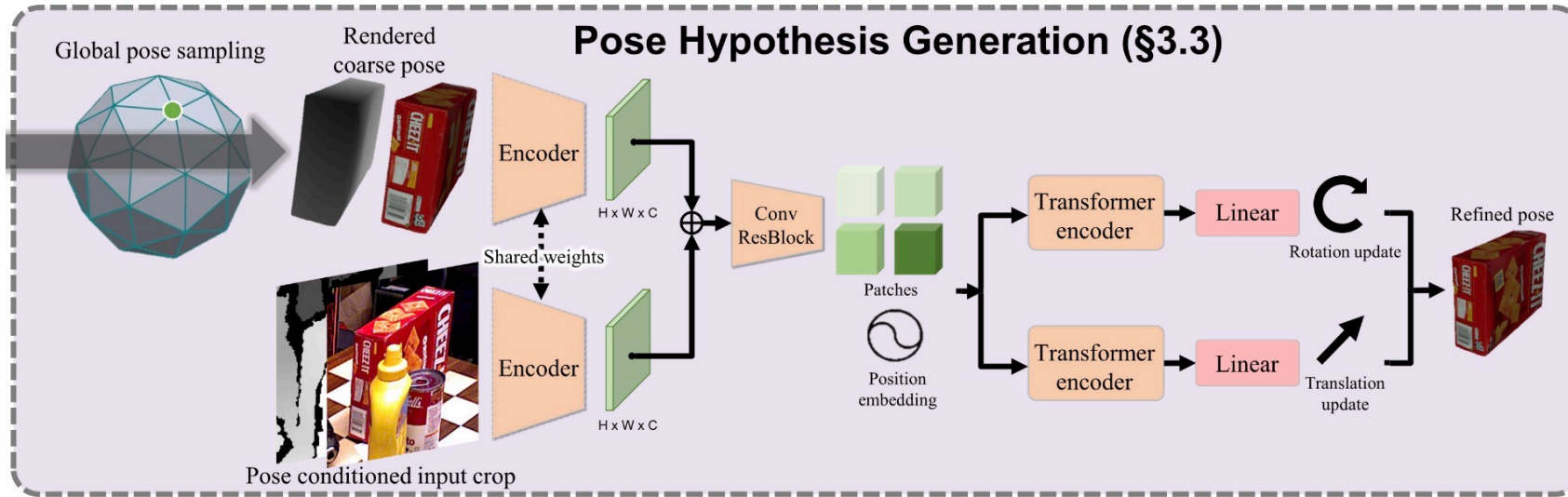
FoundationPose, a unified foundation model for 6D object pose estimation and tracking, supporting both model-based and model-free setups.

Foundation Model for 6D Pose Estimation



- Obtaining data in the real world is time- and cost-prohibitive.
- Synthetic data often lacks the size and diversity in 3D assets.
- A novel synthetic data generation pipeline using large language models, and diffusion models.

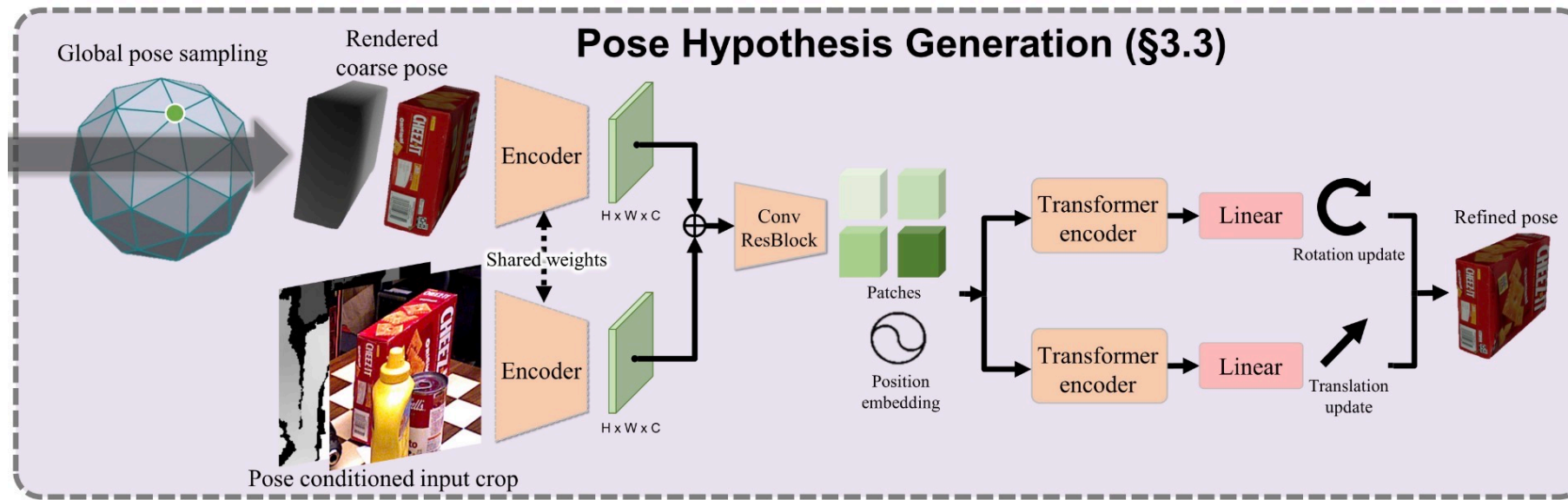
Foundation Model for 6D Pose Estimation



- **Pose Initialization**

1. Give the RGBD image, the object is detected using Mask RCNN or CNOS model.
2. Initialize the translation using the 3D point located at the center of the detected 2D bounding box along with the median depth.
3. Initialize rotations by sampling N viewpoints from an icosphere centered on the object with the camera facing the center.

Foundation Model for 6D Pose Estimation

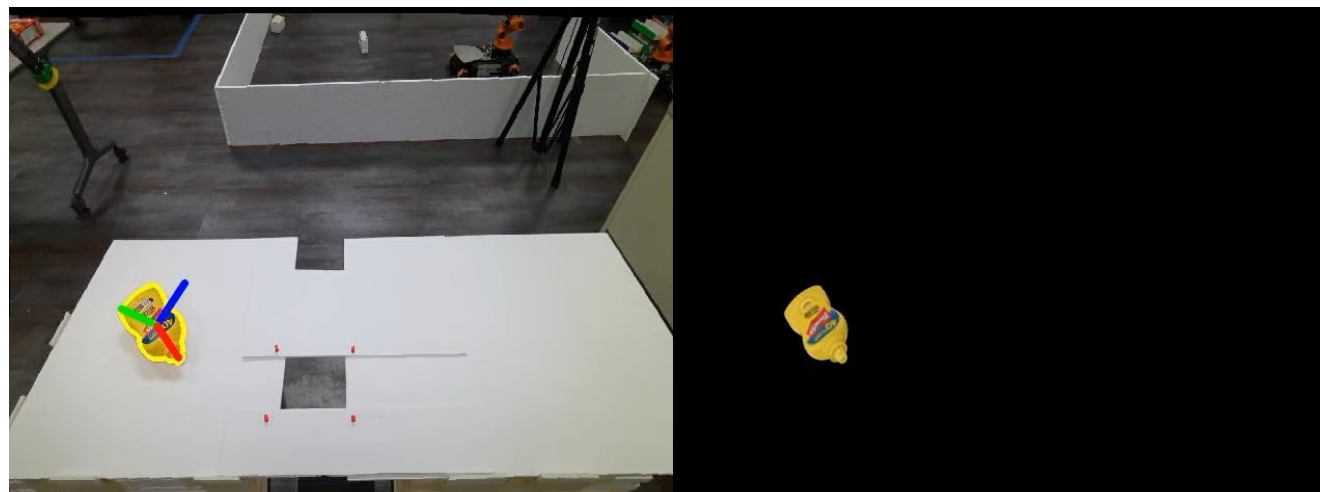


- **Pose Conditioned Input Crop**

1. Take the object rendering with coarse pose and a crop of the input, and output a refined pose.
2. Project the object origin to the image space as the crop center, and the enlarged object diameter as the crop size, enclosing the object and nearby context.
3. This cropping strategy helps the network update the translation for better alignment with the observation.

Foundation Model for 6D Pose Estimation

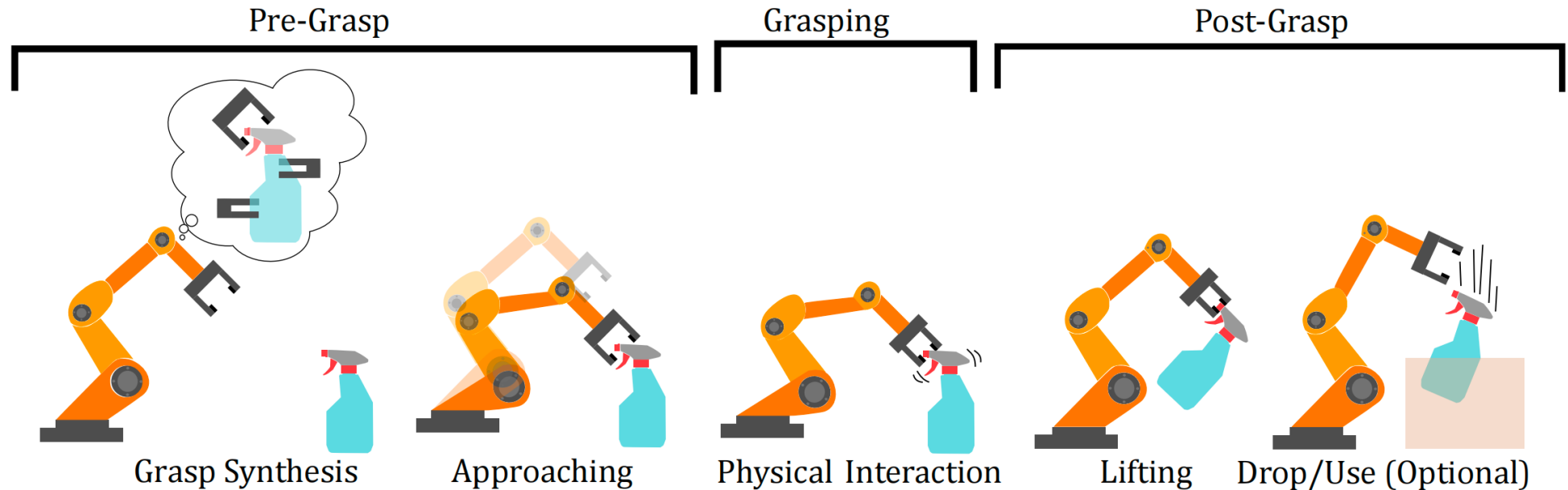
- Robot Application
- Result on YCB-Video Dataset.



Object Grasping

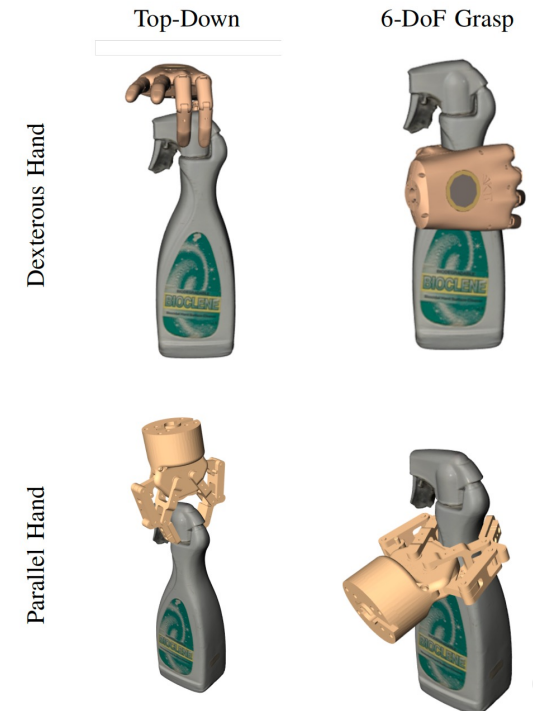
What is Grasp Synthesis?

- **Grasping** is the process of restraining an object's motion in a desired way by applying forces and torques at a set of contacts
- **Grasp Synthesis** is a high-dimensional search or optimization problem to **find gripper poses** or joint configurations



Terminology

- **Grasp Pose** defines the position and orientation of a hand
 - **4-DoF grasp**: a 3D position and 1D hand orientation aligned with the direction of gravity, a.k.a. “top-down grasping”
 - **6-DoF grasp**: defined by a 3D position and 3D orientation
- **Force Closure** means if the forces that a grasp can be applied at the set of frictional contacts are sufficient to compensate for any external wrench applied to the object

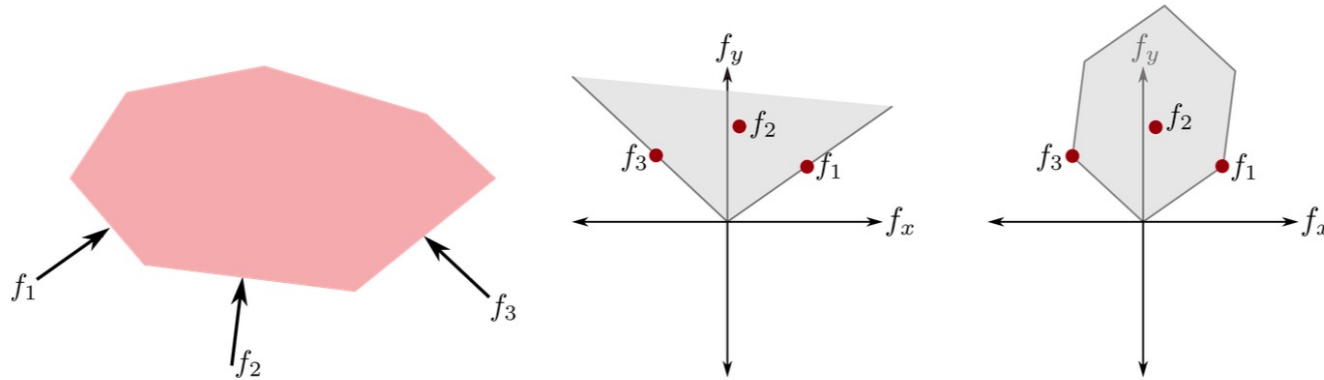


Force Closure

- In physics,
 - given a set of frictional contacts acting on a body, it is in **force closure** if the positive span of the wrench cones is the entire wrench space
 - if a rigid body is fully immobilized by a set of rigid stationary fixtures, we say it is in **form closure**
- The conditions for **force closure** are identical to the conditions for **first-order form closure** if all contacts are frictionless.
- When planning a grasp by a robot hand, **force closure** is a good minimum requirement. **Form closure** is usually too strict, requiring too many contacts.
- Simply, successful grasp $\Rightarrow$ form closure $\Rightarrow$ force Closure

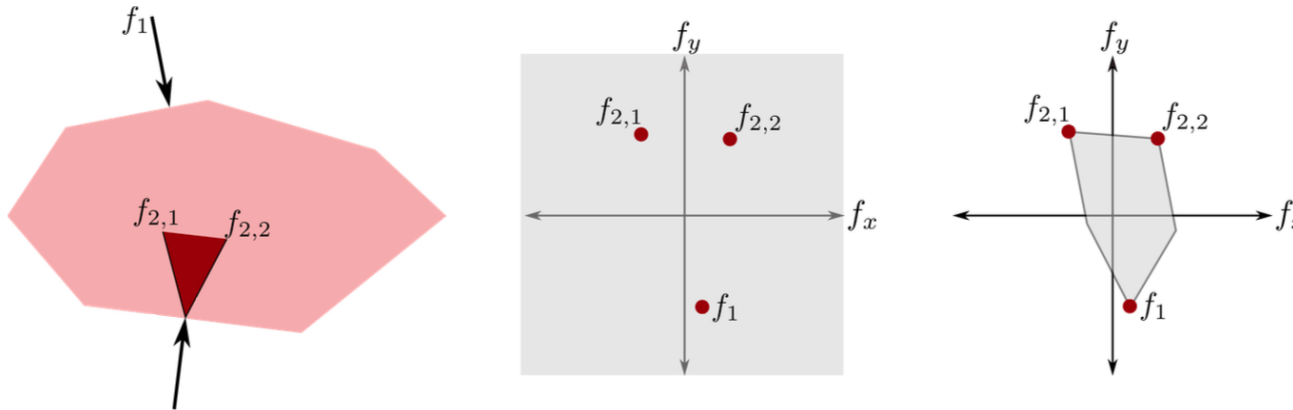
Force Closure

- A simplified 2D problem



A 2D grasp with frictionless contacts

- ✓ 3 possible force vectors with 2 linearly independent
- ✗ generate zero force with non-zero forces at each contact
- ✗ The origin is contained in the interior of the space of possible applied forces.



A 2D grasp consisting of a contact with friction and a contact without friction

- ✓ 3 possible force vectors with 2 linearly independent
- ✓ generate zero force with non-zero forces at each contact
- ✓ The origin is contained in the interior of the space of possible applied forces.

- Watch: <https://www.youtube.com/watch?v=6RWFFMtD5k8>



Introduction to Embodied AI

Next week: Lecture 7, Vision and Grasping III

