PACE: Pose Annotations in Cluttered Environments

Yang You[1,3*] Kai Xiong[1] Zhening Yang[2] Zhengxiang Huang[1] Junwei Zhou[1]
Ruoxi Shi[1] Zhou Fang[1] Adam W. Harley[3] Cewu Lu[1†]
[1]Shanghai Jiao Tong University [2]Horizon Robotics Inc. [3]Stanford University

Abstract

Pose estimation is a crucial task in computer vision, enabling tracking and manipulating objects in images or videos. While several datasets exist for pose estimation, there is a lack of large-scale datasets specifically focusing on cluttered scenes with occlusions. This limitation is a bottleneck in the development and evaluation of pose estimation methods, particularly toward the goal of real-world application in environments where occlusions are common. Addressing this, we introduce PACE (Pose Annotations in Cluttered Environments), a large-scale benchmark designed to advance the development and evaluation of pose estimation methods in cluttered scenarios. PACE encompasses 54,945 frames with 257,673 annotations across 300 videos, covering 576 objects from 44 categories and featuring a mix of rigid and articulated items in cluttered scenes. To annotate the real-world data efficiently, we developed an innovative annotation system utilizing a calibrated 3-camera setup. We test state-of-the-art algorithms in PACE along two tracks: pose estimation, and object pose tracking, revealing the benchmark's challenges and research opportunities. We plan to release PACE as a public evaluation benchmark, along the annotations tools we developed, to stimulate further advancements in the field. Our code and data is available on https://github.com/qq456cvb/PACE.

1. Introduction

The field of 3D object pose estimation is integral to a myriad of applications, particularly within robotic manipulation. Recent advancements in both instance and category-level pose estimation have been significant, bolstered by deep learning approaches, and perhaps more importantly, *data*.

PoseCNN [45] advanced pose estimation into the deep learning era, and simultaneously introduced the influential YCB-Video dataset. This benchmark has catalyzed methodological development and offered a consistent evaluation platform. Additionally, the Benchmark for 6D Object Pose Estimation (BOP) challenges have consolidated datasets and refined evaluation metrics for instance-level pose estimation.

	Bottle	Bowl	Camera	Can	Laptop	Mug
SAR-Net [24]	54.0	66.0	0.4	62.2	50.0	21.2
HS-Pose [47]	51.1	94.7	3.9	75.4	85.2	26.4

Table 1. Performance saturation on current benchmarks is evident with AP@5°5cm results, HS-Pose's performance is nearly saturated on bowl and laptop, making one curious how it performs in general.

In parallel, the NOCS [41] dataset has addressed category-level pose estimation, albeit with a smaller real-world dataset for validation and testing. Despite these strides, the field grapples with a fundamental challenge: existing evaluation datasets are too constrained to thoroughly benchmark the capabilities of pose estimation algorithms. The NOCS REAL275 dataset, for instance, spans only six categories and includes a mere 18 videos. This limitation has led to a performance saturation on current benchmarks, as depicted in Table 1, rendering it ambiguous whether methods are getting tuned to the dataset or improving in general.

In this work, we introduce PACE (Common Objects with Pose Annotations), a benchmark for pose estimation, and present a comprehensive study evaluating a wide range of pose estimation and tracking methods. Our contributions are threefold:

- The PACE dataset: This dataset includes 576 objects across 44 categories, captured in 300 video clips within diverse scenes. With an average of 183 frames per clip, the dataset encompasses 54,945 frames and 257,673 annotations, providing a large-scale benchmark for pose estimation.
- Our evaluation study: To the best of our knowledge, we are the first to analyze and report the performance of state-of-the-art pose estimation methods in a large-scale cluttered setting. These results provide valuable insights that the scalability and generalizability of state-of-the-art methods, making it clear that they are far from reliable in general.
- Our annotation pipeline: We will open-source our annotation pipeline, which harnesses a calibrated 3-camera system, enhancing the precision and scalability of anno-

*Work done at Shanghai Jiao Tong University.
†Cewu Lu is the corresponding author.

Figure 1. Sample images from the PACE dataset showcasing the diversity of objects, complexity of scenes, and the range of occlusions. These examples highlight the dataset's real-world applicability for robust pose estimation benchmarks.

tating poses in real data. This tool significantly mitigates human error and reduces the effort required for annotating 3D poses, providing a solution to one of the major bottlenecks in pose dataset creation.

Overall, our work aims to propel the development of more robust and generalizable pose estimation techniques, thereby facilitating progress towards successful pose estimation in the real world.

2. Related Works

The field of 3D object pose estimation has seen substantial progress over the past few years. This progress has been facilitated by the introduction of standardized datasets and the development of innovative algorithms.

2.1. Object Pose Datasets

Instance-Level Pose Datasets YCB-Video dataset [45] is a comprehensive resource for 6D object pose estimation, containing a large number of video frames with accurate pose annotations for 21 objects. LINEMOD-Occluded dataset [3] offers a challenging setting for pose estimation with piled multiple objects in occluded scenes. NAVI dataset [16] presents casually captured images of objects with high-quality 3D scans and precise 2D-3D alignments for advanced 3D reconstruction tasks.

Category-Level Pose Datasets CO3D [32] offers 1.5 million frames from nearly 19,000 videos across 50 MS-COCO categories for category-specific 3D reconstruction and view synthesis. The Scan2CAD [2] dataset aligns 14225 CAD models from ShapeNet to 1506 ScanNet scans, promoting CAD model alignment in RGB-D scans. Pix3D [36] is a benchmark with image-shape pairs and pixel-level 2D-3D alignment, aiding in shape reconstruction and retrieval. The HOI-4D dataset [29], with 2.4M RGB-D frames over 4000 sequences, enables research in category-level human-object interaction. HANDAL [11] focuses on pose estimation and affordance prediction for robotics-ready manipulable objects. A comparative comparison with other datasets is in Table 2.

2.2. Pose Estimation Methods

Instance-level Pose Estimation PPF (Point Pair Features) [8] set the pre-deep learning standard for instance-level pose estimation, utilizing local geometric features from point clouds. The deep learning era began with PoseCNN [45], leading to several advanced methods. DeepIM [23] approached pose estimation as an image matching task, iteratively refining estimations. DenseFusion [38] combined global and local features for pose estimation in cluttered scenes, while CosyPose [19] integrated a global refinement strategy in its end-to-end pipeline. SurfEmb [12] leveraged surface embeddings for correspondence matching, and GDRNPP [40] employed geometry-guided regression for enhanced prediction.

Category-level Pose Estimation Category-level pose estimation extends the challenge to generic object categories. NOCS [41] introduced a unified coordinate space for all objects, predicting object NOCS maps from RGB images. SGPA [4] aims to adapt the structure-guided prior in the pose estimation process, while SAR-Net [24] using shape alignment and symmetric correspondence to estimate a coarse 3D object shape and facilitate object center and size estimation. Recently, HS-Pose [47] proposes a network structure with an HS-layer that extends 3D graph convolution to extract hybrid scope latent features from point clouds for category-level object pose estimation.

2.3. Pose Tracking Methods

Instance-level Pose Tracking Methods like RBOT [37] use RGB data and 3D models to track multiple objects, employing color histograms in their cost function. PoseRBPF [5] separates rotation and translation, using an autoencoder for

	Modality	Cat.	Obj.	Vid.	Img.	Anno.	CAD	Dyn.	Occ.	Marker-free	Artic.	Piled
YCB-Video [45]	RGBD	1	21	12	20K	99K	✓	✗	✓	✓	✗	✓
LINEMOD-O [3]	RGBD	1	8	1	1.2K	9.2K	✓	✗	✓	✗	✗	✓
NAVI [16]	RGBD	1	36	324	10K	10k	✓	✗	✗	✓	✗	✗
NOCS-REAL275 [41]	RGBD	6	42	18	8K		✓	✗	✓	✗	✗	✗
Wild6D [46]	RGBD	5	1722	5166	1.1M	1.1M	✗	✗	✗	✓	✗	✗
Objectron [1]	RGB	9	17k	14k	4M	4M	✗	✗	✗	✓	✗	✗
CO3D [32]	RGB	50	19K	19K	1.5M	1.5M	✓	✗	✗	✓	✗	✗
Scan2CAD [2]	RGBD	9	3K	1506	-	14K	✓	✗	✓	✓	✗	✗
Pix3D [36]	RGBD	9	395	-	10K	10K	✓	✗	✗	✓	✗	✗
HOI-4D [29]	RGBD	16	800	4K	2.4M	-	✓	✓	✓	✓	✓	✗
HANDAL [11]	RGB	17	212	2K	308K	308K	✓	✓	✓	✓	✗	✗
PACE (Ours)	RGBD	44	576	300	55K	258K	✓	✓	✓	✓	✓	✓

Table 2. **Comparison of object pose datasets.** From left to right, the table captures the input modality, number of categories, number of instances, number of videos, number of images, number of total annotations, whether 3D CAD models are provided, whether videos include static and/or dynamic moving objects, whether objects are occluded in some frames, whether images contain artificial markers, whether poses for each part of articulated objects are provided, and whether multiple objects are piled in some frames. Compared with most previous datasets, our dataset contains dynamic and articulated objects.

rotation feature embeddings. ICG [34] iteratively refines pose using geometric cues and is effective for textureless objects, with extensions incorporating visual data [35]. The first deep learning tracker, D6DT [9], and se(3)-TrackNet [43] predict frame-to-frame relative poses, using a render-and-compare strategy.

Category-level Pose Tracking 6-PACK [39] marks the onset of category-level tracking, using DenseFusion [38] features and an attention mechanism for unsupervised keypoint ordering and interframe motion via keypoint matching. CenterPoseTrack [26] projects 2D keypoints from 3D bounding box vertices, achieving RGB-based scale-invariant tracking. BundleTrack [42] generalizes pose tracking without relying on 3D models, instead using video segmentation and pose graph optimization. CAPTRA [44] tracks 9DoF poses for rigid and articulated objects, with subnetworks for rotation regression and normalized coordinate prediction, facilitating analytical 3D size and translation computation.

3. Construction of PACE

A key contribution of this work is the establishment of a **scalable** and **reliable** annotation framework, enabling the collection of large-scale and accurate pose annotations. An overview of the pipeline is depicted in Figure 2.

3.1. Acquisition of 3D Common Object Scans

We begin by digitizing an extensive collection of commonplace objects. These items are categorized into 44 distinct classes, as represented in Figure 3.

The Einscan Pro 2X is utilized for rapid scanning of all objects, typically completing within 5 to 10 minutes per object. To expedite this process, we employ a rotatable platform to acquire multiple viewpoints of the objects. After scanning, objects are manually aligned to a standard pose within a uniform coordinate system. We center its axis-aligned bounding box at the origin and align the bounding box orientation along a common axis within the category. We annotate rotational symmetries with the corresponding rotation matrices. High-resolution meshes are then simplified to lower-resolution for smoother annotation workflows.

Articulated Objects Diverging from many prior pose estimation datasets, our collection encompasses a very wide set of objects, including articulated objects from the AKB48 [27] dataset, namely: **scissors**, **cutters**, **clips**, and **boxes**. We adopt the alignment methodology from the original AKB48 dataset, without modification. These objects are segmented into multiple parts with hierarchical relationships, presenting a nuanced challenge for pose estimation.

3.2. RGB-D Sequence Acquisition

We designed and implemented a 3-camera system to aid in data acquisition and annotation, comprising three Intel Realsense 415 RGB-D cameras affixed to a metal framework, as illustrated in the bottom of Figure 2. The advantages of this setup include:

- Tripling the data yield.
- Reducing ambiguity in pose annotation, especially regarding translation along the depth dimension, by using multi-view imagery to enforce consistency across all views.
- Enhancing tracking accuracy of static objects with Aruco markers from all three views, making PnP more stable.

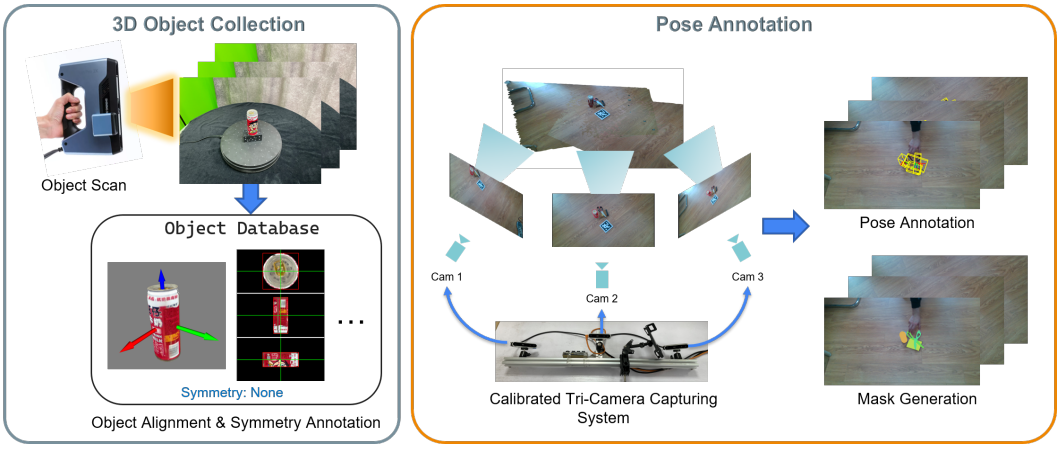


Figure 2. Overview of the PACE annotation pipeline.

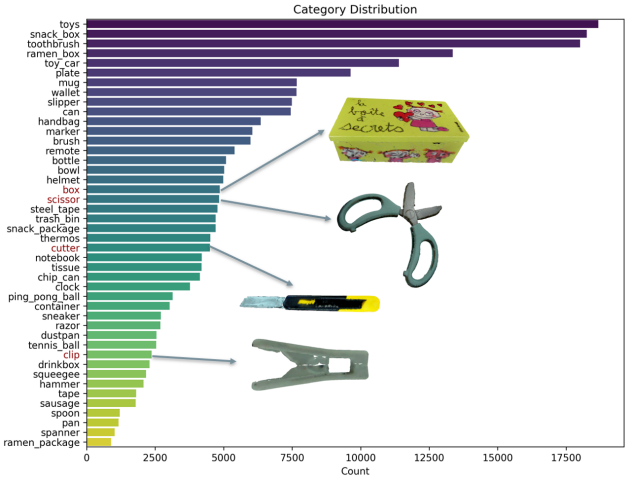


Figure 3. The distribution of object categories within the collected dataset. Articulated objects are marked in red.

Calibration of Multi-Camera Extrinsic Parameters We calibrate this multi-camera system through a semi-automatic process. Aruco markers initially intended for calibration proved insufficient for high-accuracy rotation estimation. Hence, we resorted to trifocal tensor estimation, i.e. TFT [17]. The process begins with feature extraction and matching, followed by bundle adjustment to refine the positions of the 3D landmarks and camera poses. For reliable feature matching, we employ the SuperPoint [7] descriptor and SuperGlue [33] matcher, using a stringent threshold for matching. We observe that rotational component of the resulting extrinsic parameters is generally precise, but the translation aspect suffers from scale ambiguity inherent in Structure-from-Motion approaches. We correct this by calibrating the scale against markers, applying the following formula to obtain the optimal scale factor:

$$s_{1\to2}* = \frac{\hat{\mathbf{t}}_{1\to2}\cdot\mathbf{t}'_{1\to2}}{\hat{\mathbf{t}}_{1\to2}\cdot\hat{\mathbf{t}}_{1\to2}}, s_{1\to3}* = \frac{\hat{\mathbf{t}}_{1\to3}\cdot\mathbf{t}'_{1\to3}}{\hat{\mathbf{t}}_{1\to3}\cdot\hat{\mathbf{t}}_{1\to3}},$$

where $\hat{\mathbf{t}}_{i\to j}$ is the TFT predicted translation (up to a scale) from camera i to camera j, $\mathbf{t}'_{i\to j}$ is the marker-calibrated translation in real metric scale. We set the extrinsics of the first camera to be the identity matrix.

In cases of repetitive texture patterns, manual intervention is required to establish reliable feature correspondences due to the limitations of SuperPoint+SuperGlue.

3.3. Annotation of Pose Ground-Truths

Previous methodologies have utilized Aruco markers to automate pose estimation through Perspective-n-Points; however, this approach has two drawbacks:

1. Markers in the scene detract from realism and compromise dataset integrity: training on marker-augmented imagery may result in overfitting to these artificial patterns.
2. Marker-based annotations are inapplicable to dynamic objects, thus restricting the method's utility to static scenarios.

Annotation of Static Object Poses To address the first issue, we employ markers to automate the annotation of static object poses, and then *remove* the marker appearances from the dataset. We achieve this with a marker inpainting strategy, detailed as follows. Initially at step 1, we place a marker (Marker 1) somewhere within the camera's field of view. We then record a short video with this marker in view. In step 2, we place a second marker (Marker 2) at a chosen distance from the first, and record another video. After step 2, we remove Marker 1, and begin the actual object capture process (with only Marker 2 present). After this process, we

end up with: (1) frames with Marker 1 only, which clearly depict the surface where Marker 2 will later appear; (2) frames capturing both markers, providing helpful calibration cues; (3) frames with Marker 2 only, which represent our main capture. We use the frames from the first two steps to seamlessly inpaint [30] Marker 2's area in the main capture, as depicted in Figure 4. We leverage Marker 2 for automated pose tracking, and manually correct the tracking every 40 frames in case of drift.

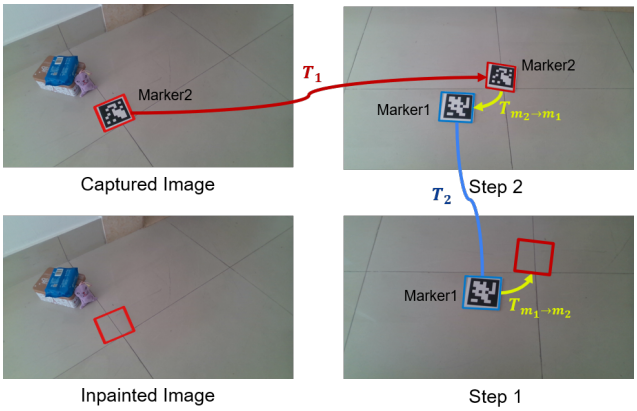


Figure 4. Illustration of the marker inpainting process.

Annotation of Dynamic Object Poses For dynamic objects, the traditional marker-based tracking approach is insufficient, as the scene markers do not move in tandem with the objects. In such instances, we harness the capabilities of BundleTrack [42], an advanced RGB image-based tracking algorithm. BundleTrack conducts feature correspondence analysis between successive frames to estimate poses, complemented by a bundle adjustment algorithm to optimize keyframes globally and minimize tracking errors. Despite BundleTrack's proficiency in approximating poses, it is prone to drift, necessitating the manual adjustment of poses every ten frames to ensure precision. This delicate step represents the most labor-intensive aspect of our annotation pipeline, given the current limitations of state-of-the-art tracking techniques in complex scenarios.

Generation of Segmentation Masks Upon successful pose annotation, we generate occlusion-aware segmentation masks, using z-buffer rendering techniques. In cases where hand interactions are involved, we employ the SAM [18] model to delineate hand masks, and subsequently subtract these from the previously computed object masks to achieve accurate segmentation.

4. Dataset Statistics

In pursuit of diversity, we placed the objects ten disparate environments, each featuring varying levels and configurations of occlusion. We document 10 video sequences per scene, each sequence encompassing 1 to 5 objects. We captured RGB and depth data using an Intel RealSense D415 camera, with a resolution of 1280×720. The capture process spans distances ranging from 0.5m to 1.5m from the objects.

4.1. Object Distribution and Diversity

The comprehensive distribution of objects is depicted in Figure 3, showcasing a diverse array of both rigid and articulated models. As demonstrated in Figure 5, the dataset encompasses a broad spectrum of object sizes, with the majority measuring approximately 0.2m along the bounding box diagonal. This includes larger items such as storage bins, adding to the heterogeneity of the dataset.

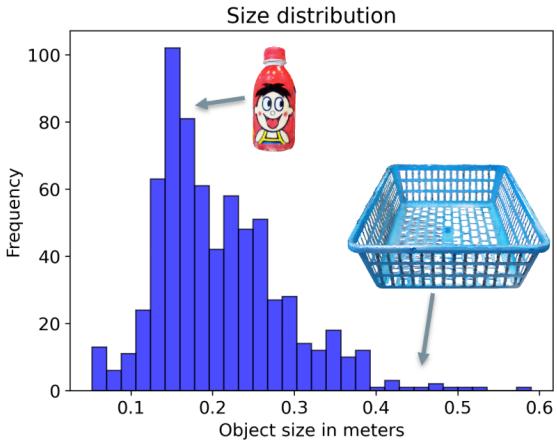


Figure 5. Distribution of object sizes within the dataset, indicating the prevalence of small to medium-sized objects and the inclusion of larger items.

4.2. Variability in Pose, Occlusion, and Environmental Context

The dataset is characterized by a rich variability in object poses, as illustrated in Figure 6, which outlines the statistical distribution of azimuth and elevation angles, indicating comprehensive spatial coverage.

Moreover, we categorized and analyzed the occlusion levels, as shown in Figure 7, delineating instances of severe, moderate, and minor occlusions. Such classification is crucial for assessing the robustness of pose estimation algorithms against varying degrees of visibility.

A visual representation of the ten distinct environmental settings used for data capture is provided in Figure 8, reflecting the contextual diversity of the dataset.

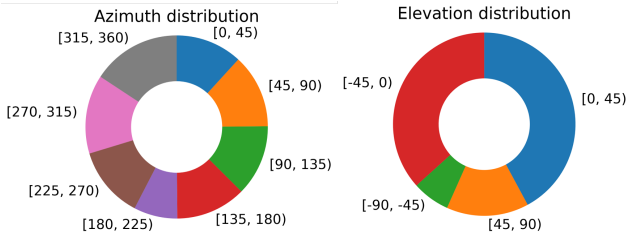


Figure 6. Statistical distribution of object poses, highlighting the diversity in azimuth and elevation angles.

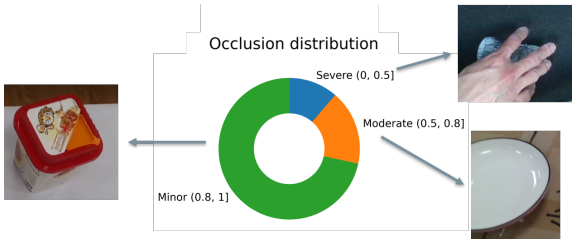


Figure 7. Distribution of occlusion levels within the dataset, which are critical for evaluating pose estimation performance in real-world conditions.

4.3. Dataset Split for Training, Validation, and Testing

To facilitate a comprehensive and equitable evaluation of pose estimation methodologies, we partition the dataset into validation and test subsets following a 20/80 ratio. This separation ensures that a substantial volume of data is available for rigorous testing.

Additionally, to support research necessitating extensive training datasets, we generate a considerable set of synthetic images utilizing a physically based renderer (PBR) [6]. The training set comprises over 52,000 images, each featuring multiple objects, culminating in a vast number of annotations. Qualitative examples of the synthetic images are presented in Figure 9.

5. Evaluation Benchmarks

State-of-the-art pose estimation algorithms typically decompose the task into two distinct stages: detecting or segmenting the object of interest, followed by pose estimation within the predicted bounding box or mask. While the latter has been the primary focus in literature, often leveraging off-the-shelf detectors, a comprehensive evaluation requires an isolated assessment of each stage. Therefore, we examine the pose estimation result under the assumption of perfect detection to allow for equitable comparison across methods. The object detection performance of current state-of-the-art methods can be found in the supplementary and is not the focus of this dataset. We also introduce a benchmark for object pose tracking, premised on the availability of ground-truth data in the initial video frame.

5.1. Pose Estimation Benchmark

This benchmark is divided into instance-level and category-level pose estimation. The former concerns known instances during training, while the latter involves unknown instances similar to those in the training set.

5.1.1 Instance-Level Pose Estimation

This task demands the prediction of rotation and translation for known instances from the training set.

Metrics: Adhering to the BOP challenge protocol [15], we utilize Average Recall (AR) of Visible Surface Discrepancy (VSD), Maximum Symmetry-Aware Surface Distance (MSSD), and Maximum Symmetry-Aware Projection Distance (MSPD) as our metrics. Detailed computation of these metrics is described in [15]. Objects with visibility fraction less than 10% are skipped for evaluation following the BOP [15] convention. Results are averaged across all instances.

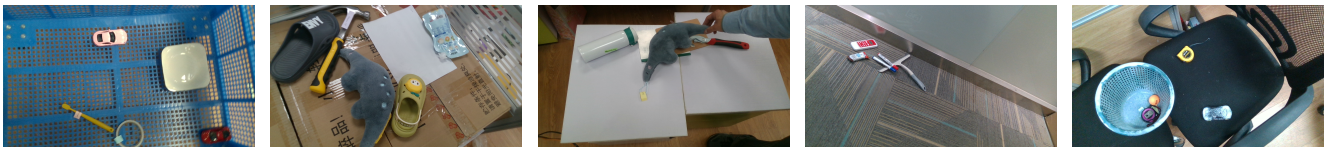
Baselines: We assess four baselines: PPF [8], CosyPose [19], SurfEmb [12], and GDRNPP [40]. While PPF does not require training data, the others are state-of-the-art methods dependent on additional training on the PBR dataset. To focus on pose estimation performance, we assume that ground-truth instance detections are available.

Result Analysis: The quantitative analysis is presented in Table 3. Notably, while state-of-the-art methods excel on the BOP benchmarks, they do not perform as well as PPF, which relies on local geometric feature matching in point clouds. The relative success of PPF highlights the potential for improvement in state-of-the-art methods, particularly regarding robustness and scalability to handle a large set of instances. Qualitative results can be found in the supplementary.

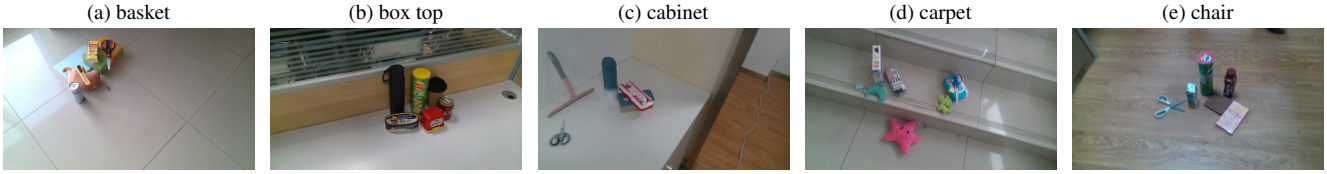
5.1.2 Category-Level Pose Estimation

Category-level pose estimation tasks involve predicting the 3D bounding box dimensions, rotation, and translation of target instances, given only the category information a priori.

Metrics: We adopt from NOCS [41], calculating mean Average Precision (AP) across predefined angle and translation thresholds. Specifically, AP@0:20° represents AP averaged from 0° to 20° at 1° intervals; AP@0:5cm is averaged from 0cm to 5cm at 0.25cm intervals; AP@0:20°,0:5cm combines these angle and translation thresholds. IoU_{25} and IoU_{50} denote AP for 3D bounding box matches at IoU thresholds of 25 and 50, respectively. Objects with visibility fraction less than 10% are skipped for evaluation following the BOP [15]



(a) basket (b) box top (c) cabinet (d) carpet (e) chair



(f) china floor (g) desk1 (h) desk2 (i) stairs (j) wooden floor

Figure 8. Sample images from the 10 environments.

	Modality	Detection	$AR_{VSD}\uparrow$	$AR_{MSSD}\uparrow$	$AR_{MSPD}\uparrow$	AR↑
PPF [8]	D	G.T.	**35.3**	**42.7**	**49.3**	**42.4**
CosyPose [19]	RGB	G.T.	1.4	0.3	11.5	4.4
SurfEmb [12]	RGB	G.T.	6.2	3.0	17.8	9.0
GDRNPP [40]	RGB-D	G.T.	3.6	2.1	15.4	7.0

Table 3. **Instance-level pose estimation results** showcasing the robustness of PPF and the potential for improvement in state-of-the-art deep learning-based methods.

Figure 9. Example images from the synthetic dataset generated with a physically based renderer (PBR).

convention. Results are separately reported for both rigid and articulated objects.

Baselines: We evaluate six recent category-level pose estimation methods: NOCS [41], HS-Pose [47], SGPA [4], DualPoseNet [25], SAR-Net [24], and ANCSH [22], the latter specifically designed for articulated objects. For adapting rigid-object-focused methods to articulated objects, each movable part is considered a distinct category. For instance, scissors with two movable parts are treated as two separate categories. Ground-truth detections are presumed for each method, except for NOCS, which outputs both instance masks and pose estimations using a unified network. The compared methods all use the depth as input.

Result Analysis: Quantitative comparisons are listed in Table 4. On rigid objects, HS-Pose shows excellent performance on IoU_{25} and IoU_{50}, as well as AP metrics for translation, but falls short on rotation AP metrics compared to SGPA, suggesting a proficiency in size and translation prediction but not rotation. SGPA excels in rotation estimation but demonstrates lower performance on bounding box size metrics. For articulated objects, all methods face challenges due to the movable parts, with ANCSH underperforming on the large-scale real-world dataset, indicative of a significant sim-to-real gap. NOCS shows instability, likely due to its reliance on RGB-based coordinate prediction, which lacks robustness and scalability. Qualitative results are available in supplementary.

5.2. Pose Tracking Benchmark

We categorize state-of-the-art pose tracking methods into model-free, which use a 3D CAD model, and model-based, requiring only the initial pose at the first frame.

Metrics: For model-free pose tracking, we use four metrics following previous work [39]. 1) $\mathbf{5°5cm}$, the percentage of predictions with rotation error $< 5°$ and translation error < 5cm. 2) **IoU25**, the percentage of intersection over union that is larger than 25% between the two 3D bounding boxes

	Detection	IoU_{25}↑	IoU_{50}↑	AP					
				0:20°↑	0:60°↑	0:5cm↑	0:15cm↑	0:20° 0:5cm ↑	0:60° 0:15cm ↑
NOCS [41]	Mask-RCNN	0.0/0.0	0.0/0.0	0.0/0.0	0.1/0.0	2.8/0.0	26.1/0.0	0.0/0.0	0.1/0.0
HS-Pose [47]	G.T.	**32.7**/0.2	**7.3**/0.0	5.2/0.0	7.0/0.4	**61.4/41.0**	**86.2/79.5**	**4.0**/0.0	6.6/0.4
SGPA [4]	G.T.	1.3/0.0	0.0/0.0	**6.7/1.1**	**13.4/9.0**	18.5/10.4	58.4/49.2	3.3/**0.3**	**11.7/7.7**
DualPoseNet [25]	G.T.	0.1/0.0	0.0/0.0	5.1/0.0	5.3/0.0	15.0/34.6	53.6/69.4	1.4/0.0	4.1/0.0
SAR-Net [24]	G.T.	25.3/**0.5**	1.1/0.0	5.2/0.1	7.1/2.6	37.6/37.1	77.3/77.9	2.8/0.1	6.3/2.6

Table 4. **Category-level pose estimation benchmark** combining the performance metrics for both rigid and articulated object pose estimation, separated by slash.

with ground-truth size, transformed by the predicted and ground-truth 6D pose, respectively. 3) $\mathbf{R_{err}}$, mean value of rotation error in degrees. 4) $\mathbf{T_{err}}$, mean value of translation error in centimeters. Here the last two metrics are respect to IoU25 since objects with IoU$\leq 25\%$ are not counted.

For model-based pose tracking, we report the area under curve (AUC) with respect to ADD, ADD-S [45] and ADD(-S). We set the maximum threshold of AUC to be 0.1m [38]. The ADD metric is first introduced in [14] to calculate average per-point distance between two point clouds, transformed by the predicted pose and the ground-truth, respectively. For symmetric objects like bowls, ADD-S metric is introduced to count for the point correspondence ambiguity. The notation ADD(-S) corresponds to computing ADD for non-symmetric objects and ADD-S for symmetric objects. The objects with visibility fraction less than 10% are skipped for evaluation following the BOP [15] convention. Results are reported by averaging over the rigid/articulated categories, separately.

	Modality	ADD↑	ADD-S↑	ADD(-S)↑
RBOT [37]	RGB	7.1/0.5	10.3/0.8	7.4/0.5
ICG [34]	RGB-D	**35.6/10.1**	**48.1/15.2**	**38.1/10.1**

Table 5. **Model-Based Pose Tracking.** We report the area under curve (AUC) with respect to ADD, ADD-S and ADD(-S). The higher value, the better performance.

Baselines: For methods relying on object mask during tracking, we directly used the ground-truth mask. Since the big gap of performance between rigid and articulated objects, we separately reported their average results.

We regarded each part of the articulated objects as independent for all tracking methods except for CAPTRA [44] which treat all parts of an object as a whole. We trained CAPTRA for each category using our synthetic PBR data.

Result Analysis: Tables 5 and 6 present the performance metrics of state-of-the-art (SOTA) tracking methods on the PACE dataset with both rigid and articulated results separated by slash. The methods exhibit limited success, with ICG [34] achieving the highest AUC with respect to ADD(-S) at only 38.1% for rigid objects and a mere 10.1% for articulated objects, as shown in Table 5. Table 6 underscores the challenge further, with the best 5°5cm accuracy below 13%, and IoU25 not exceeding 47%, indicating substantial pose estimation errors for over half of the objects.

The convention followed by 6-PACK [39] and BundleTrack [42] disregards objects with IoU$\leq 25\%$ when computing rotational and translational errors (R_{err} and T_{err}). However, in the context of our dataset where methods generally struggle, this approach could mask true performance levels. This discrepancy is exemplified by BundleTrack [42], which reports lower performance in 5°5cm and IoU25 metrics yet shows seemingly better results for R_{err} and T_{err}. Such results indicate that current SOTA methods may require significant improvements to handle the complexity presented by the PACE dataset effectively. Qualitative results can be found in the supplementary.

6. Conclusions and Future Work

This work presented a comprehensive benchmark for 3D object pose estimation and tracking through the introduction of the PACE dataset. Our findings reveal that while there have been significant advancements in pose estimation techniques, there exists a substantial performance gap when these methods are applied to real-world, diverse datasets such as PACE. Particularly, current state-of-the-art methods struggle with articulated objects and exhibit limitations in robustness and scalability. This benchmark serves not only as a testament to the progress achieved but also as a clarion call for the research community to address the complexities of real-world applications.

The results from the PACE dataset underscore a pronounced generalization gap, suggesting that existing models may not sufficiently capture the complexities inherent in

	Training-Free	Modality	5°5cm↑	IoU25↑	R_{err}↓	T_{err}↓
BundleTrack [42]	✓	RGB	6.4/**11.2**	9.1/14.1	**3.2/5.5**	2.6/**0.8**
CAPTRA* [44]	✗	D	**12.9**/4.4	**47.0**/18.5	20.2/46.7	**2.1**/1.5
CAPTRA [44]	✗	D	**12.9**/4.4	45.8/**20.6**	19.2/40.9	2.2/1.5
6-PACK [39]	✗	RGB-D	9.2/3.9	23.1/16.7	17.7/33.6	**2.1**/1.2

Table 6. **Model-Free Pose Tracking**. Both rigid and articulated results are reported. For 5°5cm and IoU25, the higher value means better performance while for R_{err} and T_{err}, the situation is reversed. Note that R_{err} and T_{err} are respect to IoU25 since objects with IoU$\leq 25\%$ are not counted. Here CAPTRA* uses the predicted 3D bounding box size while others take the ground-truth size to calculate IoU.

diverse real-world scenarios. To bridge this gap, future research could explore the potential of larger, more complex models for pose estimation that can encapsulate a wider variety of object features and environmental contexts.

7. Acknowledgements

This work was supported by the National Key Research and Development Project of China (No. 2021ZD0110700), Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), Shanghai Qi Zhi Institute, and SHEITC (2018-RGZN-02046). Yang You is also supported in part by the Outstanding Doctoral Graduates Development Scholarship of Shanghai Jiao Tong University.

References

[1] Adel Ahmadyan, Liangkai Zhang, Artsiom Ablavatski, Jianing Wei, and Matthias Grundmann. Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7822–7831, 2021. 3

[2] Armen Avetisyan, Manuel Dahnert, Angela Dai, Manolis Savva, Angel X Chang, and Matthias Nießner. Scan2cad: Learning cad model alignment in rgb-d scans. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 2614–2623, 2019. 2, 3

[3] Eric Brachmann, Alexander Krull, Frank Michel, Stefan Gumhold, Jamie Shotton, and Carsten Rother. Learning 6d object pose estimation using 3d object coordinates. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II 13*, pages 536–551. Springer, 2014. 2, 3

[4] Kai Chen and Qi Dou. Sgpa: Structure-guided prior adaptation for category-level 6d object pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2773–2782, 2021. 2, 7, 8

[5] Xinke Deng, Arsalan Mousavian, Yu Xiang, Fei Xia, Timothy Bretl, and Dieter Fox. Poserbpf: A rao–blackwellized particle filter for 6-d object pose tracking. *IEEE Transactions on Robotics*, 37(5):1328–1342, 2021. 2

[6] Maximilian Denninger, Dominik Winkelbauer, Martin Sundermeyer, Wout Boerdijk, Markus Knauer, Klaus H. Strobl, Matthias Humt, and Rudolph Triebel. Blenderproc2: A procedural pipeline for photorealistic rendering. *Journal of Open Source Software*, 8(82):4901, 2023. 6

[7] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 224–236, 2018. 4

[8] Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 998–1005. Ieee, 2010. 2, 6, 7

[9] Mathieu Garon and Jean-François Lalonde. Deep 6-dof tracking. *IEEE transactions on visualization and computer graphics*, 23(11):2410–2418, 2017. 3

[10] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 1

[11] Andrew Guo, Bowen Wen, Jianhe Yuan, and OTHERS. Handal: A dataset of real-world manipulable object categories with pose annotations, affordances, and reconstructions. 2023. 2, 3

[12] Rasmus Laurvig Haugaard and Anders Glent Buch. Surfemb: Dense and continuous correspondence distributions for object pose estimation with learnt surface embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6749–6758, 2022. 2, 6, 7, 1

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 1

[14] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Computer Vision–ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part I 11*, pages 548–562. Springer, 2013. 8

[15] Tomáš Hodaň, Martin Sundermeyer, Bertram Drost, Yann Labbé, Eric Brachmann, Frank Michel, Carsten Rother, and Jiří Matas. Bop challenge 2020 on 6d object localization. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 577–594. Springer, 2020. 6, 8

[16] Varun Jampani, Kevis-Kokitsi Maninis, Andreas Engelhardt, Arjun Karpur, Karen Truong, Kyle Sargent, Stefan Popov, André Araujo, Ricardo Martin-Brualla, Kaushal Patel, et al. Navi: Category-agnostic image collections with high-quality 3d shape and pose annotations. *arXiv preprint arXiv:2306.09109*, 2023. 2, 3

[17] Laura F Julià and Pascal Monasse. A critical review of the trifocal tensor estimation. In *Image and Video Technology: 8th Pacific-Rim Symposium, PSIVT 2017, Wuhan, China, November 20-24, 2017, Revised Selected Papers 8*, pages 337–349. Springer, 2018. 4

[18] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 5

[19] Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. Cosypose: Consistent multi-view multi-object 6d pose estimation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVII 16*, pages 574–591. Springer, 2020. 2, 6, 7, 1

[20] Feng Li, Hao Zhang, Huaizhe Xu, Shilong Liu, Lei Zhang, Lionel M Ni, and Heung-Yeung Shum. Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3041–3050, 2023. 1

[21] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022. 1

[22] Xiaolong Li, He Wang, Li Yi, Leonidas Guibas, A Lynn Abbott, and Shuran Song. Category-level articulated object pose estimation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 7

[23] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 683–698, 2018. 2

[24] Haitao Lin, Zichang Liu, Chilam Cheang, Yanwei Fu, Guodong Guo, and Xiangyang Xue. Sar-net: Shape alignment and recovery network for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6707–6717, 2022. 1, 2, 7, 8

[25] Jiehong Lin, Zewei Wei, Zhihao Li, Songcen Xu, Kui Jia, and Yuanqing Li. Dualposenet: Category-level 6d object pose and size estimation using dual pose network with refined learning of pose consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3560–3569, 2021. 7, 8

[26] Yunzhi Lin, Jonathan Tremblay, Stephen Tyree, Patricio A Vela, and Stan Birchfield. Keypoint-based category-level object pose tracking from an rgb sequence with uncertainty estimation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 1258–1264. IEEE, 2022. 3

[27] Liu Liu, Wenqiang Xu, Haoyuan Fu, Sucheng Qian, Qiaojun Yu, Yang Han, and Cewu Lu. Akb-48: A real-world articulated object knowledge base. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14809–14818, 2022. 3

[28] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 1

[29] Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21013–21022, 2022. 2, 3

[30] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 577–582. 2023. 5

[31] Giorgia Pitteri, Michaël Ramamonjisoa, Slobodan Ilic, and Vincent Lepetit. On object symmetries and 6d pose estimation from images. In *2019 International conference on 3D vision (3DV)*, pages 614–622. IEEE, 2019. 1

[32] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10901–10911, 2021. 2, 3

[33] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020. 4

[34] Manuel Stoiber, Martin Sundermeyer, and Rudolph Triebel. Iterative corresponding geometry: Fusing region and depth for highly efficient 3d tracking of textureless objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6855–6865, 2022. 3, 8

[35] Manuel Stoiber, Mariam Elsayed, Anne E Reichert, Florian Steidle, Dongheui Lee, and Rudolph Triebel. Fusing visual appearance and geometry for multi-modality 6dof object tracking. *arXiv preprint arXiv:2302.11458*, 2023. 3

[36] Xingyuan Sun, Jiajun Wu, Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Tianfan Xue, Joshua B Tenenbaum, and William T Freeman. Pix3d: Dataset and methods for single-image 3d shape modeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2974–2983, 2018. 2, 3

[37] Henning Tjaden, Ulrich Schwanecke, Elmar Schömer, and Daniel Cremers. A region-based gauss-newton approach to real-time monocular multiple object tracking. *IEEE transactions on pattern analysis and machine intelligence*, 41(8): 1797–1812, 2018. 2, 8

[38] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. Densefusion: 6d object pose estimation by iterative dense fusion. In *Proceed-*

ings of the IEEE/CVF conference on computer vision and pattern recognition, pages 3343–3352, 2019. 2, 3, 8

[39] Chen Wang, Roberto Martín-Martín, Danfei Xu, Jun Lv, Cewu Lu, Li Fei-Fei, Silvio Savarese, and Yuke Zhu. 6-pack: Category-level 6d pose tracker with anchor-based keypoints. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10059–10066. IEEE, 2020. 3, 7, 8, 9

[40] Gu Wang, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. GDR-Net: Geometry-guided direct regression network for monocular 6d object pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16611–16621, 2021. 2, 6, 7

[41] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2642–2651, 2019. 1, 2, 3, 6, 7, 8

[42] Bowen Wen and Kostas Bekris. Bundletrack: 6d pose tracking for novel objects without instance or category-level 3d models. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8067–8074. IEEE, 2021. 3, 5, 8, 9

[43] Bowen Wen, Chaitanya Mitash, Baozhang Ren, and Kostas E Bekris. se (3)-tracknet: Data-driven 6d pose tracking by calibrating image residuals in synthetic domains. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10367–10373. IEEE, 2020. 3

[44] Yijia Weng, He Wang, Qiang Zhou, Yuzhe Qin, Yueqi Duan, Qingnan Fan, Baoquan Chen, Hao Su, and Leonidas J Guibas. Captra: Category-level pose tracking for rigid and articulated objects from point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13209–13218, 2021. 3, 8, 9, 2

[45] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017. 1, 2, 3, 8

[46] Yanjie Ze and Xiaolong Wang. Category-level 6d object pose estimation in the wild: A semi-supervised learning approach and a new dataset. *Advances in Neural Information Processing Systems*, 35:27469–27483, 2022. 3

[47] Linfang Zheng, Chen Wang, Yinghan Sun, Esha Dasgupta, Hua Chen, Aleš Leonardis, Wei Zhang, and Hyung Jin Chang. Hs-pose: Hybrid scope feature extraction for category-level object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17163–17173, 2023. 1, 2, 7, 8

PACE: Pose Annotations in Cluttered Environments

Supplementary Material

A. Object Detection Benchmark

Though the primary focus of this dataset is the pose estimation, we are interested in how current state-of-the-art detection models perform on our dataset. In practice, the best method can be served as the default detection method when evaluating the pose estimation result.

A.1. Instance-Level Object Detection

Evaluation Metrics: Consistent with established literature, we employ Average Precision (AP), Average Precision at IoU thresholds of 50% (AP_{50}) and 75% (AP_{75}), along with Average Recall (AR) as the metrics for evaluation.

Baseline Methods: The task is categorized into bounding box (BBox) detection and instance segmentation (Mask). We benchmark the performance against three models: Mask R-CNN [13], YOLO-X [10], and MaskDINO [20]. MaskDINO is notable for its claim of providing precise bounding box and instance segmentation results. Models are trained on a provided PBR dataset, treating each instance as an individual "category", culminating in 576 unique categories. Results are averaged across all instances in the test set.

Results and Analysis: Quantitative outcomes are presented in Table 7. We see that YOLO-X is the best in terms of the bounding box detection while MaskDINO is the best in instance segmentation.

	Type	AP↑	AP_{50}↑	AP_{75}↑	AR↑
YOLO-X [10]	BBox	**45.4**	**60.0**	**51.5**	**61.5**
MaskDINO (BBox) [20]	BBox	31.5	42.5	36.1	48.8
Mask R-CNN [13]	Mask	26.1	41.7	30.2	33.6
MaskDINO (Mask) [20]	Mask	**29.5**	**42.0**	**33.2**	**44.1**

Table 7. **Instance-level object detection results.**

A.2. Category-Level Object Detection

Evaluation Metrics: The metrics for category-level object detection are identical to those used for instance-level detection.

Baseline Methods: This task is bifurcated into bounding box (BBox) detection and instance segmentation (Mask). We evaluate the performance of YOLO-X [10], MaskDINO [20], and the zero-shot detector GLIP [21]. GLIP is capable of inferring bounding boxes based on textual descriptions of target categories. For instance segmentation, we compare Mask R-CNN [13], MaskDINO [20], and Grounded SAM [28], which can generate instance masks from text prompts in a zero-shot fashion. Non-zero-shot methods are trained on images from the provided PBR training set.

Results and Analysis: Quantitative findings are detailed in Table 8. YOLO-X again is the winner is detecting bounding boxes and MaskDINO is the winner in instance segmentation. Though zero-shot methods can achieve some results but they are still far behind the supervised methods, partially due to the text ambiguity in describing the objects.

	Type	ZS	AP↑	AP_{50}↑	AP_{75}↑	AR↑
YOLO-X [10]	BBox	✗	**50.9**	**66.4**	**57.9**	**64.3**
MaskDINO (BBox)	BBox	✗	46.8	65.3	53.3	59.7
GLIP [21]	BBox	✓	22.2	30.3	26.1	59.8
Mask R-CNN [13]	Mask	✗	40.9	60.3	**47.8**	51.4
MaskDINO (Mask)	Mask	✗	**42.9**	**65.1**	47.4	**54.6**
Grounded-SAM [28]	Mask	✓	8.2	11.7	9.6	15.0

Table 8. **Category-level object detection results.** This table will reflect the comparative effectiveness of both zero-shot and traditional detection methods in category-level detection tasks.

B. Baseline Adaptations

This section delineates the adaptations made to the baseline methodologies enabling their evaluation on our dataset. Unless otherwise stated, the configurations adhere to the defaults specified in their respective originating papers.

B.1. Handling Object Symmetry

For baselines tasked with rotational regression, we address the ambiguity presented by object symmetry, where multiple rotations correspond to a single input by normalizing these rotations into distinct, unambiguous targets, following the method outlined in [31].

B.2. Instance-Level Pose Estimation Baselines

CosyPose [19] To maintain comparability, we exclude the random background pasting augmentation from CosyPose. Moreover, to accommodate our hardware constraints, we reduce the image resolution by half.

SurfEmb [12] With SurfEmb, images are decoded into a shared continuous embedding space alongside 3D point

embeddings. Due to the extensive variety of objects in our dataset (576 in total), a separate image decoder for each would exceed our GPU's memory capacity. Consequently, we implement a unified decoder across all objects.

B.3. Pose Tracking Baselines

CAPTRA [44] We treat all objects as non-symmetric for training purposes, given that CAPTRA's loss is designed to handle only asymmetry or continuous symmetry along the y-axis. We also limit our scope to articulated objects comprising two parts, aligning with CAPTRA's fixed-part count assumption. To manage the higher image resolution and variable object size in our dataset compared to NOCS, we decrease the *radius* parameter when computing the axis-aligned bounding box *aabb* and downsample the back-projected point clouds prior to ball filtering.

C. Qualitative Comparison on Pose Estimation/Tracking

In this section, we present additional illustrations that further elucidate the performance of various baseline methodologies. Specifically, Figure 10 provides a visualization of the efficacy of instance-level pose estimation methods. Similarly, Figure 11 demonstrates the performance of category-level pose estimation techniques. Additionally, Figure 12 and Figure 13 offer visual representations of the performance metrics for model-based and model-free tracking methods, respectively. These visualizations serve to complement the quantitative analyses provided earlier, offering a more comprehensive understanding of each method's effectiveness.

D. Annotation Software and Pipeline Details

Our annotation interface features a primary display that simultaneously presents views from a triple-camera setup. Adjacently, a sidebar is integrated, showcasing the current annotations within the scene, along with a catalog of potential 3D models from our database. Users can initiate the pose estimation by aligning corresponding points on the 3D models with their 2D image counterparts, utilizing a RANSAC-PnP algorithm. Subsequent refinements to the pose are facilitated through incremental rotations and translations, which can be executed via keys.

For enhanced user experience and efficiency, we have designed a tripartite panel system. This allows users to seamlessly toggle between 2D annotation, 3D annotation, and 2D segmentation workflows. As illustrated in Figure 14, this configuration enables real-time previews of annotations in both 2D and 3D, ensuring an intuitive and dynamic annotation process.

E. Snapshots for All the Objects

Figure 15 gives the model snapshots for all our collected objects.

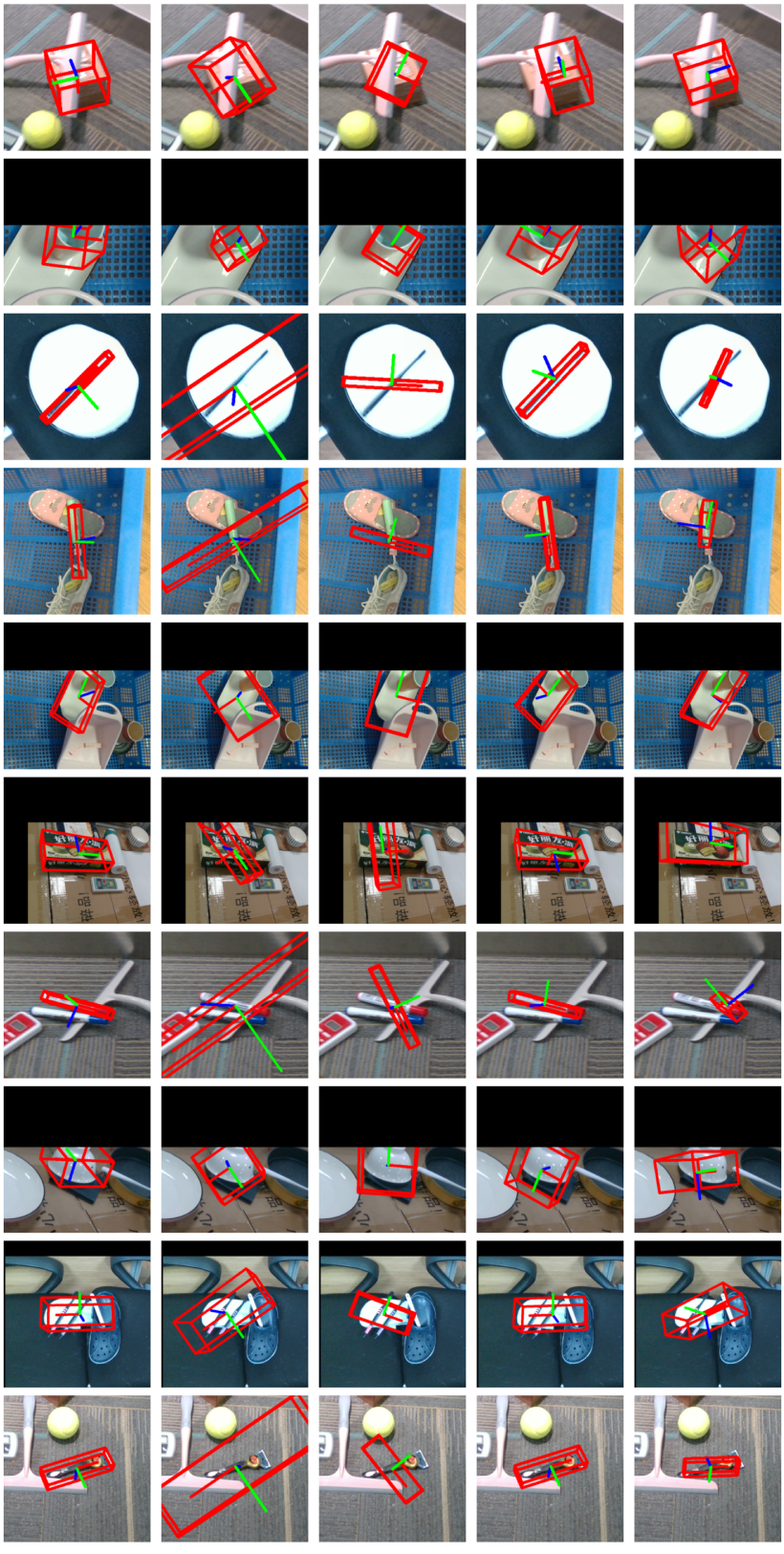


Figure 10. Qualitative comparisons of instance-level pose estimation methods highlighting the robust performance of the PPF method across various instances.

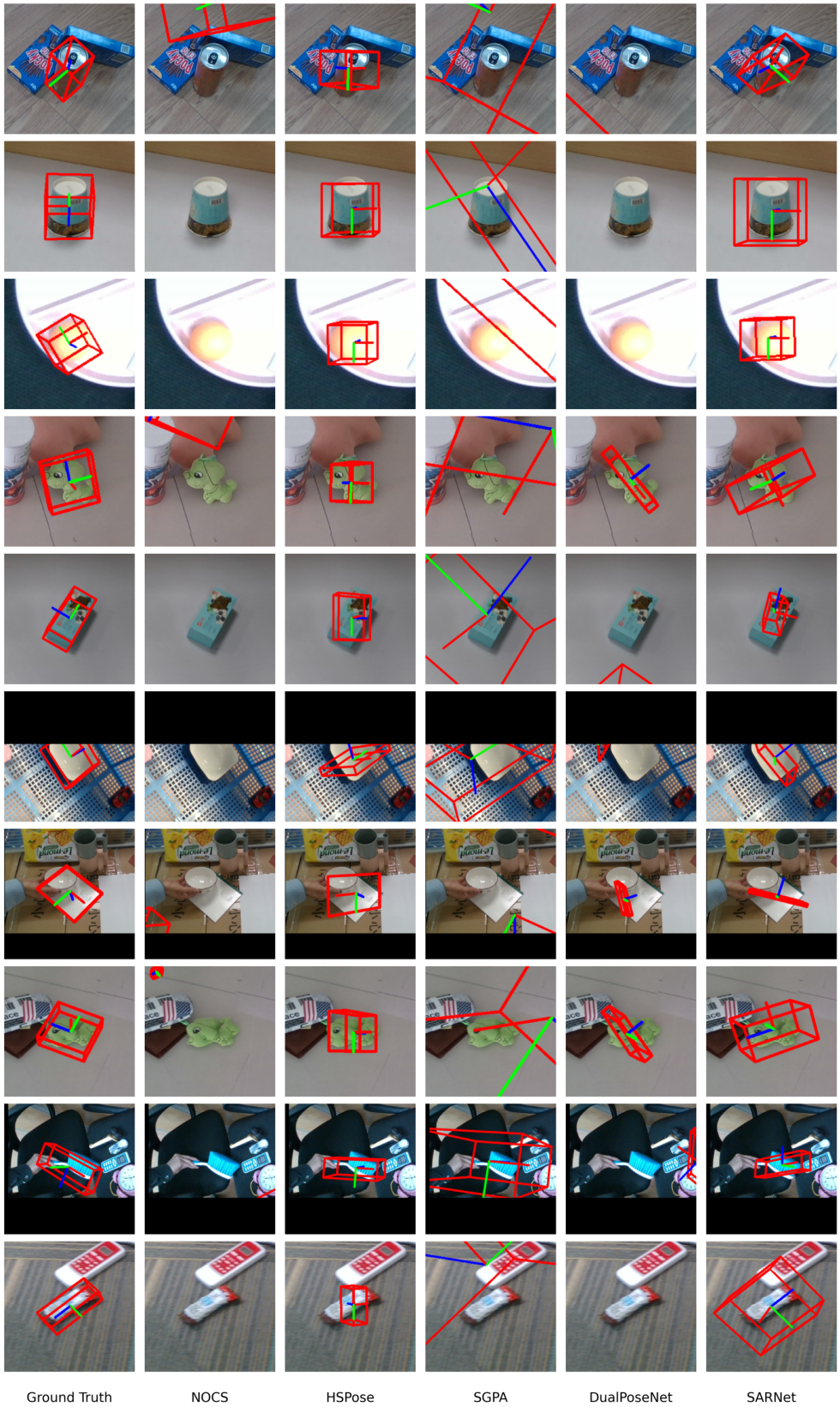


Figure 11. Qualitative comparisons of category-level pose estimation methods.

Figure 12. Qualitative comparisons of model-based tracking methods.

Figure 13. Qualitative comparisons of model-free tracking methods.

Figure 14. From top to bottom: 2D pose annotation panel, 3D pose annotation panel with integrated point clouds from all three views, 2D segmentation annotation panel.

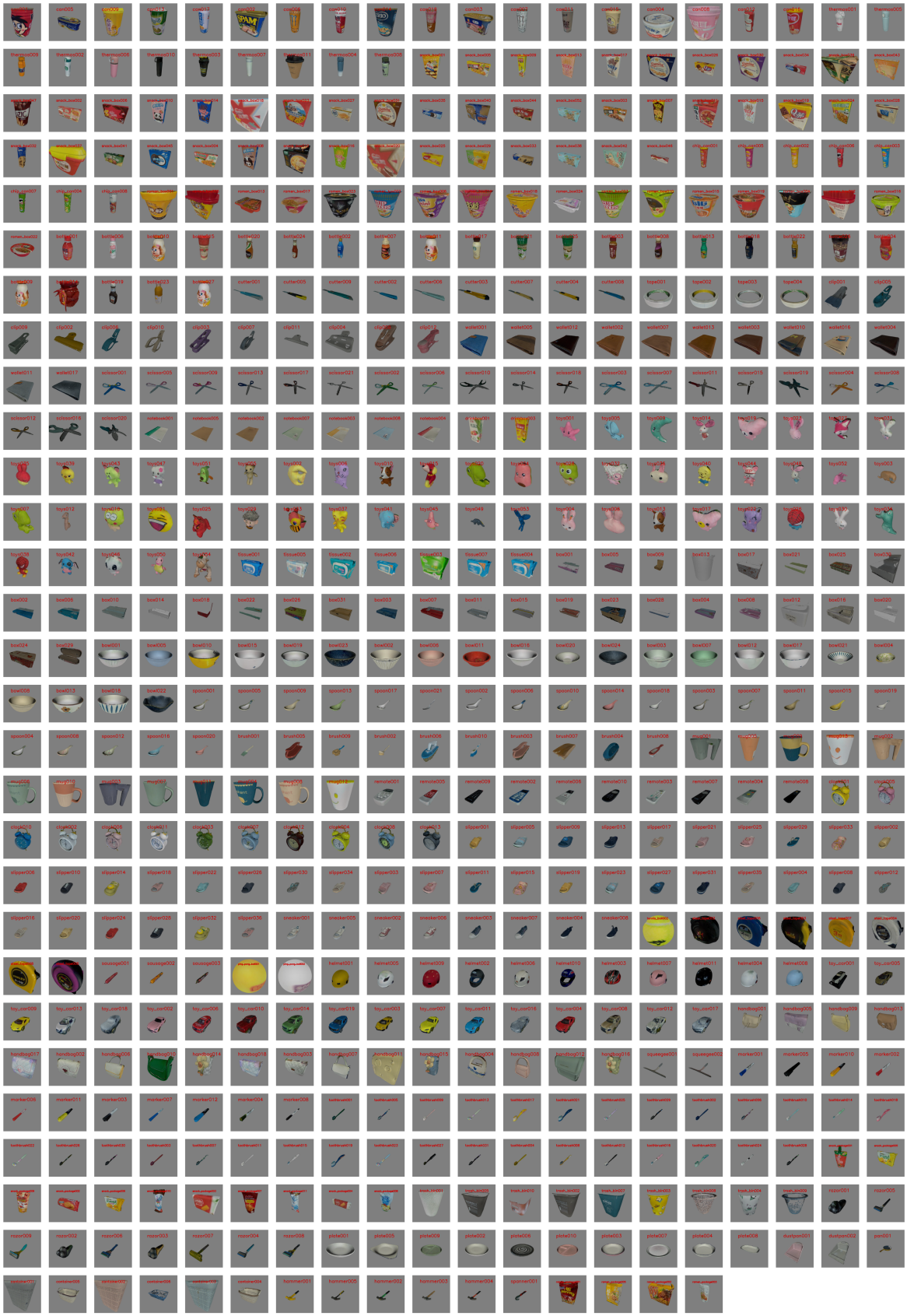


Figure 15. Snapshots from all the collected objects.