

# StereOBJ-1M: Large-scale Stereo Image Dataset for 6D Object Pose Estimation

Xingyu Liu      Shun Iwase      Kris M. Kitani  
Carnegie Mellon University

## Abstract

We present a large-scale stereo RGB image object pose estimation dataset named the **StereOBJ-1M** dataset. The dataset is designed to address challenging cases such as object transparency, translucency, and specular reflection, in addition to the common challenges of occlusion, symmetry, and variations in illumination and environments. In order to collect data of sufficient scale for modern deep learning models, we propose a novel method for efficiently annotating pose data in a multi-view fashion that allows data capturing in complex and flexible environments. Fully annotated with 6D object poses, our dataset contains over 396K frames and over 1.5M annotations of 18 objects recorded in 183 scenes constructed in 11 different environments. The 18 objects include 8 symmetric objects, 7 transparent objects, and 8 reflective objects. We benchmark two state-of-the-art pose estimation frameworks on StereOBJ-1M as baselines for future work. We also propose a novel object-level pose optimization method for computing 6D pose from keypoint predictions in multiple images.

## 1. Introduction

Effectively leveraging 3D cues from visual data to infer the pose of an object is crucial for applications such as augmented reality (AR) and robotic manipulation. Compared to objects with opaque and Lambertian surfaces, estimating the pose of transparent and reflective objects is especially challenging. To leverage depth information from sensors, previous approaches have explored deep models that take RGB-D maps as input [36, 37, 8, 29, 34]. Unfortunately, as the experiments in [18, 40, 31, 13] have shown, existing commercial depth-sensing methods, such as time-of-flight (ToF) or projected light sensors, failed to capture the depths of transparent or reflective surfaces. As a result, monocular RGB-D maps cannot serve as a reliable input for object pose estimation models in these challenging scenarios. Based on this observation, we focus on using stereo RGB images as our input modality, allowing for object pose estimation on a wider range of objects, including transparent or highly reflective objects.



Figure 1: **StereOBJ-1M dataset.** Upper: CAD models of the objects in the dataset. Lower: four data stereo image pair samples with bounding box annotations.

A major challenge in modern object pose estimation is that of acquiring a large-scale training dataset. To increase data size for training large-scale neural networks, previous works have explored leveraging synthetically rendered [35, 28, 10] or augmented images [37] with 3D mesh models. However, photorealistic rendering is still challenging with only basic graphics rendering tools and limited expertise. Synthetic image datasets that are currently available typically introduce a very large domain gap. This is especially true for transparent and reflective objects where variations in illumination and background scenes are crucial but difficult to model.

To address the challenge of costly pose data acquisition, and to enable further training and evaluation of modern object pose estimation models, we introduce **a novel method for capturing and labeling a large-scale dataset** with high efficiency and quality. Our method is based on multi-view geometry to accurately localize fiducial markers, cameras, and object keypoints in the scene. We use a hand-held stereo camera to record video data. With the help of two other static cameras mounted to tripods, the positions of a

dataset	data type	stereo	depth	occlusion	transparent objects	reflective objects	# of frames	# of outdoor environments	# of scenes	# of objects	# of annotations
FAT [35]	synthetic	✓	✓	✓	✗	✗	60,000	0	3,075	21	205,359
CAMERA [37]	mixed reality	✗	✓	✓	✗	✗	300,000	0	30	42	4,350,656
YCB [1]	real	✗	✓	✓	✗	✗	133,936	0	92	21	613,917
LINEMOD [9]	real	✗	✓	✓	✗	✗	18,000	0	15	15	15,784
GraspNet-1Billion [4]	real	✗	✓	✓	✗	✗	97,280	0	190	88	970,000
T-LESS [10]	real	✗	✓	✓	✗	✗	47,762	0	-	30	47,762
KPAM [22]	real	✗	✓	✓	✗	✓	100,000	0	-	91	-
LabelFusion [23]	real	✗	✓	✓	✗	✓	352,000	0	138	12	1,000,000
REAL275 [37]	real	✗	✓	✗	✗	✗	7,072	0	13	42	35,356
TOD [18]	real	✓	✓	✗	✓	✗	64,000	0	10	20	64,000
<b>StereOBJ-1M (Ours)</b>	real	✓	✗	✓	✓	✓	<b>396,509</b>	<b>3</b>	183	18	<b>1,517,835</b>

Table 1: Dataset Comparisons. Our StereOBJ-1M dataset is the only large-scale 6DoF object pose dataset that provides stereo RGB as input modality, includes transparent and reflective objects and is captured in both indoor and outdoor environments. In terms of capacity, our dataset is also the largest real-image dataset in size and the dataset with the most scene diversity.

set of fiducial markers can be calculated on the fly, from which the pose of every recorded frame can be automatically computed. By annotating 2D object keypoints in just a few frames selected in a long recorded video, the 3D locations of the keypoints can be computed by triangulation. The 6D poses of objects can then be calculated by aligning 3D CAD models to the keypoints before being propagated to all other frames.

Using the procedure outlined above, we generate **StereOBJ-1M dataset**, the first pose dataset with stereo RGB as input modality with over 100K frames. It is also the largest 6D object pose dataset in history: it consists of 396,509 high-resolution stereo frames and over 1.5 million 6D pose annotations of 18 objects recorded in 183 indoor and outdoor scenes. The capacity of StereOBJ-1M is sufficient for training large-scale neural networks without additional synthetic images. The average labeling error of StereOBJ-1M is 2.3mm which is the best annotation precision among all public object pose datasets.

We implement two state-of-the-art methods [28, 18] as the baseline comparisons for 6D pose estimation using stereo on the StereOBJ-1M dataset. To handle 2D-3D correspondences predictions in two or more images, we propose a novel object-level 6D pose optimization approach named **Object Triangulation**. Contrary to classic triangulation that optimizes the 3D location of a *point*, we directly optimize the 6D pose of an *object* from 2D keypoint locations in multiple images. Experiment results show that Object Triangulation consistently improves pose estimation over monocular input while classic triangulation can yield worse results. With Object Triangulation, the stereo variants of both baseline methods significantly outperform their monocular counterparts on StereOBJ-1M, by at least 25% in ADD(-S) AUC and 14% in ADD(-S) accuracy, which highlights the importance of stereo modality in object pose estimation. We expect that StereOBJ-1M will serve as a common benchmark dataset for stereo RGB-based object pose estimation.

## 2. Related Work

**Pose Annotation Methods.** The first category of pose data annotation methods relies on capturing RGB-D images, reconstructing 3D point clouds, and labeling pose by constructing a 3D mesh [22], or fitting 3D object mesh models to 3D point clouds [23, 1, 9]. However, this type of method cannot reliably deal with transparent objects where depth sensing is usually not possible. The second category of pose data annotation methods adopts keypoint as representation and leverages multi-view geometry for triangulation [12, 18]. Our novel data annotation method is keypoint and multi-view based. Different from previous methods, we record the scenes using a stereo RGB camera whose poses are computed on the fly based on fiducial markers whose locations are also computed on the fly.

**Stereo Methods.** Studying correspondence, depth, and other downstream tasks from two or multiple RGB images has been a long-standing topic in computer vision and robotics. Previous works have explored stereo-based methods for 3D object detection [17], disparity estimation [41], point-based 3D reconstruction [3] and keypoint detection [18]. Recently, multi-view based methods have been proposed for object pose estimation [16, 15]. Our 6D object pose dataset provides binocular stereo RGB images as the input modality, allowing stereo-based deep methods to be trained on object pose data. In addition, the annotation process of our dataset utilizes multi-view geometry.

**Keypoint in Pose Representation.** Keypoints is a popular intermediate representation for object or human pose. Previous work has explored deep learning methods for localizing keypoints of an object [33, 27, 18, 8, 14] or a human [32, 25, 2] from a RGB image. Several public object pose estimation datasets are also constructed by using keypoints to simplify pose annotation [23, 18]. Our data annotation pipeline also uses keypoints as a bridge to the 6D pose, where the 3D positions of the keypoints are calculated through multi-view triangulation.

**Related Dataset.** Most existing pose datasets provide

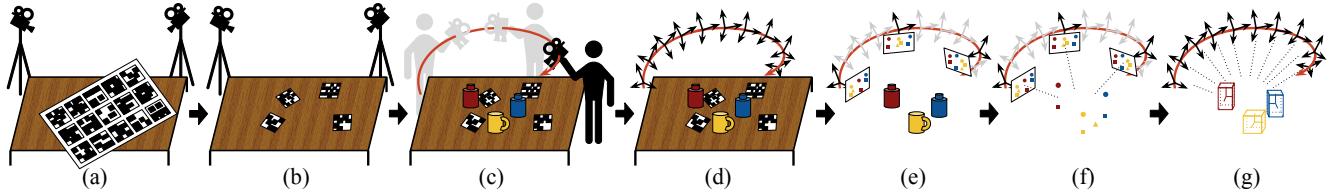


Figure 2: Our data capturing and labeling pipeline for one captured video: (a) use PnP to calculate fixed cameras’ global poses from fiducial marker board ; (b) triangulation of fiducial marker locations; (c) scanning the scene with stereo camera; (d) use PnP to calculate moving camera’s global poses from fiducial markers; (e) annotate keypoints on sampled images; (f) triangulation of object 3D keypoints; (g) 6D pose fitting from 3D keypoints and propagation to all images.

RGB-D as input modality [1, 7, 9, 4, 10, 22, 23, 37]. Since directly labeling 3D object pose in real RGB images is costly and inaccurate, most existing datasets rely on capturing RGB-D images and fitting 3D mesh models to 3D point clouds as their labeling method [23, 1, 37, 10, 9]. TOD [18] is the first object pose dataset with binocular stereo RGB as the input modality, and it uses a data labeling method based on multi-view geometry. However, TOD records in a studio environment and does not include occluded objects. Our dataset provides binocular stereo RGB as input modality and records objects with occlusion in 11 different real environments. A more comprehensive comparison of datasets is illustrated in Table 1.

### 3. Data Capturing & Labeling Pipeline

One of the major challenges in the pose estimation of 3D objects is the acquisition and annotation of large-scale and high-quality real object pose data. Limitations of previous efforts are in the following three aspects.

**Sensor Modality.** Most existing datasets such as [1, 7, 9, 10, 22, 23, 37] only provide monocular RGBD from commercial depth sensors as 3D cue. These datasets and the associated labeling methods did not and could not handle transparent or reflective objects on which depth sensing is not reliable. Moreover, different technologies of depth sensing, e.g. infrared and LiDAR, may return different depths for the same objects and scenes. Thus a model trained on one RGBD-based pose dataset may not be able to generalize to another with a different depth-sensing technology.

**Data Annotation.** Existing data annotation methods usually require annotators to manually align object CAD model to 3D sensor signals, e.g. reconstructed 3D point clouds from depth maps, which are expensive and inaccurate. Limited by the cost of data annotation, the sizes of public real-world datasets such as [7, 9, 10, 37] are in the order of 10K or fewer images, which are insufficient for training large-scale deep neural net models. An alternative solution is to leverage synthetically rendered or augmented images. However, the problem of the domain gap is still yet to be solved and is especially challenging for transparent and reflective objects.

**Scene Environment.** Datasets such as [18, 4, 10, 9] were captured in a small number ( $< 3$ ) of special indoor environments or studios and lack the diversity of real scenes. It is hard for models trained on such data to generalize to unseen environments, especially for transparent and reflective objects where background scenes and illumination are crucial.

#### 3.1. Data Capturing and Labeling

To address the above problems, we propose a novel method for efficiently capturing and labeling 3D object pose data. We opt to use stereo RGB modality to provide 3D cues for the data. For labeling, our philosophy is to abandon depth sensing and utilize multi-view geometry for high-precision 3D localization of object keypoints for pose fitting. An overview of our pipeline is illustrated in Figure 2. It consists of the following seven steps which respectively correspond to Figure 2(a)-(g).

**1. Pose Calculation for Static Cameras.** We set up two static cameras that record the scene simultaneously. The cameras are held by two tripods. To obtain the camera poses in the world coordinate, we place a large customized plastic board printed with an array of fiducial markers into the scene such that most of the fiducial markers are visible in both static cameras. The accurate 3D positions of the fiducial markers on the board are measured by a vernier caliper and act as the world coordinate in the rest of the pipeline. The poses of the two static cameras in world coordinates  $[\mathbf{R}_1^S, \mathbf{T}_1^S] \in \mathbb{R}^{3 \times 4}$  and  $[\mathbf{R}_2^S, \mathbf{T}_2^S] \in \mathbb{R}^{3 \times 4}$  are calculated with Perspective-n-Point (PnP) algorithm [5].

**2. Triangulation of Fiducial Markers.** We remove the plastic fiducial marker array board and place several other small fiducial markers into the scene such that they are visible in both static cameras. The dimensions of the small fiducial marker boards are also accurately measured by a vernier caliper. From  $[\mathbf{R}_1^S, \mathbf{T}_1^S]$  and  $[\mathbf{R}_2^S, \mathbf{T}_2^S]$ , we use triangulation to locate the 3D locations of the corners of the small fiducial markers in world coordinates  $\{\mathbf{x}_i^F \in \mathbb{R}^{4 \times 3}\}$ . During this step, the two static cameras continue to record video and their poses remain unchanged.

**3. Scene Construction and Scanning.** To construct the scene, we first place a few randomly selected objects from

dataset	RGBD datasets	TOD [18]	<b>StereOBJ-1M</b>
3D labeling	depth map	multi-view	
labeling error	$\geq 1.7\text{cm}$ <sup>1</sup>	0.34cm	<b>0.23cm</b>

Table 2: Labeling error measured in 3D RMSE.

our dataset and mingle them with the small fiducial markers. Other occluding objects can also be included if necessary. Note that during this step, the positions of the small fiducial markers must remain unchanged while the static cameras can be removed. Then a human data collector holds a stereo RGB camera, slowly moves it around the scene, scans the objects from different viewpoints, and record a stereo video. The scanning paths are selected aiming to cover as many viewpoints as possible.

**4. Pose Calculation for moving camera.** Given the recorded stereo RGB video of length  $L$ , we use PnP algorithm again to calculate the poses of moving stereo camera in world coordinates  $[\mathbf{R}_j^M, \mathbf{T}_j^M] \in \mathbb{R}^{3 \times 4}$  for every frame  $j \in \{1, 2, \dots, L\}$  of the video, using the small fiducial marker locations  $\{\mathbf{x}_i^F\}$ . To reduce the error of PnP, in practice only the frames with at least two small fiducial markers or eight corners detected are selected as valid frames.

**5. Keypoint Annotation.** From all valid frames, we select a few to annotate the 2D locations of projected object keypoints on the images. The frames are selected using farthest point sampling (FPS) such that their camera translations  $\mathbf{T}_j^M$  are as far away from each other as possible. The keypoints of an object are defined by experts and are easy to be spotted and accurately located, e.g. corners. Note that it is possible that only a subset of the total keypoints are annotated in one particular frame.

**6. Keypoint Triangulation.** For each keypoint of an object, we retrieve all frames in which the keypoint is annotated. Using the moving camera pose  $[\mathbf{R}_j^M, \mathbf{T}_j^M]$  and the 2D annotations, the 3D location of the keypoints in the world coordinate can be calculated by multi-view triangulation.

**7. Pose Fitting.** To obtain the 6D poses of the objects in the world coordinate, we solve an Orthogonal Procrustes problem [6] to fit the object CAD models to the annotated 3D keypoints. Finally, the object poses are propagated to all valid frames via an inverse transform of the camera pose  $[\mathbf{R}_j^M, \mathbf{T}_j^M]$ .

### 3.2. Labeling Error Analysis

An intriguing question that needs to be answered is: how accurate is our labeling method? We assume the error of the dimensions of the large fiducial marker array board and the small fiducial markers are negligible since they are both accurately measured by vernier caliper. Then the labeling

<sup>1</sup>One of the most recent and advanced commercial depth sensors, Microsoft Azure Kinect, has random depth sensing error of 17mm in standard deviation: <https://docs.microsoft.com/en-us/azure/kinect-dk/hardware-specification>

error can come from two steps: automatic detection of small fiducial marker boards and the annotation of keypoint 2D locations, which contributes to the error in two nonlinear optimizations respectively: camera pose estimation and 3D point estimation from multiple views.

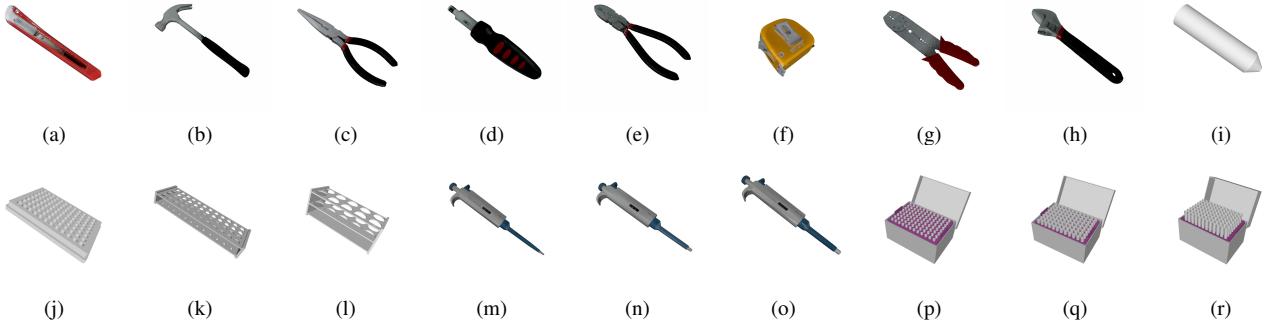
We use Monte Carlo simulation to quantify the pose annotation error with a similar procedure as in [18]. Specifically, we dither the keypoint 2D projections according to the keypoint re-projection RMSE statistics and estimate the 3D keypoint error as an approximate of the labeling error. We report keypoint label error of 2.3mm RMSE as illustrated in Table 2. The reasons for label error improvement over [18] are two folds. First, our stereo camera has a higher resolution than [18] and allows more accurate labeling in 2D. Second, our object scanning paths are determined by human data collectors on the fly instead of being hard-coded and performed by robot [18], thus are more flexible and can adapt to specific scenes to cover more viewpoints and provide wider baselines for triangulation.

### 3.3. Comparison to Previous Labeling Methods

We point out that the idea of using multi-view and keypoints for pose labeling can also be found in human pose estimation scenarios such as the Panoptic Studio dataset [12]. Unlike [12] which relies on 480 fixed cameras mounted in a specially constructed studio for triangulation, our data acquisition method is affordable and portable — it only requires three cameras and two tripods, and can therefore be deployed in diverse indoor and outdoor environments. On the contrary, to construct datasets such as [4, 18], a studio equipped with multiple sensors or robot assists has to be specially constructed. In addition to the logistic cost, such settings are not flexible enough for environments in the wild and therefore suffer from the lack of diversity of data.

TOD [18] is the first object pose estimation dataset that provides stereo RGB modality. Our data capturing pipeline is different from [18] in terms of moving camera pose calculation. Datasets such as [18, 9, 10] rely on customized board printed with fiducial markers, and objects are placed near the center of the board. Thus only the simplest planar terrain can be used with the objects and lacks diversity. Instead, in our data pipeline, we distribute small fiducial markers into the scene and calculate their locations on the fly with the help of two static cameras. This allows the objects to be placed in more flexible and complex background terrains.

Our data pipeline has a much higher data efficiency than TOD [18]. With the proposed data pipeline, in each constructed scene, we can capture and annotate more than 2,000 valid frames with a single scan. As a comparison, [18] only captures 80 frames per scene with the help of a robot arm. An explanation is that in [18], the predefined automatic scanning path of the robot is limited by its opera-



**Figure 3: 3D CAD models of the 18 Objects in our StereOBJ-1M dataset.** (a) blade\_razor; (b) hammer; (c) needle\_nose\_pliers (d) screwdriver; (e) side\_cutters; (f) tape\_measure; (g) wire\_stripper; (h) wrench; (i) centrifuge\_tube; (j) microplate; (k) tube\_rack\_2; (l) tube\_rack\_50; (m) pipette\_0.5-10; (n) pipette\_10-100; (o) pipette\_100-1000; (p) sterile\_rack\_10; (q) sterile\_rack\_200; (r) sterile\_rack\_1000. During scanning, reflective parts of the objects are covered with white scanning spray. Among the objects, (c)(d)(e)(g)(j)(k)(l) have discrete 2-fold rotational symmetry; (i) has continuous rotational symmetry.

tional space. In our data pipeline, the scanning is performed by humans and can adapt to different scenes, which results in (1) more valid frames per video; (2) larger coverage of viewpoints; and (3) wider baseline during triangulation and therefore higher precision.

## 4. StereOBJ-1M Dataset

With the proposed method, we construct StereOBJ-1M, a large-scale dataset, and benchmark for 3D object pose estimation from stereo RGB images. In this section, we provide technical details to our StereOBJ-1M dataset in terms of object 3D models and data sample illustration.

### 4.1. Objects in Dataset

There are 18 objects included in our dataset. Among them, 10 objects are plastic tools used in biochemical labs and 8 objects are metal mechanics tools, which together include both transparent and reflective instances. We provide 3D CAD models of the 18 objects as illustrated in Figure 3. The CAD models are obtained using a high-precision EinScan Pro 2X Plus scanner [21] which has a scan accuracy of 0.04mm. During scanning, reflective metallic parts of the objects are covered with white scanning spray. Among the 18 objects, there are 8 objects with discrete 2-fold rotational symmetry and one with continuous rotational symmetry. Among the 18 objects, microplate, tube\_rack\_2ml and tube\_rack\_50ml are transparent; centrifuge\_tube, sterile\_rack\_10ml, sterile\_rack\_200ml and sterile\_rack\_1000ml are translucent.

The set of the objects used in our dataset has a special feature: it includes visually similar but different object instances. For example, as illustrated in Figure 3, the three

pipettes are almost identical in their geometric features. In our dataset, we include image sequences where two or more similar but different object instances are present in the same scene. Thus it poses a new research question for the computer vision community: *how to detect and estimate the poses of visually very similar but different objects?* We expect that this question can be studied with our dataset.

### 4.2. Data Collection and Annotations

We collected the data in 8 real-life indoor environments, including desktop, washbasin, wooden floor etc. In addition to the indoor environments, we adopt 3 *outdoor* environments to enrich the diversity in background scenes. In each environment, we shuffle the objects and occlusion clutters several times to construct multiple scenes. In total, we constructed 183 scenes. A stereo video was recorded in every constructed scene. The lengths of the video range from 2 to 7 minutes. When sampled at 15 frames/sec, the recorded videos yield 396,509 stereo frames in total. On average, there are more than 2,100 stereo frames in every scene. Our dataset consists of 183 videos and contains over 1.5 million object pose annotations. The number of annotations of each object in each environment is illustrated in Figure 4.

Viewpoint coverage of each object is illustrated in Figure 5. For objects such as microplate and sterile tip racks, there is only one possible side when putting on a desktop, so at most 50% of viewpoint coverage. Annotations in our dataset are 6D poses for every object in the scene, from which object instance segmentation masks, 2D and 3D bounding boxes, and normalized coordinate maps [37] can be inferred. We visualize some data samples from our dataset in Figure 6. As illustrated in the figure, the annotations of our dataset have high quality.

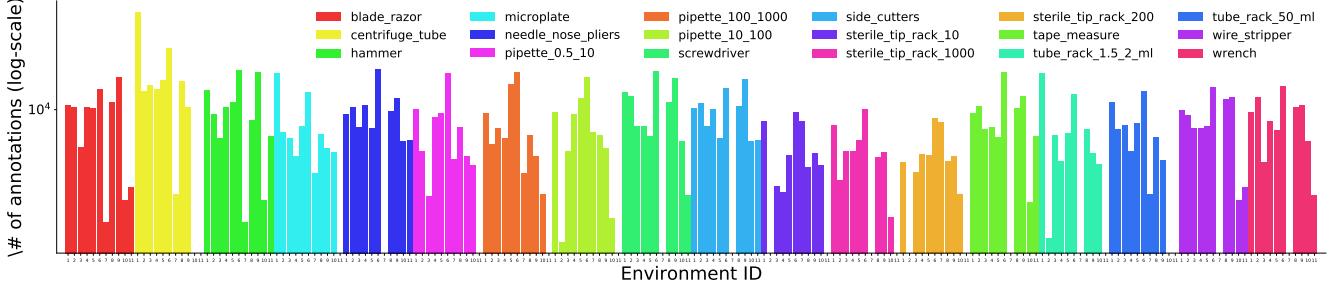


Figure 4: The total number of annotations of each object in each environment. Environment IDs range from 1 to 11.

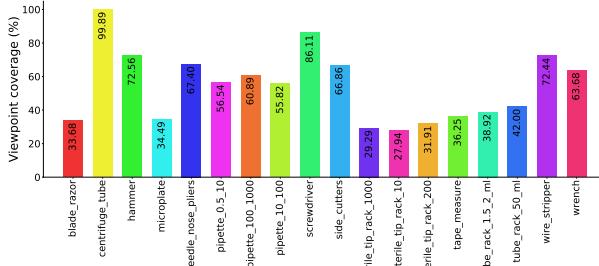


Figure 5: Overall viewpoint coverage percentage of all objects in StereOBJ-1M.

### 4.3. Benchmark and Evaluation

**Train/Validation/Test Split.** The image sequences are divided into train/validation and test sets such that scenes presented in the training set are held out in the validation and test set. The test set contains 32 image sequences that are selected to cover most environments and ensure every object is tested in at least 4,000 images across at least 3 different scenes. In the baseline experiments in Section 5, we did not render additional synthetic data except basic geometric and photometric augmentation, because the capacity of StereOBJ-1M training set is sufficient to train large deep models. However, users of the dataset can still opt to render additional data using the 3D mesh models we provide.

Among the objects, `centrifuge_tube` is the only category with multiple instances recorded in a scene and is used in the multi-object pose detection task. The rest of 17 objects are used in single-object pose estimation task which is the main focus of this paper. Results for pose detection of `centrifuge_tube` are provided in supplementary.

**Evaluation Metrics** We use the popular ADD [9] and ADD-S [39] in our evaluation for 6D pose. When computing ADD distance, we transform the model point set by the predicted and the ground truth poses respectively, and compute the mean 3D Euclidean distance between the two point sets. Given an object with 3D model point set of  $\mathcal{M} = \{\mathbf{x}_i \in \mathbb{R}^3\}$ , the ADD distance is calculated as:

$$\text{ADD} = \frac{1}{|\mathcal{M}|} \sum_{\mathbf{x} \in \mathcal{M}} \|(\mathbf{Rx} + \mathbf{T}) - (\mathbf{R}^* \mathbf{x} + \mathbf{T}^*)\|_2 \quad (1)$$

where  $[\mathbf{R}^* | \mathbf{T}^*]$  and  $[\mathbf{R} | \mathbf{T}]$  are the ground truth and estimated 6D poses. For symmetric objects, ADD-S [39] is used instead. When computing ADD-S distance, the 3D distances are calculated as the average of each point's closest distance to the other point set:

$$\text{ADD-S} = \frac{1}{|\mathcal{M}|} \sum_{\mathbf{x}_1 \in \mathcal{M}} \min_{\mathbf{x}_2 \in \mathcal{M}} \|(\mathbf{Rx}_1 + \mathbf{T}) - (\mathbf{R}^* \mathbf{x}_2 + \mathbf{T}^*)\|_2 \quad (2)$$

We use the following two evaluation metrics. (1) **ADD-S**: ADD(-S) accuracy measures the proportion of correct pose predictions. A pose prediction is considered correct if the ADD(-S) distance is less than the threshold of 10% of the model's diameter. (2) **ADD(-S) AUC**: the area under ADD(-S) accuracy-threshold curve where the maximum threshold is set to 10cm.

## 5. Experiments

### 5.1. Baseline Methods

We implement and evaluate two methods on StereOBJ-1M dataset as baselines for future experiments. Specifically, we implement PVNet [28] and KeyPose [18], two classic keypoint-based 6D pose estimation frameworks that have achieved state-of-the-art performance on various datasets.

**PVNet** [28] is a single-RGB keypoint-based method. It represents keypoints using a 2D direction field and estimates the 2D locations of the keypoints by RANSAC-based voting scheme [5]. The 6D poses are determined by solving a Perspective-n-Point (PnP) problem [5].

**KeyPose** [18] is a stereo-RGB keypoint-based method. Different from PVNet, it localizes object keypoint by predicting heatmaps in both stereo images. The 6D object poses are calculated by keypoint triangulation from two-view stereo and Orthogonal Procrustes pose fitting.

### 5.2. Monocular Image Experiments

We conduct monocular image experiments where only the left images are used as input to predict the 6D pose. The stereo method KeyPose [18] is adapted to its monocular variant where only keypoints in the left stereo images

input modality	monocular RGB		binocular stereo RGB			
pose optimization	PnP [5]		classic triangulation		object triangulation	
metrics	ADD(-S) AUC-10cm	ADD(-S) accuracy-0.1d	ADD(-S) AUC-10cm	ADD(-S) accuracy-0.1d	ADD(-S) AUC-10cm	ADD(-S) accuracy-0.1d
blade_razor	20.40	3.64	40.01	0.02	<b>54.52</b>	<b>12.19</b>
hammer	9.96	3.68	18.96	2.08	<b>37.84</b>	<b>17.62</b>
microplate	38.69	24.66	58.35	22.40	<b>58.80</b>	<b>38.25</b>
needle_nose_pliers	38.25	23.06	63.52	11.87	<b>74.17</b>	<b>51.55</b>
pipette_0.5_10	20.84	14.35	18.10	2.26	<b>34.40</b>	<b>20.95</b>
pipette_100_1000	12.41	1.62	15.12	0.35	<b>22.38</b>	<b>1.74</b>
pipette_10_100	22.13	11.44	24.45	1.03	<b>45.02</b>	<b>24.10</b>
screwdriver	31.74	21.24	64.31	21.08	<b>71.37</b>	<b>46.95</b>
side_cutters	17.93	6.09	60.71	9.11	<b>68.84</b>	<b>38.10</b>
sterile_tip_rack_1000	74.92	67.63	39.24	11.32	<b>77.29</b>	<b>70.74</b>
sterile_tip_rack_10	68.56	60.35	37.17	2.03	<b>73.96</b>	<b>62.55</b>
sterile_tip_rack_200	71.66	63.43	39.50	2.03	<b>75.45</b>	<b>64.85</b>
tape_measure	18.47	1.38	57.42	0.00	<b>68.85</b>	<b>14.67</b>
tube_rack_1.5_2_ml	28.15	15.06	<b>57.72</b>	<b>34.68</b>	43.67	31.82
tube_rack_50_ml	63.24	59.80	57.51	32.61	<b>73.26</b>	<b>69.31</b>
wire_stripper	30.92	21.60	64.75	26.47	<b>81.40</b>	<b>70.98</b>
wrench	8.11	0.82	33.81	0.02	<b>42.40</b>	<b>7.69</b>
average	33.90	23.52	44.16	10.55	<b>59.04</b>	<b>37.89</b>

Table 3: The results of **KeyPose** [18] on single-object pose estimation in terms of **ADD(-S) AUC** and **ADD(-S) accuracy** on StereOBJ-1M dataset. The input modality include monocular and binocular stereo RGB images.

input modality	monocular RGB		binocular stereo RGB			
pose optimization	PnP [5]		classic triangulation		object triangulation	
metrics	ADD(-S) AUC-10cm	ADD(-S) accuracy-0.1d	ADD(-S) AUC-10cm	ADD(-S) accuracy-0.1d	ADD(-S) AUC-10cm	ADD(-S) accuracy-0.1d
blade_razor	24.50	10.88	41.70	0.02	<b>75.82</b>	<b>47.09</b>
hammer	12.10	3.36	17.22	2.26	<b>38.95</b>	<b>21.34</b>
microplate	16.92	6.79	<b>44.56</b>	9.10	43.35	<b>20.91</b>
needle_nose_pliers	8.98	4.26	59.48	9.12	<b>74.60</b>	<b>52.41</b>
pipette_0.5_10	7.23	2.49	20.35	2.54	<b>39.58</b>	<b>18.73</b>
pipette_100_1000	2.31	0.00	11.81	0.19	<b>25.26</b>	<b>0.81</b>
pipette_10_100	17.30	4.58	20.13	0.80	<b>48.89</b>	<b>24.51</b>
screwdriver	42.98	28.80	57.54	16.66	<b>76.27</b>	<b>56.38</b>
side_cutters	51.01	29.13	62.24	12.00	<b>83.78</b>	<b>68.69</b>
sterile_tip_rack_1000	64.44	52.95	20.83	3.09	<b>71.55</b>	<b>61.72</b>
sterile_tip_rack_10	63.16	<b>51.58</b>	19.48	0.52	<b>66.50</b>	50.55
sterile_tip_rack_200	62.92	47.18	22.33	0.23	<b>73.28</b>	<b>59.16</b>
tape_measure	51.64	7.33	56.00	0.00	<b>79.49</b>	<b>29.59</b>
tube_rack_1.5_2_ml	32.31	<b>21.98</b>	<b>47.97</b>	17.46	32.31	<b>21.98</b>
tube_rack_50_ml	69.56	66.34	50.71	21.12	<b>74.87</b>	<b>72.35</b>
wire_stripper	71.65	54.54	55.40	15.90	<b>82.37</b>	<b>71.86</b>
wrench	16.96	4.27	33.51	0.03	<b>60.00</b>	<b>23.13</b>
average	36.23	23.32	37.72	6.53	<b>61.58</b>	<b>41.25</b>

Table 4: The results of **PVNet** [28] on single-object pose estimation in terms of **ADD(-S) AUC** and **ADD(-S) accuracy** on StereOBJ-1M dataset. The input modality include monocular and binocular stereo RGB images.

are predicted using heatmaps, and the 6D poses are calculated by solving PnP problem [5]. The results are illustrated in columns 1-2 in Tables 3 and 4. PVNet and KeyPose respectively achieve 33.90% and 36.10% in average ADD(-S) AUC and 23.52%, and 23.32% in average ADD(-S) accu-

racy. Among the objects, the performance of both baseline methods suffers especially on pipette categories, which highlights the challenge of pose estimation of visually similar but different object instances.

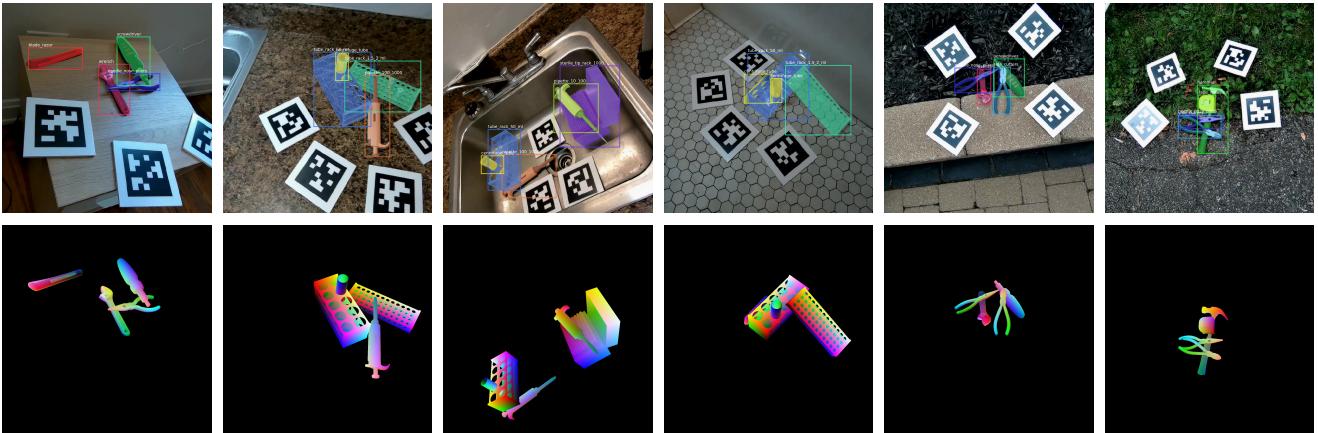


Figure 6: **Visualization of data samples from StereOBJ-1M dataset.** The first row is left stereo images with semantic masks and bounding boxes superimposed. In the second row, we use normalized coordinate map [26, 37] to illustrate the 6D poses of the corresponding objects, where the coordinates of the object surface points are normalized to  $[0, 1]^3$  and converted to RGB values in  $[0, 255]^3$  at projected pixels.

### 5.3. Stereo Image Experiments

We conduct stereo experiments where both stereo images are used as input to predict 6D pose. The monocular method PVNet [28] is adapted to its stereo variant where keypoints in both stereo images are predicted individually. For both baseline methods, suppose  $[u_k^L, v_k^L]$  and  $[u_k^R, v_k^R]$  are the predicted 2D locations of  $k$ -th keypoint in left and right cameras respectively,  $\Pi_L$  and  $\Pi_R$  are the camera projection of left and right cameras respectively, and  $\mathbf{x}_k^* \in \mathbb{R}^3$  is the  $k$ -th keypoint in the canonical object pose. We investigate the following two methods for calculating 6D pose from keypoints predictions in both stereo images.

**Classic Triangulation.** Given  $[u_k^L, v_k^L]$  and  $[u_k^R, v_k^R]$  for all  $k$ , a naïve method to compute object 6D pose is to follow classic point-level triangulation used in KeyPose [18], i.e. triangulate 3D keypoints from stereo and fit them to canonical object 3D keypoints by solving an Orthogonal Procrustes problem, to obtain the estimated pose  $[\mathbf{R}_c|\mathbf{T}_c]$ :

$$\begin{aligned} \mathbf{x}_k &= \arg \min_{\mathbf{x} \in \mathbb{R}^3} \|\Pi_L(\mathbf{x}) - [u_k^L, v_k^L]\|_2^2 + \|\Pi_R(\mathbf{x}) - [u_k^R, v_k^R]\|_2^2 \\ [\mathbf{R}_c|\mathbf{T}_c] &= \arg \min_{\mathbf{R}, \mathbf{T}} \sum_{k=1}^K \|(\mathbf{R}\mathbf{x}_k^* + \mathbf{T}) - \mathbf{x}_k\|_2^2 \quad (3) \end{aligned}$$

where  $\mathbf{x}_k \in \mathbb{R}^3$  is the triangulated 3D location of the  $k$ -th keypoint. The second step in (3) can use RANSAC [5].

**Object Triangulation.** We propose a novel object-level triangulation approach as a stronger baseline. Compared to classic triangulation which optimizes the 3D location of a *point*, we directly optimize the 6D pose of an object from 2D keypoint predictions in both images. Mathematically, Object Triangulation combines the two steps in Equation

(3) into one unified optimization of the 6D pose  $[\mathbf{R}_o|\mathbf{T}_o]$ :

$$[\mathbf{R}_o|\mathbf{T}_o] = \arg \min_{\mathbf{R}, \mathbf{T}} \sum_{k=1}^K \|\Pi_L(\mathbf{R}\mathbf{x}_k^* + \mathbf{T}) - [u_k^L, v_k^L]\|_2^2 + \|\Pi_R(\mathbf{R}\mathbf{x}_k^* + \mathbf{T}) - [u_k^R, v_k^R]\|_2^2 \quad (4)$$

We use the Levenberg-Marquardt algorithm [24] as the non-linear optimization method together with RANSAC [5]. The results of the two baseline architectures with two pose optimization methods are illustrated in columns 3-6 in Tables 3 and 4. Baseline methods with Object Triangulation consistently improve over monocular variants on all object categories significantly while classic triangulation can yield worse results. With Object Triangulation, the stereo variants of both baseline methods significantly outperform their monocular counterparts on StereOBJ-1M, by at least 25% in ADD(-S) AUC and 14% in ADD(-S) accuracy.

## 6. Conclusions

In this work, we propose a novel object pose data capturing and annotation pipeline and present a large-scale object pose dataset with stereo RGB as input. We benchmark two state-of-the-art algorithms for 6D object pose estimation and propose a novel method for stereo object pose optimization that outperforms classic triangulation method. In addition to pose estimation, our dataset enables future research directions such as object reconstruction and scene flow estimation [19, 20] from stereo RGB.

**Acknowledgement.** This work is funded in part by JST AIP Acceleration, Grant Number JPMJCR20U1, Japan.

## References

- [1] Berk Calli, Aaron Walsman, Arjun Singh, Siddhartha Srinivas, Pieter Abbeel, and Aaron M. Dollar. Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols. *CoRR*, abs/1502.03143, 2015. [2](#) [3](#)
- [2] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *TPAMI*, 2019. [2](#)
- [3] Rui Chen, Songfang Han, Jing Xu, and Hao Su. Point-based multi-view stereo network. In *ICCV*, 2019. [2](#)
- [4] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Grasnet-1billion: A large-scale benchmark for general object grasping. In *CVPR*, 2020. [2](#) [3](#) [4](#)
- [5] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981. [3](#) [6](#) [7](#) [8](#)
- [6] John C Gower. Generalized procrustes analysis. *Psychometrika*, 1975. [4](#)
- [7] Till Grenzförfner, Martin Günther, and Joachim Hertzberg. Ycb-m: A multi-camera rgb-d dataset for object recognition and 6dof pose estimation. In *ICRA*, 2020. [3](#)
- [8] Yisheng He, Wei Sun, Haibin Huang, Jianran Liu, Haoqiang Fan, and Jian Sun. Pvnet3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation. In *CVPR*, 2020. [1](#) [2](#)
- [9] Stefan Hinterstoisser, Stefan Holzer, Cedric Cagniart, Slobodan Ilic, Kurt Konolige, Nassir Navab, and Vincent Lepetit. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In *ICCV*, 2011. [2](#) [3](#) [4](#) [6](#)
- [10] Tomáš Hodan, Pavel Haluza, Štepán Obdržálek, Jiri Matas, Manolis Lourakis, and Xenophon Zabulis. T-less: An rgb-d dataset for 6d pose estimation of texture-less objects. In *WACV*, 2017. [1](#) [2](#) [3](#) [4](#) [10](#)
- [11] WEEVIEW INC. weeview: 3d camera. <https://www.weeview.co/>. [10](#)
- [12] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Scott Godisart, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social interaction capture. *TPAMI*, 2017. [2](#) [4](#)
- [13] Kyungmin Kim and Hyunjung Shim. Robust approach to reconstructing transparent objects using a time-of-flight depth camera. *Optics express*, 2017. [1](#)
- [14] Jogendra Nath Kundu, MV Rahul, Aditya Ganeshan, and R Venkatesh Babu. Object pose estimation from monocular image using multi-view keypoint correspondence. In *ECCV*, 2018. [2](#)
- [15] Yann Labb  , Justin Carpentier, Mathieu Aubry, and Josef Sivic. Cosypose: Consistent multi-view multi-object 6d pose estimation. In *ECCV*, 2020. [2](#)
- [16] Chi Li, Jin Bai, and Gregory D Hager. A unified framework for multi-view multi-class object pose estimation. In *ECCV*, 2018. [2](#)
- [17] Peiliang Li, Xiaozhi Chen, and Shaojie Shen. Stereo r-cnn based 3d object detection for autonomous driving. In *CVPR*, 2019. [2](#)
- [18] Xingyu Liu, Rico Jonschkowski, Anelia Angelova, and Kurt Konolige. Keypose: Multi-view 3d labeling and keypoint estimation for transparent objects. In *CVPR*, 2020. [1](#) [2](#) [3](#) [4](#) [6](#) [7](#) [8](#) [10](#)
- [19] Xingyu Liu, Charles R Qi, and Leonidas J Guibas. Flownet3d: Learning scene flow in 3d point clouds. In *CVPR*, 2019. [8](#)
- [20] Xingyu Liu, Mengyuan Yan, and Jeannette Bohg. Meteor-net: Deep learning on dynamic 3d point cloud sequences. In *ICCV*, 2019. [8](#)
- [21] SHINING 3D Tech. Co. Ltd. Einscan pro 2x plus. <https://www.einscan.com/handheld-3d-scanner/2x-plus/>. [5](#)
- [22] Lucas Manuelli, Wei Gao, Peter Florence, and Russ Tedrake. Keypoint affordances for category-level robotic manipulation. In *ISRR*, 2019. [2](#) [3](#)
- [23] Pat Marion, Peter R Florence, Lucas Manuelli, and Russ Tedrake. Label fusion: A pipeline for generating ground truth labels for real rgbd data of cluttered scenes. In *ICRA*, 2018. [2](#) [3](#)
- [24] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM*, 1963. [8](#)
- [25] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3DV*, 2017. [2](#)
- [26] Kiru Park, Timothy Patten, and Markus Vincze. Pix2pose: Pix2pose: Pixel-wise coordinate regression of objects for 6d pose estimation. In *ICCV*, 2019. [8](#) [12](#)
- [27] Georgios Pavlakos, Xiaowei Zhou, Aaron Chan, Konstantinos G Derpanis, and Kostas Daniilidis. 6-dof object pose from semantic keypoints. In *ICRA*, 2017. [2](#)
- [28] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Jun Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *CVPR*, 2019. [1](#) [2](#) [6](#) [7](#) [8](#) [10](#)
- [29] Nuno Pereira and Lu  s A Alexandre. Maskedfusion: mask-based 6d object pose estimation. In *ICMLA*, 2020. [1](#)
- [30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. [10](#)
- [31] Shreeyak Sajjan, Matthew Moore, Mike Pan, Ganesh Nagaraja, Johnny Lee, Andy Zeng, and Shuran Song. Clear grasp: 3d shape estimation of transparent objects for manipulation. In *ICRA*, 2020. [1](#)
- [32] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *ECCV*, 2018. [2](#)
- [33] Supasorn Suwajanakorn, Noah Snavely, Jonathan J Tompson, and Mohammad Norouzi. Discovery of latent 3d keypoints via end-to-end geometric reasoning. In *NIPS*, 2018. [2](#)
- [34] Meng Tian, Liang Pan, Marcelo H Ang, and Gim Hee Lee. Robust 6d object pose estimation by learning rgbd features. In *ICRA*, 2020. [1](#)
- [35] Jonathan Tremblay, Thang To, and Stan Birchfield. Falling things: A synthetic dataset for 3d object detection and pose estimation. In *CVPR Workshops*, 2018. [1](#) [2](#)
- [36] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Mart  n-Mart  n, Cewu Lu, Li Fei-Fei, and Silvio Savarese. Densefusion: 6d

- object pose estimation by iterative dense fusion. In *CVPR*, 2019. 1
- [37] 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 *CVPR*, 2019. 1, 2, 3, 5, 8, 12
- [38] John Wang and Edwin Olson. Apriltag 2: Efficient and robust fiducial detection. In *IROS*, 2016. 10
- [39] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *RSS*, 2018. 6
- [40] Chi Xu, Jiale Chen, Mengyang Yao, Jun Zhou, Lijun Zhang, and Yi Liu. 6dof pose estimation of transparent object from a single rgb-d image. *Sensors*, 2020. 1
- [41] Guorun Yang, Hengshuang Zhao, Jianping Shi, Zhidong Deng, and Jiaya Jia. Segstereo: Exploiting semantic information for disparity estimation. In *ECCV*, 2018. 2

## A. Overview

In this document, we provide additional details on StereOBJ-1M dataset as presented in the main paper. We present additional baseline results on instance-level pose detection for `centrifuge_tube` class in Section B. In Section C, we provide details on the hardware of data capturing. In Section D, we provide more details on viewpoint distribution of each object class. Lastly, in Section E, we visualize more data samples from our dataset.

## B. Multi-instance Pose Detection Results

In the main paper, we report the results of two baselines on **single-object pose estimation** of 17 out of 18 objects on the test set where there is at most one object instance from a category in a scene. However, for `centrifuge_tube`, there are usually multiple instances recorded in a scene. Therefore, `centrifuge_tube` is used in **multi-object pose detection** task. In this task, the framework is supposed to perform instance-level detection and pose estimation simultaneously.

To adapt to instance-level pose detection, we modify the baseline formulation by introducing additional 2D object detection before pose estimation. Given a detected rough 2D bounding box of an object instance, we crop the image patch and send it to pose estimation baselines, i.e. PVNet [28] and KeyPose [18], to estimate the 2D keypoint locations and therefore 6D pose of that object instance. The 2D object detector we used is Faster-RCNN [30].

We use Average Precision (AP) as the evaluation metrics of multi-instance pose detection. When calculating AP in 2D object detection, a detection result is considered correct if the IoU between the detected bounding box and a ground truth bounding box is larger than a threshold. Different from 2D object detection, we consider a pose detection result to be correct if the ADD(-S) distance between the detected 6D

Table 5: AP results of **object pose detection** with single RGB image as input.

method	PVNet [28]	KeyPose [18]
<code>centrifuge_tube</code>	15.19	17.64

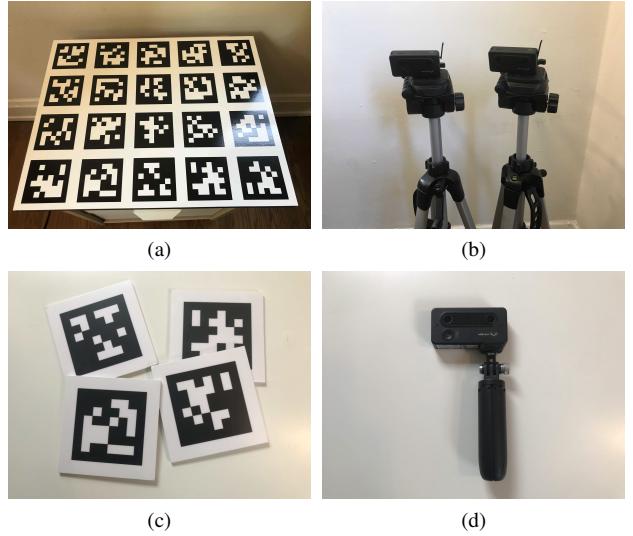


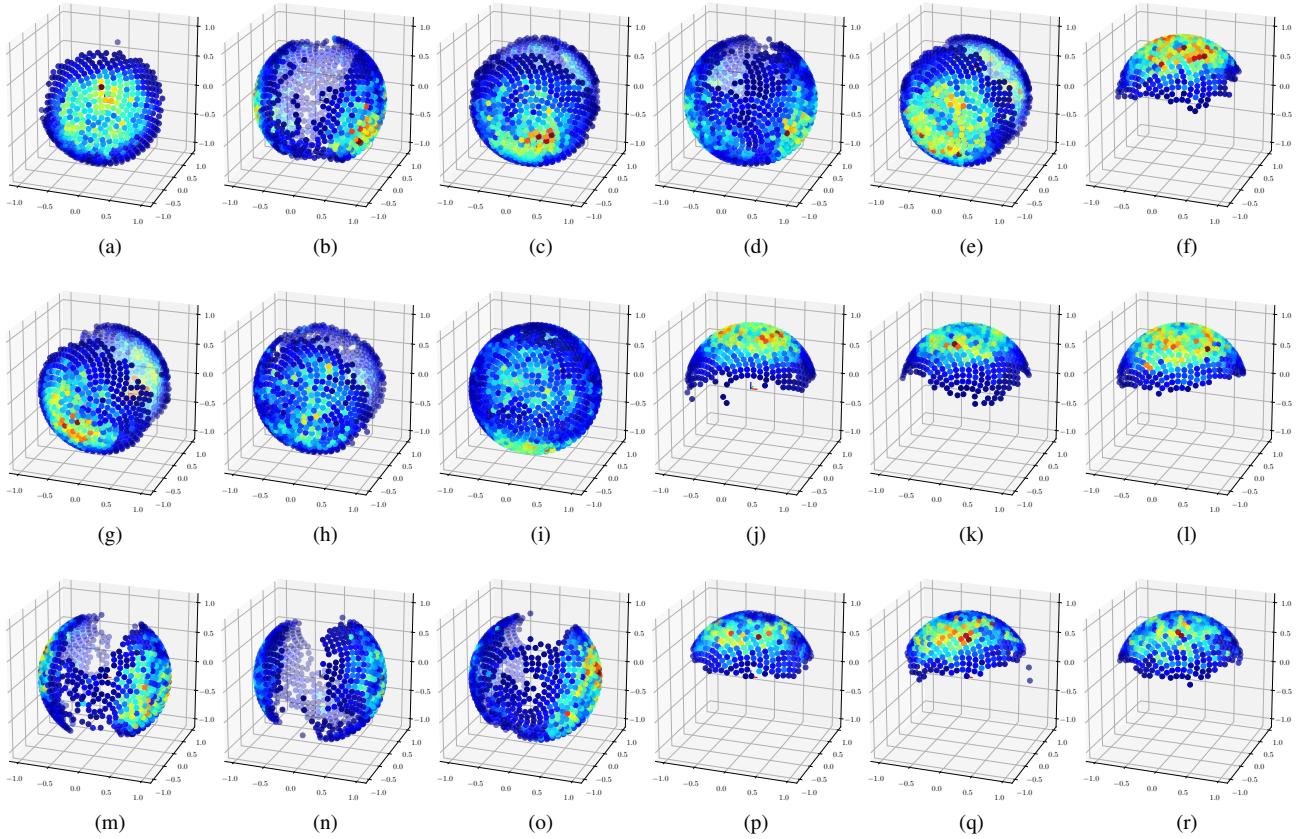
Figure 7: **Hardware for data collection.** (a) large fiducial marker board; (b) static cameras with tripods; (c) small fiducial markers; (d) moving stereo camera.

pose and a ground truth pose is smaller than a threshold. We use 10% of the object diameter as the threshold of ADD(-S). We report the pose detection results with single-RGB image as input in Table 5. We notice that the above baseline suffers when two or multiple instances object overlap in the image and are included in the same image patch. In this case, the pose estimation framework cannot distinguish different instances and fails in keypoint prediction.

## C. Data Capturing Hardware

We present the hardware used for capturing the data in Figure 7, including a large fiducial marker board, several small fiducial markers, two static cameras with two tripods, and one moving stereo camera. We used the same Weewiew stereo camera [11] for all three cameras, though the two stereo cameras can be monocular. Weewiew stereo camera has a stereo baseline of approximately 4.5cm which is close to the distance between the two human eyes. All three cameras are calibrated.

The fiducial markers are the first 20 AprilTags [38]. The large fiducial marker board is printed on a 20in × 16in plastic picture frame. Though the large fiducial marker board needs to be accurately measured by its physical dimensions with a vernier caliper, the small fiducial markers do not.



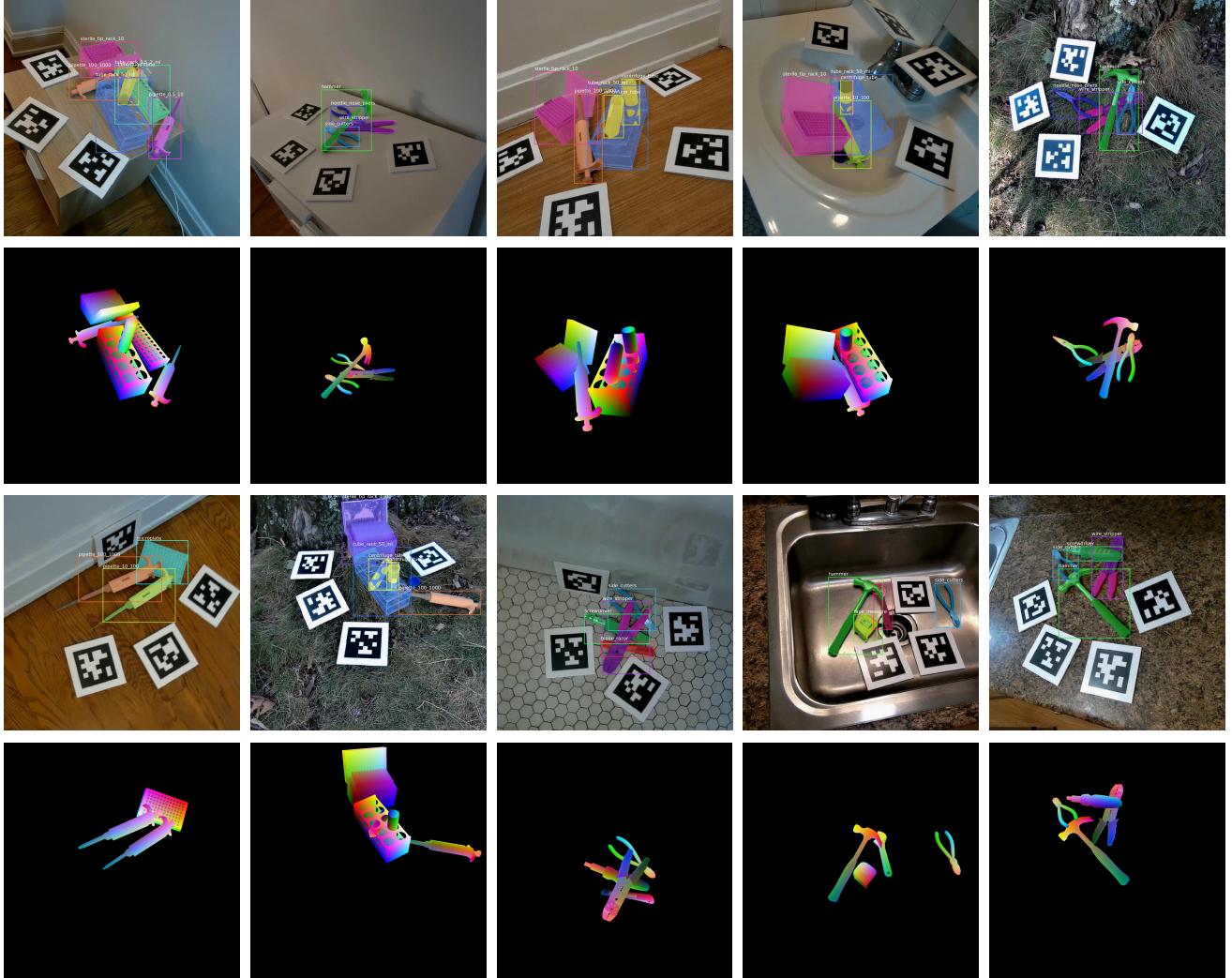
**Figure 8: Viewpoint distribution of the 18 objects in our dataset.** (a) blade\_razor; (b) hammer; (c) needle\_nose\_pliers (d) screwdriver; (e) side\_cutters; (f) tape\_measure; (g) wire\_stripper; (h) wrench; (i) centrifuge\_tube; (j) microplate; (k) tube\_rack\_2; (l) tube\_rack\_50; (m) pipette\_0.5\_10; (n) pipette\_10\_100; (o) pipette\_100\_1000; (p) sterile\_rack\_10; (q) sterile\_rack\_200; (r) sterile\_rack\_1000.

## D. Viewpoint Coverage Distribution

Viewpoint coverage percentage is illustrated in Section 4.2 and Figure 5 of the main paper. We illustrate a more detailed viewpoint distribution for each object in Figure 8. The viewpoints are drawn as 3D points on the unit sphere centered at the object center. Their positions on the unit sphere are determined by the Azimuth and Elevation of the viewpoint. Their density on the sphere is shown by heatmap color. Notice that for objects such as microplate and tape\_measure, there is no viewpoint distributed on the  $-z$  space, because there is only one possible side up when being put on a desktop.

## E. More Visualizations of Data Samples

We provide more visualizations of data samples from our dataset. As illustrated in Figure 9, the data annotation has high precision.



**Figure 9: Visualization of data samples from StereOBJ-1M dataset.** The first row is left stereo images with semantic masks and bounding boxes superimposed. In the second row, we use normalized coordinate map [26, 37] to illustrate the 6D poses of the corresponding objects, where the coordinates of the object surface points are normalized to  $[0, 1]^3$  and converted to RGB values in  $[0, 255]^3$  at projected pixels.