Imitrob

Imitation Learning Dataset for Training and Evaluating 6D Object Pose Estimators

Sedlar, Jiri; Stepanova, Karla; Skoviera, Radoslav; Behrens, Jan K.; Tuna, Matus; Sejnova, Gabriela; Sivic, Josef; Babuska, Robert

DOI
10.1109/LRA.2023.3259735

Publication date
2023

Document Version
Final published version

Published in
IEEE Robotics and Automation Letters

Citation (APA)
Sedlar, J., Stepanova, K., Skoviera, R., Behrens, J. K., Tuna, M., Sejnova, G., Sivic, J., & Babuska, R. (2023). Imitrob: Imitation Learning Dataset for Training and Evaluating 6D Object Pose Estimators. *IEEE Robotics and Automation Letters*, *8*(5), 2788-2795. https://doi.org/10.1109/LRA.2023.3259735

Imitrob: Imitation Learning Dataset for Training and Evaluating 6D Object Pose Estimators

Jiri Sedlar, Karla Stepanova, Radoslav Skoviera, Jan K. Behrens, *Member, IEEE*, Matus Tuna, Gabriela Sejnova, Josef Sivic, and Robert Babuska, *Member, IEEE*

***Abstract*—This letter introduces a dataset for training and evaluating methods for 6D pose estimation of hand-held tools in task demonstrations captured by a standard RGB camera. Despite the significant progress of 6D pose estimation methods, their performance is usually limited for heavily occluded objects, which is a common case in imitation learning, where the object is typically partially occluded by the manipulating hand. Currently, there is a lack of datasets that would enable the development of robust 6D pose estimation methods for these conditions. To overcome this problem, we collect a new dataset (Imitrob) aimed at 6D pose estimation in imitation learning and other applications where a human holds a tool and performs a task. The dataset contains image sequences of nine different tools and twelve manipulation tasks with two camera viewpoints, four human subjects, and left/right hand. Each image is accompanied by an accurate ground truth measurement of the 6D object pose obtained by the HTC Vive motion tracking device. The use of the dataset is demonstrated by training and evaluating a recent 6D object pose estimation method (DOPE) in various setups.**

***Index Terms*—Learning from demonstration, computer vision for automation, perception for grasping and manipulation, 6D object pose estimation.**

Manuscript received 30 September 2022; accepted 28 February 2023. Date of publication 20 March 2023; date of current version 3 April 2023. This letter was recommended for publication by Associate Editor G. Averta and Editor M. Vincze upon evaluation of the reviewers' comments. This work was supported in part by European Regional Development Fund through Project IMPACT under Grant CZ.02.1.01/0.0/0.0/15_003/0000468, in part by European Regional Development Fund through the Project Robotics for Industry 4.0 under Grant CZ.02.1.01/0.0/0.0/15_003/0000470, in part by VEGA under Grant 1/0796/18, in part by MPO TRIO Project under Grant FV40319, in part by CTU Student Grant Agency under Grant SGS21/184/OHK3/3T/37, and in part by the Czech Science Foundation under Grant GA21-31000S. *(Corresponding author: Karla Stepanova.)*

Jiri Sedlar, Karla Stepanova, Radoslav Skoviera, Jan K. Behrens, Gabriela Sejnova, and Josef Sivic are with the Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague, 16000 Prague, Czech Republic (e-mail: jiri.sedlar@cvut.cz; karla.stepanova@cvut.cz; radoslav.skoviera@cvut.cz; jan.kristof.behrens@cvut.cz; sejnogab@fel.cvut.cz; josef.sivic@cvut.cz).

Matus Tuna is with the Faculty of Mathematics, Physics and Informatics, Comenius University in Bratislava, 82102 Bratislava, Slovakia (e-mail: tunamatus@gmail.com).

Robert Babuska is with the Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague, 16000 Prague, Czech Republic, and also with the Cognitive Robotics, Faculty of 3mE, Delft University of Technology, 2628CD Delft, The Netherlands (e-mail: r.babuska@tudelft.nl).

The dataset and code are publicly available at http://imitrob.ciirc.cvut.cz/imitrobdataset.php.

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2023.3259735, provided by the authors.

Digital Object Identifier 10.1109/LRA.2023.3259735

I. Introduction

DESPITE the recent progress [1], [2], 6D object pose estimators have rarely been applied to image sequences capturing manipulation of hand-held objects. However, such a set-up has a huge potential in imitation learning scenarios where expert demonstrations are used to teach robots new tasks (e.g. a human demonstrator manipulating a glue gun to apply glue along specified trajectories). One of the reasons is that there are no datasets and benchmarks that would allow training for such setups. To overcome this problem, we have collected and annotated a real-world hand-held tool manipulation dataset (*Imitrob*) that allows training and evaluating 6D object pose estimators in such conditions.

Acquiring data from videos of humans demonstrating a task has several potential advantages over other approaches such as kinesthetic teaching [3], teleoperation [4], or motion tracking systems with markers on the objects or human body parts [5]. First, such a setup can provide more detailed information about the interaction of the tool with the environment. Second, it is much easier for a skilled worker to perform the task in the usual way rather than to demonstrate it by holding a robot arm. Third, such a setup enables easier transfer to different robotic platforms. Finally, vast amounts of visual data are already available (e.g. instructional videos on YouTube) and can be used for learning in industrial, household, and similar settings.

However, there are also several critical challenges that need to be addressed. First, one has to deal with the fact that hand-held objects are partially occluded by the demonstrator, exhibit various symmetries, and lack a distinctive texture. These characteristics make training of 6D object pose estimators difficult. It is also hard to estimate a priori whether the tracking accuracy will be sufficient for the robotic task at hand. Second, to guarantee a reasonably short setup time (e.g. data collection, processing, and annotation), the 6D object pose estimator must be trainable on a limited amount of demonstration data. Finally, the 6D pose estimator should preferably work without a 3D model of the tool. However, current model-based pose estimation methods often require high-quality object models, which are difficult to acquire in real-world imitation learning applications.

The development of data-efficient and occlusion-insensitive 6D object pose estimators requires datasets focused on manipulation with hand-held tools, as well as a methodology for evaluating the performance with regard to imitation learning tasks. Neither of these currently exists. Our paper addresses this problem and provides tools that help to improve the performance of 6D object pose estimation in such challenging cases. Our main contributions include:

1) We have collected, annotated, and published a real-world hand-held tool manipulation dataset, called *Imitrob* [6]

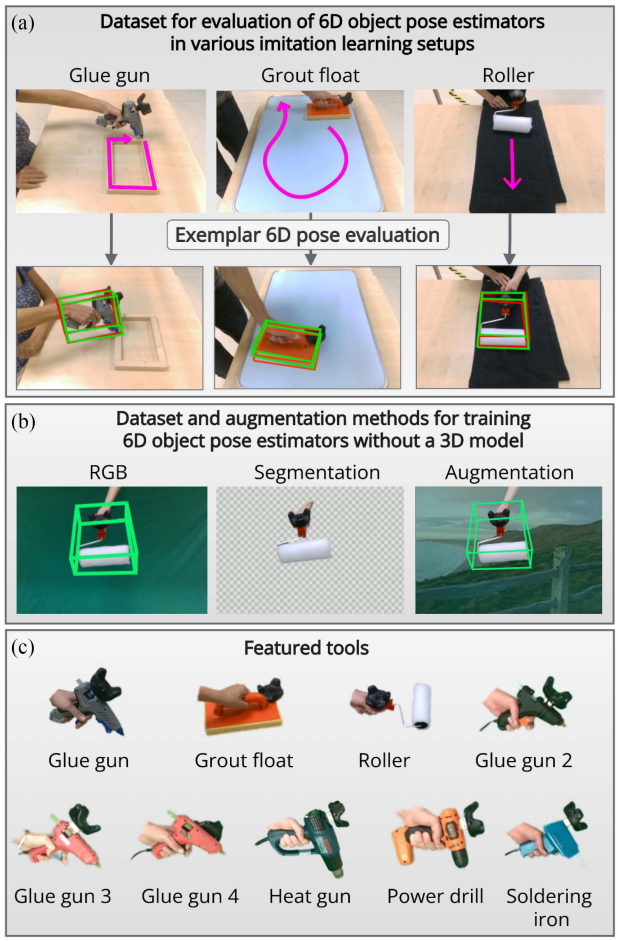


Fig. 1. *Imitrob* dataset for 6D object pose estimation of hand-held tools in real-world manipulation tasks in uninstrumented environments. (a) *ImitrobTest* dataset for benchmarking 6D object pose estimation methods, and (b) *ImitrobTrain* dataset for training 6D pose estimators without a 3D model of the tool; (c) the nine tools are in both datasets. The bounding boxes visualize the predicted (red) and reference (green) object poses.

(see Fig. 1). The dataset consists of RGB-D videos with ground truth annotations of the tool 6D poses and bounding boxes. The pose annotations are generated using the HTC Vive motion tracking system, and the bounding boxes are derived from the tracked pose and a tracing-based object surface estimation. The *Imitrob* dataset contains 9 hand-held tools (four glue guns, grout float, roller, heat gun, power drill, soldering iron), manipulated by 4 demonstrators, using left/right hand, and recorded from 2 camera viewpoints. The test part of the dataset (100 332 images, *ImitrobTest*), includes videos of 12 manipulation tasks in realistic environments. The training part of the dataset (83 778 images, *ImitrobTrain*), contains random motion of the tools in front of green background.

2) We provide a methodology (accompanied by a software package [7]) to collect ground truth training data for new objects or manipulation tasks in an affordable way. The methodology can also be used to introduce further variability into the dataset. The data acquisition methodology does not require a CAD model of the tracked tool to obtain the ground truth 6D pose. For application in industrial environments, the methodology requires only simple modifications (e.g. attaching a tracker to the tool), which we regard as essential for practical use of imitation learning in real-world set-ups, e.g. in industrial environments.
3) To illustrate how the *Imitrob* dataset can be used to compare the performance of various algorithms for hand-held object pose estimation, we trained and evaluated the accuracy of a selected 6D object pose estimator (DOPE [2]).
4) We demonstrate how the generalization capabilities of the 6D object pose estimator can be enhanced by augmentation of the *ImitrobTrain* dataset. For this purpose, we compared several data augmentation techniques; the best performance was achieved by a method that leverages the blending of the original and random background.

The dataset, code [7], and supplementary material [8] are available on the *Imitrob* project web page [6]. The supplementary material contains the calibration details (Secs. A.1-A.2), full definition of the evaluation metrics (Section A.3), details about the object segmentation methods (Section A.4), values of the DOPE estimator parameters (Section A.5), ablation studies on the impact of image resolution, batch size, and segmentation technique (Secs. A.6-A.7), complete results of all experiments, including metric values that did not fit into the main paper (Secs. A.8-A.13), comparison of the model-free estimator DOPE with a model-based object pose estimator CosyPose [1] on the power drill tool (Section A.14), and evaluation of robustness to a change in the tracker position (Section A.15).

II. Related Work

In this section, we focus on the current datasets aimed at static 6D object pose estimation and on those that include videos depicting manipulated objects for imitation learning. We also mention the state-of-the-art methods in 6D object pose estimation from RGB and RGB-D images or videos.

A. 6D Object Pose Estimation

Motivated by applications in robotics, 6D object pose estimation has recently attracted significant attention [2], [9], [10]. In the case of richly textured objects, methods based on matching of local invariant features such as SIFT [11] or SURF [12] produce reasonable results. Unfortunately, many hand-held tools are not richly textured. The more complicated 6D estimation of textureless objects can be handled by models based on Convolutional Neural Networks. Methods such as [13], [14] use CNNs to directly regress 6D object pose. In another approach, methods like [15], [16], [17], [18], [19], [20], [21] predict the correspondences between the 2D input image and either a 3D model of an object or specific keypoints on an object, which are then used to compute the object 6D pose via the PnP algorithm. The DOPE algorithm [2] is a keypoint matching method that predicts the object's 3D bounding box vertices and centroid locations in the 2D coordinate system of the input RGB image. This approach was shown to outperform other models like the PoseCNN [13]. There are also more recent methods, such as [1], [15], which estimate the 6D pose based on the alignment of 3D object models with the input images. However, these "render-and-compare" methods require a known 3D model of the object. Obtaining such accurate 3D models quickly is a nontrivial task in real-world imitation learning scenarios. Hence, we choose DOPE [2] as our exemplar 6D pose estimation method as it does not require a 3D model of an object to estimate the pose. Instead, only visual data and reference 6D pose data are needed. DOPE is thus more

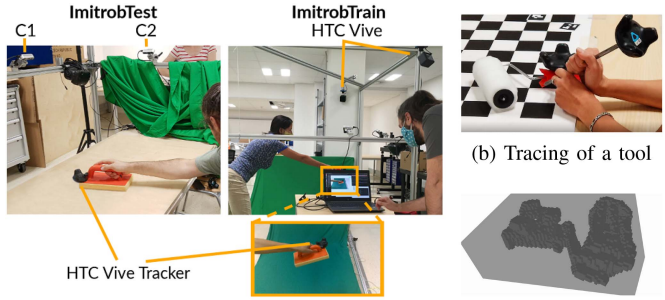
suitable for imitation learning setup as it only requires the user to record a few short training videos with the hand-held tool.

B. Datasets for 6D Object Pose Estimation

One of the frequently used static datasets for object pose estimation is Linemod [22], which consists of 15 textureless household objects with annotations and a test set that includes these objects in cluttered scenes. An extended version Linemod-Occluded [23] introduces a more challenging occluded testing scenario. The T-LESS dataset [24] features 30 objects from an industrial environment, which lack an easily discriminative texture and are symmetrical along one or more axes. Another industry-oriented dataset is ITODD [25], but its reference annotations are not available for the test images. The recent YCB-M dataset consists of real-world static scenes recorded using 7 different cameras [26]. In all of the above-mentioned datasets, the objects are static and captured by a camera moving around the object at an approximately constant distance. However, for more realistic real-world 6D object pose estimation, it is beneficial to train the estimators on datasets depicting manipulated objects. Creating such datasets is even more technically challenging. Hence, the existing datasets are small or employ methods that simplify the annotation task [13], [27]. For example, the authors of the YCB-Video dataset [13] avoided full manual annotation by keeping the recorded objects at fixed positions and moving the camera only, leading to a high correlation of the objects' relative poses throughout the data. Compared to the Linemod or YCB-Video datasets, we focus on specific tasks and tools typical for industrial manufacturing environments. A real human activity RGB dataset focused on task-oriented grasping [28] includes both synthetic and real RGB-D videos of manipulated objects. However, the annotations are mainly focused on the hand joint positions, and only a small proportion of the objects are provided with their meshes and 3D poses. Probably most related to ours is the dataset [29], which consists of three objects recorded with the Kinect sensors. Our dataset differs in the three main aspects. First, we obtain the reference 6D pose from an HTC Vive controller attached to the object, whereas [29] used manual annotation. Second, [29] provides 3 187 images in total, whereas our whole dataset contains more than 184 000 images. Third, [29] lacks variability across several subjects, left and right hand, different camera views, tasks, or clutter in the scene. Furthermore, [29] expects a known 3D model of the object and thus cannot be used for the training of 3D model-independent estimators. We are unaware of any other 6D object pose estimation video dataset besides ours that would enable the evaluation of trained models for so many different types of generalization, i.e. across different human operators, left-handed and right-handed manipulation, task variations, camera viewpoints, occlusions, and backgrounds. To demonstrate its utility, we measure the impact of each of these challenges on the accuracy of 6D object pose estimation provided by the DOPE algorithm [2].

III. Data Acquisition Setup

The basic data acquisition setup (see Fig. 2(a)) consists of a desk with two Intel RealSense D455 RGB-D cameras and an HTC Vive VR set. The data from all sensors are broadcast as Robot Operating System (ROS [30]) messages and stored in ROS bag files. The cameras produce 848 × 480 RGB-D images



(a) Data acquisition setup (b) Tracing of a tool (c) Tool bounding box

Fig. 2. Experimental setup for collection of *ImitrobTest* and *ImitrobTrain* datasets. (a) The setup consists of RGB-D cameras, HTC Vive lighthouses, and a tracker attached to the tool. (b) The surface calibration process. (c) The resulting voxel grid (dark gray) and bounding box (light gray).

at 60 Hz, and the HTC Vive produces 6D poses at 30 Hz. For the data collection, each task was performed on a table with task-related or clutter objects (see Fig. 3).

A. Sensor Setup Calibration and Data Synchronization

For the cameras, we estimated the intrinsic parameters and the radial and tangential distortion coefficients from several views of the chessboard calibration pattern using the OpenCV library [31]. The extrinsics were calibrated from a single view of the chessboard pattern. The origin of the chessboard (world) coordinate system O_{w} was defined in one of the chessboard corners, and the camera poses relative to O_{w} were estimated by solving the PnP problem in combination with the RANSAC algorithm. To calibrate the HTC Vive coordinate frame O_{htc} (in one of the lighthouses marked as HTC Vive in Fig. 2(a)) to the chessboard coordinate frame O_{w}, spherical motion patterns centered at different chessboard corners p_{w} were recorded using a tracked pointing device (tracker mounted on a pointed metal rod, shown in Fig. 2(b)). More technical details of the calibration are presented in [8]. The average deviation (residual r_{avg}) of the acquired center points from the regular chessboard grid pattern (acquired from the cameras) was below 2 mm for all experiments. The HTC Vive pose data were interpolated to calculate the reference object poses for the times when the individual camera images were captured. When the time difference between consecutive HTC Vive frames is longer than 100 ms, the corresponding camera images are discarded to ensure sufficiently accurate ground truth data.

B. HTC Vive Tracker to Tool Calibration

To be able to provide the reference bounding boxes for the *Imitrob* dataset, we first have to find the bounding boxes of the manipulated objects relative to the tracker. To find the object dimensions relative to the tracker, we traced the tool and tracker surfaces with a tracked pointing device while recording the positions of both trackers (see Fig. 2(b)). Contour tracing for surface reconstruction was described in [32]. The acquired surface points are filtered (points that are likely not part of the surface are removed), and a voxel grid with the dimensions of the object is created. Finally, we calculate a minimal bounding box aligned with the tracker's z-axis using the trimesh library [33] while evaluating volumes for different rotations. Fig. 2(c) visualizes the resulting voxel grid and bounding box for the roller. Note

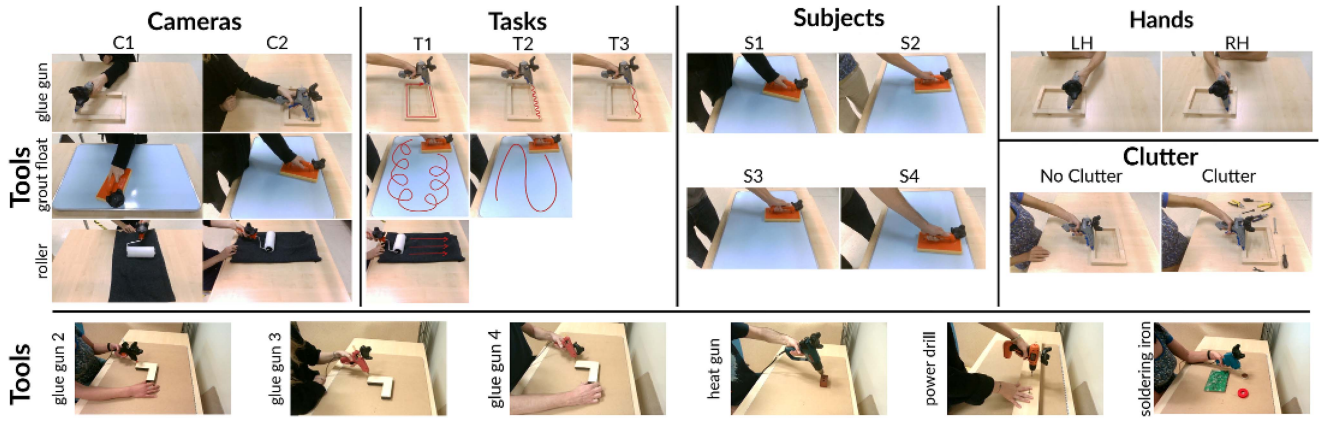


Fig. 3. Overview of the variability of setups provided in our *ImitrobTest* dataset. Tools (rows): glue gun, grout float, roller, glue gun 2, glue gun 3, glue gun 4, heat gun, power drill, soldering iron. Cameras: front (C1) and right hand side (C2), synchronized in the *ImitrobTest* dataset. Tasks: frame (T1), dense wave (T2), and sparse wave (T3) for glue gun; round (T1) and sweep (T2) for grout float; press (T1) for roller. Subjects: four demonstrators (S1, S2, S3, S4). Hands: left (LH) and right (RH) hand. Clutter: workspace with only the gluing frame (No Clutter, default) and with other objects on the table (Clutter).

that a small systematic error in the computed bounding boxes should not affect the performance of 6D pose estimators because the training and testing are executed using the same bounding box calibration. The accuracy of the pose annotations is mainly determined by the HTC Vive dynamic accuracy, which was evaluated in [34] as typically around 1 mm. The details of the whole procedure and its accuracy are described in [7] and [8].

IV. The Imitrob Dataset

Our *Imitrob* dataset provides annotated videos of manipulation tasks with hand-held tools in settings simulating a controlled factory environment. The motivation for the tools used in the *Imitrob* dataset comes from actual industrial cases. For instance, glue guns are used in the production of aerospace equipment for airplanes, including baggage bins, trolleys etc., among many other applications. Due to the large variability in this equipment, a large amount of repetitive manual labor is involved, which is very difficult to automate. Another example is the sealing of plastic foil in car doors, where rollers are used to press the foil against the metal frame on which glue has been applied. In contrast, there is less need for imitation learning for tools such as saws and screwdrivers, which are typically part of specialized robot end-effectors and thus already commonly used in many robotic applications.

We see the following three main usages of the provided dataset and methods: 1) Benchmarking 6D pose estimation methods for hand-held tools in manipulation tasks; 2) Methodology for data acquisition and 6D pose estimator training for new tools/tasks; and 3) Guideline for collecting more extensive datasets and benchmarking 6D object pose estimators on tasks with hand-held tools, e.g. in imitation learning.

The *Imitrob* dataset consists of: 1) *ImitrobTest* dataset and evaluation metrics for benchmarking 6D object pose estimation methods and 2) *ImitrobTrain* dataset and augmentation methods for training 6D pose estimators that do not require a 3D model of the object. The following sections describe these components in detail. The dataset and methods can be downloaded at [6].

A. ImitrobTest: *Benchmarking Dataset*

The *ImitrobTest* dataset (see Fig. 3) provides real-world benchmarking data for 6D object pose estimation in an imitation learning setup. It enables the evaluation of various setup combinations that one typically expects in the case of imitation learning in industrial settings. These variations include the manipulated tool, performed task, camera viewpoint, demonstrating subject, hand used for manipulation, or presence of clutter in the scene. In total, there are 208 different tool/task/camera/hand/demonstrator/clutter combinations in the *ImitrobTest* dataset. Inspired by common trajectory-dependent industrial tasks, we focus on the scenario where the robot is observing a manipulation of a tool by a human operator in order to imitate the demonstrated trajectories. The operator holds the tool in one hand and performs various tasks, such as applying hot glue with a glue gun along various trajectories, polishing a surface with a grout float, or flattening a cloth with a roller. To learn from such demonstrations, the robot has to identify the 6D pose of the tool.

1) Objects and Environment Setups: The *Imitrob* dataset features nine tools (glue gun, grout float, roller, glue gun 2, glue gun 3, glue gun 4, heat gun, power drill, soldering iron), four demonstrators (subjects S1-S4), and manipulations by the left (LH) and right (RH) hand. The 6D poses of the tools were measured by the HTC Vive see Section III-B, and the image data were recorded using two RGB-D cameras from the front (C1) and right-hand side (C2) viewpoints (see Fig. 2(a)). While we do not utilize the depth component in this work, it is included in the published dataset, along with the raw data and the code for custom data extraction. We also included challenging, textureless and small tools. The dataset contains multiple glue gun tools to enable testing how pose estimators generalize to different objects of the same type; while glue gun, glue gun 2, and glue gun 3 differ in color and size, glue gun 3 and glue gun 4 differ in the position of the HTC Vive tracker (on top vs. left side, respectively). The power drill provides a 3D model in the YCB Object set [35], which allows evaluation of model-based object pose estimation methods on this tool.

2) Tasks: The *ImitrobTest* dataset contains twelve tasks with different tool trajectories: three for glue gun, two for grout float, and one for each other tool (see Fig. 3 Tasks). In addition, the glue gun frame task was recorded in two environments: with only the gluing frame on the table (*NoClutter*, default) and with a clutter of other objects around the gluing frame (*Clutter*) (see Fig. 3 Clutter). Each task was performed by all four demonstrators (S1-S4) to simulate the variability of tool manipulation by humans. The dataset can

thus be used for learning task-specific motions from human demonstration.

3) Labeling of the Data: All RGB-D images collected in the *Imitrob* dataset are accompanied by a reference 6D pose of the tool. The 6D poses were acquired from HTC Vive at 30 Hz frequency and interpolated to match the timestamps of the camera frames (see Section III-A for the HTC Vive calibration details). The *ImitrobTest* dataset contains 100 332 annotated frames.

4) Evaluation Metrics: To evaluate the performance of 6D object pose estimators on the *ImitrobTest* dataset, we use the following three metrics (further details are available in Section A.3 of the supplementary material [8]):

1) The ADD pass rate (ADD_t) measures the percentage of predictions (P) with ADD value lower than a selected threshold (t):

$$\mathrm{ADD}_t = \frac{|\{P|\ \mathrm{ADD} \leq t\}|}{|\{P\}|} \cdot 100\% \tag{1}$$

where ADD [2] is the average distance between the corresponding predicted (p^i_{pre}) and reference (p^i_{ref}) vertices ($p^1, \ldots, p^8$) and centroid (p^9) of the object bounding box:

$$\mathrm{ADD} = \frac{1}{9}\sum_{i=1}^{9} ||p^i_{\mathrm{pre}} - p^i_{\mathrm{ref}}||_2. \tag{2}$$

A higher ADD_t value for a given threshold t indicates a better prediction accuracy of the object 3D bounding box. The ADD_t metric is useful in imitation learning where we are interested in the absolute error regardless of the size of the manipulated object. For comparison of models trained with ($\mathrm{ADD}^{\mathrm{aug}}_t$) and without ($\mathrm{ADD}^{\mathrm{noaug}}_t$) augmentation, we use the ratio of their respective ADD pass rates:

$$\mathrm{ADD}^{\mathrm{ratio}}_t = \frac{\mathrm{ADD}^{\mathrm{aug}}_t}{\mathrm{ADD}^{\mathrm{noaug}}_t}. \tag{3}$$

A higher $\mathrm{ADD}^{\mathrm{ratio}}_t$ value indicates a bigger benefit of the augmentation.

2) The rotation error (E_{rot}) measures the angle between the predicted (R_{pre}) and reference (R_{ref}) object orientations:

$$E_{\mathrm{rot}} = \arccos\left(\frac{\mathrm{trace}({R_{\mathrm{pre}}}^{-1} R_{\mathrm{ref}}) - 1}{2}\right). \tag{4}$$

A lower E_{rot} value corresponds to a better estimate of the object orientation.

3) The translation error (E_{tra}) measures the distance between the predicted (t_{pre}) and reference (t_{ref}) object positions:

$$E_{\mathrm{tra}} = ||t_{\mathrm{pre}} - t_{\mathrm{ref}}||_2. \tag{5}$$

A lower E_{tra} value indicates a better localization of the object in space.

B. ImitrobTrain: *Training Dataset*

The *ImitrobTrain* dataset (see Fig. 4) is designed for training 6D object pose estimation methods that do not require a 3D model of the object. Instead of creating a complex 3D model, the dataset captures the tools in various orientations to provide sufficient viewpoint variability for 6D object pose training. Each tool was moved randomly in one hand for a short time (20-40 s) to simulate the range of possible 6D poses during tasks. We designed this process to allow for a time and cost efficient (e.g. without the requirement for manual annotation) extension of the dataset to new tools, as 6D pose estimation methods typically require object-specific training sets.

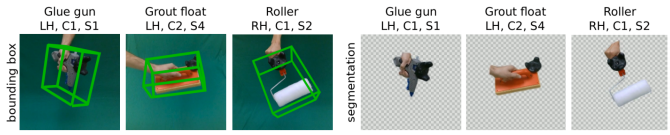


Fig. 4. Example frames from our *ImitrobTrain* dataset. *Left:* Bounding box of the tool computed from the HTC Vive data. *Right:* Segmentation of the tool and hand computed by the *MaskFBA* method. "LH, C1, S1," for example, denotes left hand, front camera, and first subject see Section IV-B. The images are cropped to show finer details.

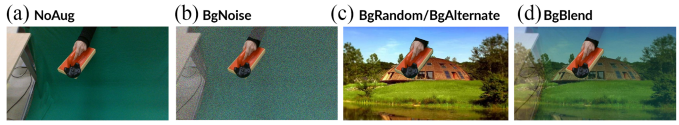


Fig. 5. Augmentation of a frame from the *ImitrobTrain* dataset. (a) Original image (*NoAug*) and background augmentation by (b) *BgNoise*, (c) *BgRandom* or *BgAlternate*, and (d) *BgBlend*.

Similarly to the *ImitrobTest* dataset, the *ImitrobTrain* dataset contains the same nine tools, four demonstrators (S1-S4), left and right hand (LH and RH), and two cameras (C1 and C2). In total, there are 144 different tool/camera/hand/demonstrator combinations in the *ImitrobTrain* dataset. In contrast to the *ImitrobTest* dataset, which contains task-specific motions and environments, in the *ImitrobTrain* dataset the tools were randomly rotated in front of a green background, which enables automatic object segmentation for background augmentation (see [6]). We include the segmented data as well as the reference 6D object pose from HTC Vive in the published *ImitrobTrain* dataset. The *ImitrobTrain* dataset contains 83 778 frames. The size of the dataset corresponds to the application area of imitation learning in industrial settings, where real data has to be collected from human demonstrators.

1) Augmentation Methods: We provide a collection of data augmentation methods suitable for the *ImitrobTrain* dataset (code at [7]). The augmentation significantly increases the size of the training set and robustness of the trained model to the variability of the test environment. First, we leverage the green background to segment the tool and hand by thresholding (*MaskThresholding*), followed by F, B, Alpha Matting [36] (*MaskFBA*) (see Fig. 4, for details see Section A.4 in [8]). Then we apply a random crop (constrained to keep all vertices of the 3D bounding box inside the image) and horizontal flip, and one of the following background randomization techniques. *BgRandom* replaces the background with a random image [37]. The other three methods keep the original background for 25% of the training images, and for the remaining 75% *BgAlternate* replaces the background with a random image, *BgBlend* blends the background with a random image, and *BgNoise* blends the background with random color noise. In our experiments, the random images were sampled from the miniImageNet [38] dataset of 60 000 images. Fig. 5 shows a training image without augmentation (*NoAug*) and after augmentation by these methods.

TABLE I
COMPARISON OF DATA AUGMENTATION METHODS (SEE SECTION IV-B)

ADD_5	*NoAug*	*BgNoise*	*BgRandom*	*BgAlternate*	*BgBlend*
glue gun	17.0	18.9	36.5	45.4	**57.8**
grout float	45.4	43.6	60.4	71.8	**73.9**
roller	25.3	26.3	39.8	**52.2**	48.6
average	29.2	29.6	45.6	56.5	**60.1**
$\mathrm{ADD}_5^{\mathrm{ratio}}$	-	1.0	1.6	1.9	**2.1**

TABLE II
GENERALIZATION ACROSS CAMERA VIEWPOINTS, LEFT/RIGHT, AND DEMONSTRATORS (SEE SECTION V) FOR TOOLS GLUE GUN, GROUT FLOAT, AND ROLLER (AVERAGE VALUES)

ADD_5	Training camera	Test C1	Test C2	Training hand	Test LH	Test RH	Subject scenario	Test
BgBlend	Same	56.3	**52.2**	Same	57.2	40.2	AllToAll	**58.8**
	Other	0.1	1.1	Other	21.0	27.0	ThreeToDiff	52.0
	Both	**69.3**	49.0	Both	**60.5**	**57.0**	OneToSame	31.1
NoAug	Same	28.7	41.8	Same	29.4	22.5	AllToAll	28.9
	Other	0.0	0.8	Other	13.7	18.4	ThreeToDiff	22.3
	Both	23.5	34.8	Both	31.3	25.6	OneToSame	18.7
$\mathrm{ADD}_5^{\mathrm{ratio}}$	Same	2.0	1.2	Same	**1.9**	1.8	AllToAll	2.0
	Other	-	1.4	Other	1.5	1.5	ThreeToDiff	**2.3**
	Both	**2.9**	**1.4**	Both	1.9	**2.2**	OneToSame	1.7

V. EXEMPLAR 6D OBJECT POSE ESTIMATION PERFORMANCE

In the following experiments we demonstrate the utility of the *Imitrob* dataset for 6D object pose estimation training and testing. For this purpose we use the 6D object pose estimator DOPE [2] (see Section A.5 in [8] for implementation details). We train all models on the *ImitrobTrain* dataset (see Section IV-B) and evaluate their performance on the *ImitrobTest* dataset (see Section IV-A). We compute the ADD pass rates for threshold $t = 5$ cm (ADD_5), as well as the rotation (E_{rot}) and translation (E_{tra}) errors. First, we present an ablation study to motivate our choice of the data augmentation method (computed for glue gun, grout float, and roller). Then we focus on the ability of the 6D object pose estimator to generalize to various training/test setups, including combinations of front/side camera, left/right hand, subjects (all computed for glue gun, grout float, and roller), and background clutter (glue gun task frame). Finally, we report performance for each tool and manipulation task. Full results, including ADD values for thresholds $t = 2$ cm (ADD_2) and 10 cm (ADD_{10}), ablation studies on the impact of image resolution, batch size, and segmentation method, comparison of model-free DOPE and model-based CosyPose object pose estimation methods on the power drill tool, as well as robustness to a different tracker position between tools glue gun 3 and glue gun 4are available in the supplementary material [8] (see Secs. A.6-A.15).

A. Ablation Experiments

1) Benefits of Data Augmentation: Table I shows the effect of the background augmentation methods from Section IV-B on the performance of the 6D object pose estimator DOPE. For the object segmentation step, we use the *MaskFBA* method, which outperforms the simple *MaskThresholding* (see Section A.7 in [8]). Real-world images (*BgRandom*, *BgAlternate*, *BgBlend*) clearly outperform color noise (*BgNoise*) as a random background for augmentation. Moreover, it is beneficial to keep the original background for a portion (in our case 25%) of training images (*BgAlternate*, *BgBlend*) rather than to replace the background everywhere (*BgRandom*). The best results were achieved by *BgBlend*, which (after the random crop, horizontal flip, and segmentation by the *MaskFBA* method) blends the original background with a random image for 75% of the training images and keeps the original background for the remaining 25% of the training images. Compared to training without augmentation (*NoAug*), the use of *BgBlend* augmentation increased the ADD_5 accuracy more than twofold (from 29.2% to 60.1%). Thus, in all other experiments, we use the *BgBlend* background augmentation.

2) Generalization Across Camera Viewpoints: In this experiment, we study the robustness of the 6D object pose estimator with respect to the camera viewpoint. The *Imitrob* dataset contains one front camera (C1) and one right-hand side camera (C2). Table II compares results for the following scenarios: a) Same: training and testing on the same camera; b) Other: training on one camera and testing on the other; c) Both: training on both cameras. The accuracy of using a different camera viewpoint between training and testing was very low; on average, the results were better for the transfer from C1 to C2 ($\mathrm{ADD}_5 = 1.1\%$) than for the transfer from C2 to C1 (0.1%). The accuracy was significantly higher when the camera used for testing was included in the training. The best results for evaluation on C1 were achieved by models trained on both C1 and C2 (Both), while the best results for evaluation on C2 were achieved by models trained only on C2 (Same).

3) Generalization Across Left/Right Hand: We explore the generalization of the 6D pose estimator to manipulation of the tool by the left (LH) or right (RH) hand. Table II compares the results for the following cases: a) Same: training and testing on the same hand; b) Other: training on one hand and testing on the other; c) Both: training on both hands. While training and testing on the same hand (Same, $\mathrm{ADD}_5 = 48.7\%$) is clearly better than on the opposite hand (Other, 24.0%), using both LH and RH for training further improved the accuracy (Both, 58.8%). The data augmentation was more beneficial for training on both hands (Both, $\mathrm{ADD}_5^{\mathrm{ratio}} = 2.1\times$) than for training only on the same (Same, $1.9\times$) or opposite hand (Opposite, $1.5\times$). The ADD_5 and $\mathrm{ADD}_5^{\mathrm{ratio}}$ values in this paragraph are averages across LH and RH in the test set.

4) Generalization Across Demonstrators: To be transferable, the 6D pose estimation algorithm should be invariant to the subject that manipulates the tool. We examine the generalization of the DOPE estimator across 4 different subjects (S1-S4) using the following setups: a) AllToAll: train one model on all 4 subjects (i.e. train and test on S1-S4); b) ThreeToDiff: train one model on 3 subjects and test it on the remaining one (e.g. train on S1-S3 and test on S4); and c) OneToSame: train one model for each subject and test it on the same subject (e.g. train and test on S1). Table II averages the model accuracy for each setup across all test subjects (i.e. S1-S4). The AllToAll setup ($\mathrm{ADD}_5 = 58.8\%$) outperformed the ThreeToDiff setup (52.0%), which in turn clearly outperformed the OneToSame setup (31.1%). Additionally, the augmentation improves the accuracy more for ThreeToDiff ($\mathrm{ADD}_5^{\mathrm{ratio}} = 2.3\times$) than for the AllToAll ($2.0\times$) and OneToSame ($1.7\times$) setups.

5) Robustness to Clutter: To explore the generalization of the 6D object pose estimator to clutter in the test data, we compare its performance for the glue gun task frame with only the gluing frame on the table (*NoClutter*, default) and with a clutter of other

TABLE III
ROBUSTNESS TO CLUTTER (SEE SECTION V) FOR TOOL GLUE GUN AND TASK FRAME

ADD_5	*NoClutter*	*Clutter*
BgBlend	**61.8**	**61.5**
NoAug	22.8	4.9
ADD_5^{ratio}	2.7	12.6

TABLE IV
COMPARISON OF PERFORMANCE ON DIFFERENT TOOLS AND MANIPULATION TASKS

Tool	Task	ADD_5 (%)	E_{rot} (deg)	E_{tra} (cm)
glue gun	frame	53.3	11.8	5.0
	densewave	61.9	5.0	3.6
	sparsewave	66.0	5.0	3.4
	average	60.4	7.3	4.0
grout float	round	74.4	**3.9**	2.7
	sweep	**82.7**	4.3	**2.2**
	average	78.6	4.1	2.5
roller	press	50.5	8.7	3.7
glue gun 2	lshape	9.0	38.5	9.9
glue gun 3	lshape	4.7	40.3	10.2
glue gun 4	lshape	23.4	20.9	8.4
heat gun	heating	13.2	14.3	7.0
power drill	down	59.8	8.0	3.8
soldering iron	soldering	12.8	35.6	9.0
average	-	34.7	19.8	6.5

5 cm ADD pass rate (ADD_5) accuracy and average rotation (E_{rot}) and translation (E_{tra}) errors for different tools and tasks. invalid detections were excluded from the computation of E_{rot} and E_{tra}.

objects around the frame (*Clutter*) (see Table III). While the model trained without data augmentation (*NoAug*) was clearly worse on *Clutter* ($ADD_5 = 4.9\%$) than on *NoClutter* (22.8%), the use of data augmentation not only clearly improved the performance on both subsets but also increased the accuracy on *Clutter* (61.5%) to the same level as on *NoClutter* (61.8%). The ADD_5^{ratio} improvement ratio was $12.6\times$ for *Clutter*, compared with $2.7\times$ for *NoClutter*, indicating a big benefit of training with data augmentation for 6D object pose estimation in cluttered environment.

B. Final Results

1) Performance on Different Tools and Tasks: Table IV shows 5 cm ADD pass rates and rotation and translation errors for individual tools and tasks (see Fig. 3). The tested object pose estimator performed comparably on different tasks of the same tool. The best results were achieved for grout float ($ADD_5 = 78.6\%$, $E_{rot} = 4.1°$, and $E_{tra} = 2.5$ cm), while the most challenging tools included glue gun 2 and glue gun 3 (small size and large occlusions) and soldering iron (textureless glossy surface). Overall, the average 5 cm ADD pass rate was $ADD_5 = 34.7\%$ and the average rotation and translation errors were $E_{rot} = 19.8°$ and $E_{tra} = 6.5$ cm, respectively.

2) Qualitative Results: Fig. 6 presents example qualitative results of the 6D object pose estimator DOPE trained on the *ImitrobTrain* dataset using the best data augmentation (i.e. random crop and horizontal flip, segmentation by the *MaskFBA* method, and background randomization by the *BgBlend* method, see Section IV-B) and tested on the *ImitrobTest* set. The predicted bounding box is shown in red while the reference bounding box, acquired through the camera-to-tracker and tracker-to-tool calibration (see Section III), is green.

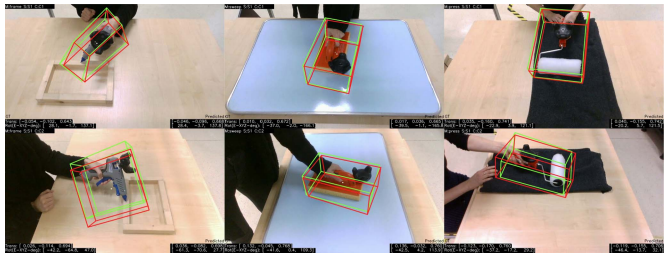


Fig. 6. Example qualitative results on the *ImitrobTest* dataset. Comparison of the reference bounding box (green) and the bounding box predicted by the 6D object pose estimator DOPE (red) trained on the *ImitrobTrain* dataset. More results are available in the supplementary video on the *Imitrob* project web page [6].

VI. CONCLUSION AND LESSONS LEARNED

In this letter, we address the problem of 6D pose estimation of hand tools manipulated by human demonstrators in an industrial environment from RGB image data. To investigate this problem, we have collected a challenging real-world benchmark video dataset (*Imitrob* dataset) of twelve manipulation tasks with nine different tools performed by four human demonstrators using left/right hand and recorded from two camera viewpoints (front and side).

We performed a broad range of experiments with various parts of the *ImitrobTrain* and *ImitrobTest* datasets using the object pose estimation method DOPE [2] to show the suitability of the *Imitrob* dataset for benchmarking 6D object pose estimation methods as well as to point out the limitations of current methods in this setup. The experiments imply that it is crucial to include in the training data the camera viewpoint used during the inference. Manipulation by the hand opposite to the side camera led to lower occlusion of the tool and higher accuracy of the pose estimator. The performance of models that used the same subject/camera/hand combination in both training and test data was often boosted by adding other demonstrators, camera viewpoint, or the other hand into the training set (see Table II).

To enhance the training data, we proposed several data augmentation methods that we provide together with the dataset. The best results were achieved by the background blending method *BgBlend* (see Table I), which increased the generalization capability of the trained models in all setups. To our knowledge, this is the first application of the background blending augmentation, previously used in image classification [39], [40], in the 6D object pose estimation domain.

The pose estimation accuracy correlated with the size and texture of the tool as well as with the performed task or clutter in the test environment (see Table IV). The best results ($ADD_5 = 78.6\%$, $E_{rot} = 4.1°$, $E_{tra} = 2.5$ cm) were achieved for grout float, which is large and moved along a plane, while the worst accuracy was observed for small tools with textureless surface and less restricted movement. Although the achieved accuracy of the evaluated method (DOPE) may not be sufficient for some industrial applications, the results are promising and show that the *Imitrob* dataset can be used to benchmark and select 6D object pose estimation methods for various tasks based on the required accuracy. We hope that the presented dataset will trigger

further development of 6D object pose estimation methods so that learning by demonstration using only visual information will soon become a reality.

ACKNOWLEDGMENT

This work was also supported by EU Horizon Europe Programme through the Project AGIMUS under Grant 101070165.

REFERENCES

[1] Y. Labbé, J. Carpentier, M. Aubry, and J. Sivic, "CosyPose: Consistent multi-view multi-object 6D pose estimation," in *Proc. IEEE Eur. Conf. Comput. Vis.*, 2020, pp. 574–591.

[2] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield, "Deep object pose estimation for semantic robotic grasping of household objects," *Proc. Conf. Robot Learn.*, vol. 87 pp. 306–316, 2018.

[3] S. Manschitz, J. Kober, M. Gienger, and J. Peters, "Learning movement primitive attractor goals and sequential skills from kinesthetic demonstrations," *Robot. Auton. Syst.*, vol. 74, pp. 97–107, 2015.

[4] B. Akgun, K. Subramanian, and A. L. Thomaz, "Novel interaction strategies for learning from teleoperation," *Proc. AAAI Fall Symp. Ser.*, vol. 12, p. 07, 2012.

[5] M. Ehrenmann, R. Zollner, S. Knoop, and R. Dillmann, "Sensor fusion approaches for observation of user actions in programming by demonstration," in *Proc. Conf. Documentation Int. Conf. Multisensor Fusion Integration Intell. Syst.*, 2001, pp. 227–232.

[6] CIIRC CTU in Prague, "Imitrob dataset version 2.0," 2023. [Online]. Available: http://imitrob.ciirc.cvut.cz/imitrobdataset.php

[7] CIIRC CTU in Prague, "GitHub for imitrob dataset version 2.0," 2023. [Online]. Available: https://github.com/imitrob/imitrob_dataset_code

[8] CIIRC CTU in Prague, "Imitrob dataset version 2.0, supplementary material," 2023. [Online]. Available: http://imitrob.ciirc.cvut.cz/imitrob_dataset/imitrob_supplement.pdf

[9] R. Skoviera et al., "Teaching robots to imitate a human with no on-teacher sensors. What are the key challenges?," in *Proc. IROS WS Towards Intell. Social Robots*, 2019. [Online]. Available: http://intelligent-social-robots-ws.com/previous-ws/materials-2018/

[10] C. Sahin, G. Garcia-Hernando, J. Sock, and T.-K. Kim, "A review on object pose recovery: From 3D bounding box detectors to full 6D pose estimators," *Image Vis. Comput.*, vol. 96, 2020, Art. no. 103898.

[11] M. E. Munich, P. Pirjanian, E. Di Bernardo, L. Goncalves, N. Karlsson, and D. Lowe, "SIFT-ing through features with ViPR," *IEEE Robot. Automat. Mag.*, vol. 13, no. 3, pp. 72–77, Sep. 2006.

[12] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proc. 9th Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.

[13] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "PoseCNN: A. convolutional neural network for 6D object pose estimation in cluttered scenes," in *Proc. Robot.: Sci. Syst.*, 2018. [Online]. Available: https://www.roboticsproceedings.org/rss14/index.html

[14] C. Wang et al., "DenseFusion: 6D object pose estimation by iterative dense fusion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3343–3352.

[15] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, "DeepIM: Deep iterative matching for 6D pose estimation," in *Proc. IEEE Eur. Conf. Comput. Vis.*, 2018, pp. 683–698.

[16] S. Zakharov, I. Shugurov, and S. Ilic, "DPOD: 6D pose object detector and refiner," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 1941–1950.

[17] K. Park, T. Patten, and M. Vincze, "Pix2Pose: Pixel-wise coordinate regression of objects for 6D pose estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7668–7677.

[18] G. Pitteri, S. Ilic, and V. Lepetit, "CorNet: Generic 3D corners for 6D pose estimation of new objects without retraining," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*, 2019, pp. 2807–2815.

[19] T. Hodan, D. Barath, and J. Matas, "EPOS: Estimating 6D pose of objects with symmetries," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11703–11712.

[20] Y. He, W. Sun, H. Huang, J. Liu, H. Fan, and J. Sun, "PVN3D: A. deep point-wise 3D keypoints voting network for 6DoF pose estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11632–11641.

[21] M. Rad and V. Lepetit, "BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2017, pp. 3828–3836.

[22] S. Hinterstoisser et al., "Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes," in *Proc. Asian Conf. Comput. Vis.*, 2012, pp. 548–562.

[23] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6D object pose estimation using 3D object coordinates," in *Proc. IEEE Eur. Conf. Comput. Vis.*, 2014, pp. 536–551.

[24] T. Hodan, P. Haluza, S. Obdrzalek, J. Matas, M. Lourakis, and X. Zabulis, "T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects," in *Proc. Winter Conf. Appl. Comput. Vis.*, 2017, pp. 536–551.

[25] B. Drost, M. Ulrich, P. Bergmann, P. Hartinger, and C. Steger, "Introducing MVTec ITODD - A dataset for 3D object recognition in industry," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*, 2017, pp. 2200–2208.

[26] T. Grenzdörffer, M. Günther, and J. Hertzberg, "YCB-M: A multi-camera RGB-D dataset for object recognition and 6DoF pose estimation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 3650–3656.

[27] P. Marion, P. R. Florence, L. Manuelli, and R. Tedrake, "Label Fusion: A pipeline for generating ground truth labels for real RGBD data of cluttered scenes," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 3235–3242.

[28] M. Kokic, D. Kragic, and J. Bohg, "Learning task-oriented grasping from human activity datasets," *IEEE Robot. Automat. Lett.*, vol. 5, no. 2, pp. 3352–3359, Apr. 2020.

[29] A. Krull, F. Michel, E. Brachmann, S. Gumhold, S. Ihrke, and C. Rother, "6-DoF model based tracking via object coordinate regression," in *Proc. Asian Conf. Comput. Vis.*, 2014, pp. 384–399.

[30] M. Quigley et al., "ROS: An open-source robot operating system," in *Proc. ICRA Workshop Open Source Softw.*, vol. 3 no. 3.2, p. 5, 2009.

[31] G. Bradski, "The OpenCV Library," *Dr Dobb's J. Softw. Tools*, vol. 25, no. 11, pp. 120–123, 2000.

[32] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," in *Proc. SIGGRAPH*, 1992, pp. 71–78.

[33] M. Dawson-Haggerty et al., "Trimesh [computer software]," 2019. [Online]. Available: https://trimsh.org/

[34] M. Borges, A. Symington, B. Coltin, T. Smith, and R. Ventura, "HTC Vive: Analysis and accuracy improvement," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 2610–2615.

[35] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The YCB Object and Model set: Towards common benchmarks for manipulation research," in *Proc. Int. Conf. Adv. Robot.*, 2015, pp. 510–517.

[36] M. Forte and F. Pitié, "F, B, Alpha Matting," 2020, *arXiv:2003.07711*.

[37] Z. Li, J. Sedlar, J. Carpentier, I. Laptev, N. Mansard, and J. Sivic, "Estimating 3D motion and forces of human–Object interactions from internet videos," *Int. J. Comput. Vis.*, vol. 130, no. 2, pp. 363–383, 2022.

[38] O. Vinyals et al., "Matching networks for one shot learning," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 3637–3645.

[39] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. Int. Conf. Learn. Representations*, 2018. [Online]. Available: https://openreview.net/group?id=ICLR.cc/2018/Conference#accepted-poster-papers

[40] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6023–6032.