BOP Challenge 2022 on Detection, Segmentation and Pose Estimation of Specific Rigid Objects

Martin Sundermeyer[1,2] Tomáš Hodaň[3] Yann Labbé[4] Gu Wang[5]
Eric Brachmann[6] Bertram Drost[7] Carsten Rother[8] Jiří Matas[9]

[1]Google [2]German Aerospace Center [3]Reality Labs at Meta [4]INRIA Paris [5]Tsinghua University
[6]Niantic [7]MVTec [8]Heidelberg University [9]Czech Technical University in Prague

Abstract

We present the evaluation methodology, datasets and results of the BOP Challenge 2022, the fourth in a series of public competitions organized with the goal to capture the status quo in the field of 6D object pose estimation from an RGB/RGB-D image. In 2022, we witnessed another significant improvement in the pose estimation accuracy – the state of the art, which was 56.9 AR_C in 2019 (Vidal et al.) and 69.8 AR_C in 2020 (CosyPose), moved to new heights of 83.7 AR_C (GDRNPP). Out of 49 pose estimation methods evaluated since 2019, the top 18 are from 2022. Methods based on point pair features, which were introduced in 2010 and achieved competitive results even in 2020, are now clearly outperformed by deep learning methods. The synthetic-to-real domain gap was again significantly reduced, with 82.7 AR_C achieved by GDRNPP trained only on synthetic images from BlenderProc. The fastest variant of GDRNPP reached 80.5 AR_C with an average time per image of 0.23s. Since most of the recent methods for 6D object pose estimation begin by detecting/segmenting objects, we also started evaluating 2D object detection and segmentation performance based on the COCO metrics. Compared to the Mask R-CNN results from CosyPose in 2020, detection improved from 60.3 to 77.3 AP_C and segmentation from 40.5 to 58.7 AP_C. The online evaluation system stays open and is available at: `bop.felk.cvut.cz`.

1. Introduction

Estimating the 6D pose, *i.e.*, the 3D translation and 3D rotation, of specific rigid objects from a single image is an important task for application fields such as robotic manipulation, augmented reality, or autonomous driving. The BOP Challenge 2022 is the fourth in a series of public challenges that are part of the BOP[1] project aiming to continuously report the state of the art in 6D object pose estimation. The first challenge was organized in 2017 [21] and the results were published in [20]. Results of the second challenge from 2019 [17], the third from 2020 [22], and the fourth from 2022 are included and discussed in this paper.

Participants of the 2022 challenge were competing on three tasks: 6D object localization, 2D object detection, and 2D object segmentation. The 6D object localization task has the same evaluation methodology and leaderboard since 2019, while the latter two tasks were introduced in 2022.

In the 6D object localization task, methods report their predictions on the basis of two sources of information. Firstly, at training time, a method is given 3D object models and training images showing the objects in known 6D poses. Secondly, at test time, the method is provided with a test image and a list of object instances visible in the image, and the goal is to estimate 6D poses of the listed instances. The images consist of RGB-D (aligned color and depth) channels and intrinsic camera parameters are known.

The 2D object detection and segmentation tasks were introduced to address the design of the majority of recent object pose estimation methods, which start by detecting/segmenting objects and then estimate their poses from the predicted image regions. Evaluating the detection/segmentation and pose estimation stages separately enables a better understanding of advances in the two stages. To create an opportunity for detector-agnostic comparison of pose estimation methods and to allow participants to focus only on the pose estimation stage, we also provided default detections and segmentations from Mask R-CNN [12] trained for CosyPose [29], the winning method in 2020.

The challenge primarily focuses on the practical scenario where no real images are available at training time, only the 3D object models and images synthesized using the models. While capturing real images of objects under various conditions and annotating the images with 6D object poses requires a significant human effort [18], the 3D models are either available before the physical objects, which is often

[1]BOP stands for Benchmark for 6D Object Pose Estimation [20].

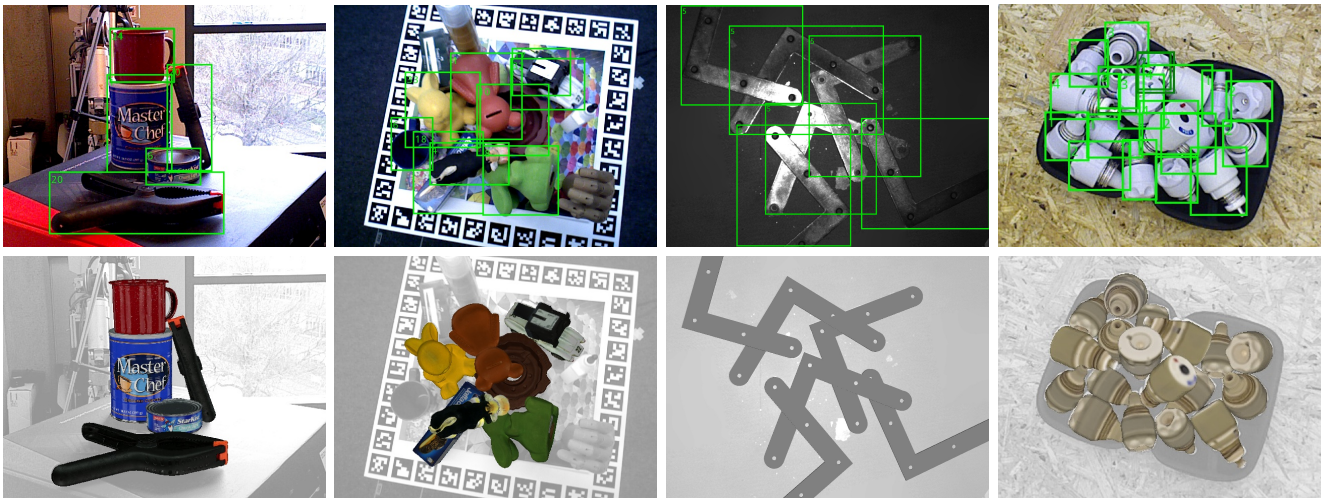

Figure 1. **2D object detection followed by 6D pose estimation from the detected regions** is a strategy used by the majority of recent 6D object pose estimation methods. This figure shows detections (top) and 3D object models rendered in estimated poses (bottom) produced by the 2022 top-performing method, GDRNPP [34, 51], on challenging images from YCB-V [55], HB [25], ITODD [6], and T-LESS [18].

the case for manufactured objects, or can be reconstructed at an admissible cost. Approaches for reconstructing 3D models of opaque, matte and moderately specular objects are established [37, 40] and promising approaches for transparent and highly specular objects are emerging [10, 36, 49, 53].

In the 2019 challenge, methods using the depth image channel were mostly based on point pair features (PPF's) [7] and clearly outperformed methods relying only on the RGB channels, all of which were based on deep neural networks (DNN's). DNN-based methods need large amounts of annotated training images, which had been typically obtained by OpenGL rendering of the 3D object models on random backgrounds [14,26]. However, as suggested in [23], the evident domain gap between these "render & paste" training images and real test images limits the potential of the DNN-based methods. To reduce the gap between the synthetic and real domains and thus to bring fresh air to the DNN world, we joined the development of BlenderProc[2] [2, 3], an open-source, physically-based renderer (PBR). For the 2020 challenge, we then provided participants with 350K PBR training images (see [22] for examples), which helped the DNN-based methods to achieve noticeably higher accuracy and to finally catch up with the PPF-based methods.

In the 2022 challenge, DNN-based methods for 6D object localization clearly outperformed PPF-based methods in both accuracy and speed, with the performance gains coming mostly from advances in network architectures and training schemes. The largest improvements were achieved on challenging industry-relevant datasets ITODD [6] and T-LESS [18], and on the HB dataset [25] which includes diverse objects captured under various levels of occlusion. Remarkably, RGB methods from 2022 surpassed RGB-D methods from 2020, the performance gap between methods trained only on PBR images and methods trained also on real images noticeably shrinked, and some methods started training on the depth image channel in addition to the RGB channels. On the new 2D object detection and segmentation tasks, large gains were achieved w.r.t. a baseline from 2020.

Sec. 2 of this paper defines the evaluation methodology, Sec. 3 introduces datasets, Sec. 4 describes the experimental setup and analyzes the results, Sec. 5 presents the awards of the BOP Challenge 2022, and Sec. 6 concludes the paper.

2. Evaluation Methodology

Methods are evaluated on the task of 6D object localization, as in 2019 and 2020 [22], and additionally on the tasks of 2D object detection and 2D object segmentation. The tasks are defined below together with accuracy scores that are used to compare methods. Participants could submit their results to any of the three tasks. Note that although all BOP datasets currently include RGB-D images (Sec. 3), a method may have used any of the image channels.

2.1. 2D Object Detection and Segmentation Tasks

Training input: At training time, a detection/segmentation method is provided a set of training images showing objects annotated with ground-truth 2D bounding boxes (for the detection task) and binary masks (for the segmentation task). The boxes are *amodal* (covering the whole object silhouette, including the occluded parts) while the masks are *modal* (covering only the visible object part). The method can also use 3D mesh models that are available for the objects (*e.g.*, to synthesize extra training images).

[2] github.com/DLR-RM/BlenderProc

Test input: At test time, the method is given an image showing an arbitrary number of instances of an arbitrary number of objects from a considered dataset. No prior information about the visible object instances is provided.

Test output: The method produces a list of amodal 2D bounding boxes (for detection) and modal binary masks (for segmentation) with confidences.

Metrics: Following the the evaluation methodology from the COCO 2020 Object Detection Challenge [31], the detection/segmentation accuracy is measured by the Average Precision (AP). Specifically, a per-object AP_O score is calculated by averaging the precision at multiple Intersection over Union (IoU) thresholds: $[0.5, 0.55, \ldots, 0.95]$. The accuracy of a method on a dataset D is measured by AP_D calculated by averaging per-object AP_O scores, and the overall accuracy on the core datasets (Sec. 3) is measured by AP_C defined as the average of the per-dataset AP_D scores.

Analagous to the 6D localization task, only instances for which at least 10% of the projected surface area is visible need to be detected/segmented. Correct predictions for objects that are visible from less than 10% are filtered out and not counted as false positives. Up to 100 predictions per image (with the highest confidences) are considered.

2.2. 6D Object Localization Task

As in the 2019 and 2020 editions of the challenge, methods are evaluated on the task of 6D localization of a **v**arying number of **i**nstances of a **v**arying number of **o**bjects from a single image. This variant of the 6D object localization task is referred to as ViVo and defined as follows.[3]

Training input: A method is provided a set of training images showing objects annotated with 6D poses, and 3D mesh models of the objects (typically with a color texture). A 6D pose is defined by a matrix $\mathbf{P} = [\mathbf{R} \,|\, \mathbf{t}]$, where $\mathbf{R}$ is a 3D rotation matrix, and $\mathbf{t}$ is a 3D translation vector. The matrix $\mathbf{P}$ defines a rigid transformation from the 3D space of the object model to the 3D space of the camera.

Test input: The method is given an image unseen during training and a list $L = [(o_1, n_1), \ldots, (o_m, n_m)]$, where n_i is the number of instances of object o_i visible in the image.

Test output: The method outputs a list $E = [E_1, \ldots, E_m]$, where E_i is a list of n_i pose estimates with confidences for instances of object o_i.

Metrics: The 6D object localization task is evaluated as in the 2020 challenge [22]. In short, the error of an estimated pose w.r.t. the ground-truth pose is calculated by three pose-error functions: Visible Surface Discrepancy (VSD) which treats indistinguishable poses as equivalent by considering only the visible object part, Maximum Symmetry-Aware Surface Distance (MSSD) which considers a set of pre-identified global object symmetries and measures the surface deviation in 3D, and Maximum Symmetry-Aware Projection Distance (MSPD) which considers the object symmetries and measures the perceivable deviation. An estimated pose is considered correct w.r.t. a pose-error function e, if $e < \theta_e$, where $e \in \{\text{VSD}, \text{MSSD}, \text{MSPD}\}$ and θ_e is the threshold of correctness. The fraction of annotated object instances for which a correct pose is estimated is referred to as Recall. The Average Recall w.r.t. a function e, denoted as AR_e, is defined as the average of the Recall rates calculated for multiple settings of the threshold θ_e and also for multiple settings of a misalignment tolerance τ in the case of VSD. The accuracy of a method on a dataset D is measured by: $\text{AR}_D = (\text{AR}_{\text{VSD}} + \text{AR}_{\text{MSSD}} + \text{AR}_{\text{MSPD}}) \,/\, 3$, which is calculated over estimated poses of all objects from D. The overall accuracy on the core datasets is measured by AR_C defined as the average of the per-dataset AR_D scores.[4]

3. Datasets

BOP currently includes twelve datasets in a unified format – sample test images are in Fig. 2 and dataset parameters in Tab. 1. Seven from the twelve were selected as core datasets: LM-O, T-LESS, ITODD, HB, YCB-V, TUD-L, IC-BIN. A method had to be evaluated on all core datasets to be considered for the main challenge awards (Sec. 5).

Each dataset includes 3D object models and training and test RGB-D images annotated with ground-truth 6D object poses. The object models are provided in the form of 3D meshes (in most cases with a color texture) which were created manually or using KinectFusion-like systems for 3D reconstruction [37]. While all test images are real, training images may be real and/or synthetic. The seven core datasets include a total of 350K photorealistic PBR (physically-based rendered) training images generated and automatically annotated with BlenderProc [2–4]. Example images, a description of the generation process and an analysis of the importance of PBR training images are in Sec. 3.2 and 4.3 of the 2020 challenge paper [22]. Datasets T-LESS, TUD-L and YCB-V include also real training images, and most datasets additionally include training images obtained by OpenGL rendering of the 3D object models on a black background. Test images were captured in scenes with graded complexity, often with clutter and occlusion. Datasets HB and ITODD include also real validation images – in this case, the ground-truth poses are publicly available only for the validation and not for the test images.

[3] See Sec. A.1 in [22] for a discussion on why the methods are evaluated on 6D object localization instead of 6D object detection, where no prior information about the visible object instances is provided [19].

[4] When calculating AR_C, scores are not averaged over objects before averaging over datasets, which is done when calculating AP_C (Sec. 2.1) to comply with the original COCO evaluation methodology [31].

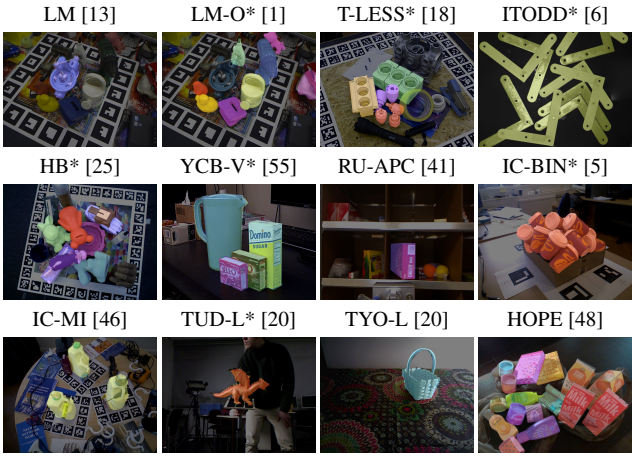

Figure 2. **An overview of the BOP datasets.** The seven core datasets are marked with a star. Shown are RGB channels of sample test images which were darkened and overlaid with colored 3D object models in the ground-truth 6D poses.

		Train. im.		Val im.	Test im.		Test inst.	
Dataset	Obj.	Real	PBR	Real	All	Used	All	Used
LM-O [1]	8	–	50K	–	1214	200	9038	1445
T-LESS [18]	30	37584	50K	–	10080	1000	67308	6423
ITODD [6]	28	–	50K	54	721	721	3041	3041
HB [25]	33	–	50K	4420	13000	300	67542	1630
YCB-V [55]	21	113198	50K	–	20738	900	98547	4123
TUD-L [20]	3	38288	50K	–	23914	600	23914	600
IC-BIN [5]	2	–	50K	–	177	150	2176	1786
LM [13]	15	–	50K	–	18273	3000	18273	3000
RU-APC [41]	14	–	–	–	5964	1380	5964	1380
IC-MI [46]	6	–	–	–	2067	300	5318	800
TYO-L [20]	21	–	–	–	1670	1670	1670	1670
HOPE [48]	28	–	–	50	188	188	3472	2898

Table 1. **Parameters of the BOP datasets.** The core datasets are listed in the upper part. PBR training images rendered by BlenderProc [2,3] are provided for all core datasets. Most datasets include also OpenGL-rendered training images of 3D object models on a black background (not shown in the table). If a dataset includes both validation and test images, ground-truth annotations are public only for the validation images. All test images are real. Column "Test inst./All" shows the number of annotated object instances for which at least 10% of the projected surface area is visible in the test image. Columns "Used" show the number of test images and object instances used in the BOP Challenge 2019, 2020, and 2022.

The datasets can be downloaded from the BOP website[5] and more details can be found in Chapter 7 of [15].

4. Results and Discussion

This section presents results of the BOP Challenge 2022, compares them with results from 2019 and 2020 challenge editions, and summarizes the main messages for our field.

[5] bop.felk.cvut.cz/datasets

In total, 49 methods were evaluated on the ViVo variant of the 6D object localization task on all seven core datasets – 11 methods in 2019, 15 in 2020, and 23 in 2022. Additionally, 8 methods were evaluated on the new detection task and 8 methods on the new segmentation task.

4.1. Experimental Setup

Participants of the BOP Challenge 2022 were submitting results of their methods to the online evaluation system at bop.felk.cvut.cz from May 1, 2022 until the deadline on October 16, 2022. The methods were evaluated on the ViVo variant of the 6D object localization task as described in Sec. 2.2 and on the 2D object detection and segmentation tasks as described in Sec. 2.1. The evaluation scripts are publicly available in the BOP toolkit.[6]

A method had to use a fixed set of hyper-parameters across all objects and datasets. For training, a method may have used the provided object models and training images, and rendered extra training images using the object models. However, not a single pixel of test images may have been used for training, nor the individual ground-truth poses or object masks provided for the test images. Ranges of the azimuth and elevation camera angles, and a range of the camera-object distances determined by the ground-truth poses from test images is the only information about the test set that may have been used during training.

Only subsets of test images were used to remove redundancies and speed up the evaluation, and only object instances for which at least 10% of the projected surface area is visible were considered in the evaluation.

4.2. 6D Object Localization Results

An overview of the 6D object localization results is in Tab. 2 and properties of the evaluated methods in Tab. 3. In 2022, all 23 of the new submissions rely on DNN's in their pipelines and 18 of them outperform CosyPose [29], the top-performing method from the 2020 challenge. The best method from 2022, GDRNPP [34, 51], is purely learning-based and achieves 83.7 AR_C, outperforming CosyPose by substantial 13.9 points in AR_C (#1−#19 in Tab. 2). Gains in accuracy are most notable on the industrial ITODD dataset [6] where GDRNPP reaches 67.9 AR_C (+36.6 AR_C w.r.t. CosyPose). This result is significant as ITODD reflects a challenging industrial scenario and was previously dominated by PPF-based approaches, the best of which, KoenigHybrid [27] (#24), achieved 48.3 AR_C.

GDRNPP dominates in 2022: The GDRNPP method was evaluated in seven variants, four of which are on top of the leaderboard. The variants were tailored towards different BOP 2022 awards (Sec. 5) by relying on different data domains and modalities and on different detection and pose re-

[6] github.com/thodan/bop_toolkit

#	Method	LM-O	T-LESS	TUD-L	IC-BIN	ITODD	HB	YCB-V	AR_C	Time
1	GDRNPP-PBRReal-RGBD-MModel [34, 51]	77.5	87.4	96.6	72.2	67.9	92.6	92.1	83.7	6.26
2	GDRNPP-PBR-RGBD-MModel [34, 51]	77.5	85.2	92.9	72.2	67.9	92.6	90.6	82.7	6.26
3	GDRNPP-PBRReal-RGBD-MModel-Fast [34, 51]	79.2	87.2	93.6	70.2	58.8	90.9	83.4	80.5	0.23
4	GDRNPP-PBRReal-RGBD-MModel-Offi. [34, 51]	75.8	82.4	96.6	70.8	54.3	89.0	89.6	79.8	6.41
5	Extended_FCOS+PFA-MixPBR-RGBD [24]	79.7	85.0	96.0	67.6	46.9	86.9	88.8	78.7	2.32
6	Extended_FCOS+PFA-MixPBR-RGBD-Fast [24]	79.2	77.9	95.8	67.1	46.0	86.0	88.0	77.1	0.64
7	RCVPose3D-SingleModel-VIVO-PBR [54]	72.9	70.8	96.6	73.3	53.6	86.3	84.3	76.8	1.34
8	ZebraPoseSAT-EffnetB4+ICP(DefaultDet) [43]	75.2	72.7	94.8	65.2	52.7	88.3	86.6	76.5	0.50
9	Extended_FCOS+PFA-PBR-RGBD [24]	79.7	80.2	89.3	67.6	46.9	86.9	82.6	76.2	2.63
10	SurfEmb-PBR-RGBD [11]	76.0	82.8	85.4	65.9	53.8	86.6	79.9	75.8	9.05
11	GDRNPP-PBRReal-RGBD-SModel [34, 51]	75.7	85.6	90.6	68.0	35.6	86.4	81.7	74.8	0.56
12	Coupled Iterative Refinement (CIR) [32]	73.4	77.6	96.8	67.6	38.1	75.7	89.3	74.1	–
13	GDRNPP-PBRReal-RGB-MModel [34, 51]	71.3	78.6	83.1	62.3	44.8	86.9	82.5	72.8	0.23
14	ZebraPoseSAT-EffnetB4 [43]	72.1	80.6	85.0	54.5	41.0	88.2	83.0	72.0	0.25
15	ZebraPoseSAT-EffnetB4(DefaultDet) [43]	70.7	76.8	84.9	59.7	41.7	88.7	81.6	72.0	0.25
16	ZebraPose-SAT [43]	72.1	78.7	86.1	54.9	37.9	84.7	82.8	71.0	–
17	Extended_FCOS+PFA-MixPBR-RGB [24]	74.5	77.8	83.9	60.0	35.3	84.1	80.6	70.9	3.02
18	GDRNPP-PBR-RGB-MModel [34, 51]	71.3	79.6	75.2	62.3	44.8	86.9	71.3	70.2	0.28
19	CosyPose-ECCV20-SYNT+REAL-ICP [29]	71.4	70.1	93.9	64.7	31.3	71.2	86.1	69.8	13.74
20	ZebraPoseSAT-EffnetB4 (PBR_Only) [43]	72.1	72.3	71.7	54.5	41.0	88.2	69.1	67.0	–
21	PFA-cosypose [24, 29]	71.4	73.8	83.7	59.6	24.6	71.2	80.7	66.4	–
22	Extended_FCOS+PFA-PBR-RGB [24]	74.5	71.9	73.2	60.0	35.3	84.1	64.8	66.3	3.50
23	SurfEmb-PBR-RGB [11]	66.3	73.5	71.5	58.8	41.3	79.1	64.7	65.0	8.89
24	Koenig-Hybrid-DL-PointPairs [27]	63.1	65.5	92.0	43.0	48.3	65.1	70.1	63.9	0.63
25	CosyPose-ECCV20-SYNT+REAL-1VIEW [29]	63.3	72.8	82.3	58.3	21.6	65.6	82.1	63.7	0.45
26	CRT-6D	66.0	64.4	78.9	53.7	20.8	60.3	75.2	59.9	0.06
27	Pix2Pose-BOP20_w/ICP-ICCV19 [38]	58.8	51.2	82.0	39.0	35.1	69.5	78.0	59.1	4.84
28	ZTE_PPF	66.3	37.4	90.4	39.6	47.0	73.5	50.2	57.8	0.90
29	CosyPose-ECCV20-PBR-1VIEW [29]	63.3	64.0	68.5	58.3	21.6	65.6	57.4	57.0	0.48
30	Vidal-Sensors18 [50]	58.2	53.8	87.6	39.3	43.5	70.6	45.0	56.9	3.22
31	CDPNv2_BOP20 (RGB-only & ICP) [30]	63.0	46.4	91.3	45.0	18.6	71.2	61.9	56.8	1.46
32	Drost-CVPR10-Edges [7]	51.5	50.0	85.1	36.8	57.0	67.1	37.5	55.0	87.57
33	CDPNv2_BOP20 (PBR-only & ICP) [30]	63.0	43.5	79.1	45.0	18.6	71.2	53.2	53.4	1.49
34	CDPNv2_BOP20 (RGB-only) [30]	62.4	47.8	77.2	47.3	10.2	72.2	53.2	52.9	0.94
35	Drost-CVPR10-3D-Edges [7]	46.9	40.4	85.2	37.3	46.2	62.3	31.6	50.0	80.06
36	Drost-CVPR10-3D-Only [7]	52.7	44.4	77.5	38.8	31.6	61.5	34.4	48.7	7.70
37	CDPN_BOP19 (RGB-only) [30]	56.9	49.0	76.9	32.7	6.7	67.2	45.7	47.9	0.48
38	CDPNv2_BOP20 (PBR-only & RGB-only) [30]	62.4	40.7	58.8	47.3	10.2	72.2	39.0	47.2	0.98
39	leaping from 2D to 6D [33]	52.5	40.3	75.1	34.2	7.7	65.8	54.3	47.1	0.43
40	EPOS-BOP20-PBR [16]	54.7	46.7	55.8	36.3	18.6	58.0	49.9	45.7	1.87
41	Drost-CVPR10-3D-Only-Faster [7]	49.2	40.5	69.6	37.7	27.4	60.3	33.0	45.4	1.38
42	Félix&Neves-ICRA2017-IET2019 [39, 42]	39.4	21.2	85.1	32.3	6.9	52.9	51.0	41.2	55.78
43	Sundermeyer-IJCV19+ICP [45]	23.7	48.7	61.4	28.1	15.8	50.6	50.5	39.8	0.86
44	Zhigang-CDPN-ICCV19 [30]	37.4	12.4	75.7	25.7	7.0	47.0	42.2	35.3	0.51
45	PointVoteNet2 [9]	65.3	0.4	67.3	26.4	0.1	55.6	30.8	35.1	–
46	Pix2Pose-BOP20-ICCV19 [38]	36.3	34.4	42.0	22.6	13.4	44.6	45.7	34.2	1.22
47	Sundermeyer-IJCV19 [45]	14.6	30.4	40.1	21.7	10.1	34.6	44.6	28.0	0.20
48	SingleMultiPathEncoder-CVPR20 [44]	21.7	31.0	33.4	17.5	6.7	29.3	28.9	24.1	0.19
49	DPOD (synthetic) [57]	16.9	8.1	24.2	13.0	0.0	28.6	22.2	16.1	0.23

Table 2. **6D object localization results on the seven core datasets.** The methods are ranked by the AR_C score which is the average of the per-dataset AR_D scores defined in Sec. 2.2. The last column shows the average image processing time (in seconds).

finement methods. Having results of these variants enables to understand the importance of individual aspects of the pipeline. The common ground is the Geometrically-Guided Direct Regression Network (GDR-Net) [51], which takes an RGB object crop as input and densely predicts 2D-3D correspondences, identities of surface fragments [16], and a mask of the visible object part. Then, instead of applying P*n*P-RANSAC [16], the predictions are concatenated and fed into a small CNN with a fully connected head that regresses a scale-invariant translation [30] and a 3D rotation using the allocentric 6D representation [28]. The 3D rotation loss takes into account object symmetries that are provided in the BOP datasets. For BOP 2022, GDR-Net [51] was modified by exchanging the ResNet34 backbone with ConvNext [35], predicting both modal and amodal masks as intermediate representations, and applying stronger domain randomization. The winning GDRNPP variant trains YOLOX [8] for object detection and GDR-Net for pose es-

#	Method	Year	Type	DNN per	Det./seg.	Refinement	Train im.	...type	Test im.
1	GDRNPP-PBRReal-RGBD-MModel [34, 51]	2022	DNN	Object	YOLOX	∼CIR	RGB-D	PBR+real	RGB-D
2	GDRNPP-PBR-RGBD-MModel [34, 51]	2022	DNN	Object	YOLOX	∼CIR	RGB-D	PBR	RGB-D
3	GDRNPP-PBRReal-RGBD-MModel-Fast [34, 51]	2022	DNN	Object	YOLOX	Depth adjust.	RGB	PBR+real	RGB-D
4	GDRNPP-PBRReal-RGBD-MModel-Offi. [34, 51]	2022	DNN	Object	Default (synt+real)	∼CIR	RGB-D	PBR+real	RGB-D
5	Extended_FCOS+PFA-MixPBR-RGBD [24]	2022	DNN	Dataset	Extended FCOS	PFA	RGB	PBR+real	RGB-D
6	Extended_FCOS+PFA-MixPBR-RGBD-Fast [24]	2022	DNN	Dataset	Extended FCOS	PFA	RGB	PBR+real	RGB-D
7	RCVPose3D-SingleModel-VIVO-PBR [54]	2022	DNN	Dataset	RCVPose3D	ICP	RGB-D	PBR+real	RGB-D
8	ZebraPoseSAT-EffnetB4+ICP(DefaultDet) [43]	2022	DNN	Object	Default (synt+real)	ICP	RGB	PBR+real	RGB-D
9	Extended_FCOS+PFA-PBR-RGBD [24]	2022	DNN	Dataset	Extended FCOS	PFA	RGB	PBR	RGB-D
10	SurfEmb-PBR-RGBD [11]	2022	DNN	Dataset	Default (PBR)	Custom	RGB-D	PBR	RGB-D
11	GDRNPP-PBRReal-RGBD-SModel [34, 51]	2022	DNN	Dataset	YOLOX	Depth adjust.	RGB	PBR+real	RGB-D
12	Coupled Iterative Refinement (CIR) [32]	2022	DNN	Dataset	Default (synt+real)	CIR	RGB-D	PBR+real	RGB-D
13	GDRNPP-PBRReal-RGB-MModel [34, 51]	2022	DNN	Object	YOLOX	–	RGB	PBR+real	RGB
14	ZebraPoseSAT-EffnetB4 [43]	2022	DNN	Object	FCOS	–	RGB	PBR+real	RGB
15	ZebraPoseSAT-EffnetB4(DefaultDet) [43]	2022	DNN	Object	Default (synt+real)	–	RGB	PBR+real	RGB
16	ZebraPose-SAT [43]	2022	DNN	Object	FCOS	–	RGB	PBR+real	RGB
17	Extended_FCOS+PFA-MixPBR-RGB [24]	2022	DNN	Dataset	Extended FCOS	PFA	RGB	PBR+real	RGB
18	GDRNPP-PBR-RGB-MModel [34, 51]	2022	DNN	Object	YOLOX	–	RGB	PBR	RGB
19	CosyPose-ECCV20-SYNT+REAL-ICP [29]	2020	DNN	Dataset	Default (synt+real)	DeepIM+ICP	RGB	PBR+real	RGB-D
20	ZebraPoseSAT-EffnetB4 (PBR_Only) [43]	2022	DNN	Object	FCOS	–	RGB	PBR	RGB
21	PFA-cosypose [24, 29]	2022	DNN	Dataset	MaskRCNN	PFA	RGB-D	PBR+real	RGB
22	Extended_FCOS+PFA-PBR-RGB [24]	2022	DNN	Dataset	Extended FCOS	PFA	RGB	PBR	RGB
23	SurfEmb-PBR-RGB [11]	2022	DNN	Dataset	Default (PBR)	Custom	RGB	PBR	RGB
24	Koenig-Hybrid-DL-PointPairs [27]	2020	DNN/PPF	Dataset	Retina/MaskRCNN	ICP	RGB	Synt+real	RGB-D
25	CosyPose-ECCV20-SYNT+REAL-1VIEW [29]	2020	DNN	Dataset	Default (synt+real)	∼DeepIM	RGB	PBR+real	RGB
26	CRT-6D	2022	DNN	Dataset	Default (synt+real)	Custom	RGB	PBR+real	RGB
27	Pix2Pose-BOP20_w/ICP-ICCV19 [38]	2020	DNN	Object	MaskRCNN	ICP	RGB	PBR+real	RGB-D
28	ZTE_PPF	2022	DNN/PPF	Dataset	Default (synt+real)	ICP	RGB	PBR+real	RGB-D
29	CosyPose-ECCV20-PBR-1VIEW [29]	2020	DNN	Dataset	Default (PBR)	∼DeepIM	RGB	PBR	RGB
30	Vidal-Sensors18 [50]	2019	PPF	–	–	ICP	–	–	D
31	CDPNv2_BOP20 (RGB-only & ICP) [30]	2020	DNN	Object	FCOS	ICP	RGB	Synt+real	RGB-D
32	Drost-CVPR10-Edges [7]	2019	PPF	–	–	ICP	–	–	RGB-D
33	CDPNv2_BOP20 (PBR-only & ICP) [30]	2020	DNN	Object	FCOS	ICP	RGB	PBR	RGB-D
34	CDPNv2_BOP20 (RGB-only) [30]	2020	DNN	Object	FCOS	–	RGB	Synt+real	RGB
35	Drost-CVPR10-3D-Edges [7]	2019	PPF	–	–	ICP	–	–	D
36	Drost-CVPR10-3D-Only [7]	2019	PPF	–	–	ICP	–	–	D
37	CDPN_BOP19 (RGB-only) [30]	2020	DNN	Object	RetinaNet	–	RGB	Synt+real	RGB
38	CDPNv2_BOP20 (PBR-only & RGB-only) [30]	2020	DNN	Object	FCOS	–	RGB	PBR	RGB
39	leaping from 2D to 6D [33]	2020	DNN	Object	Unknown	–	RGB	Synt+real	RGB
40	EPOS-BOP20-PBR [16]	2020	DNN	Dataset	–	–	RGB	PBR	RGB
41	Drost-CVPR10-3D-Only-Faster [7]	2019	PPF	–	–	ICP	–	–	D
42	Félix&Neves-ICRA2017-IET2019 [39, 42]	2019	DNN/PPF	Dataset	MaskRCNN	ICP	RGB-D	Synt+real	RGB-D
43	Sundermeyer-IJCV19+ICP [45]	2019	DNN	Object	RetinaNet	ICP	RGB	Synt+real	RGB-D
44	Zhigang-CDPN-ICCV19 [30]	2019	DNN	Object	RetinaNet	–	RGB	Synt+real	RGB
45	PointVoteNet2 [9]	2020	DNN	Object	–	ICP	RGB-D	PBR	RGB-D
46	Pix2Pose-BOP20-ICCV19 [38]	2020	DNN	Object	MaskRCNN	–	RGB	PBR+real	RGB
47	Sundermeyer-IJCV19 [45]	2019	DNN	Object	RetinaNet	–	RGB	Synt+real	RGB
48	SingleMultiPathEncoder-CVPR20 [44]	2020	DNN	All	MaskRCNN	–	RGB	Synt+real	RGB
49	DPOD (synthetic) [57]	2019	DNN	Dataset	–	–	RGB	Synt	RGB

Table 3. **Properties of evaluated 6D object localization methods.** Column *Year* is the year of submission, *Type* indicates whether the method relies on deep neural networks (DNN's) or point pair features (PPF's), *DNN per..* shows how many DNN models were trained, *Det./seg.* is the object detection or segmentation method, *Refinement* is the pose refinement method, *Train im.* and *Test im.* show image channels used at training and test time respectively, and *Train im. type* is the domain of training images. All test images are real.

timation on the provided PBR and real RGB images, and refines the poses by a multi-hypotheses refinement method inspired by Coupled Iterative Refinement (CIR) [32], which is trained on PBR and real RGB-D images.

Training on depth: Methods RCVPose3D [54] (#7) and CIR [32] (#12; a variant is also used in #1, 2, 4), started benefiting from learning on the depth channel in addition to the RGB channels (only PointVoteNet2 [9] applied a neural network to the depth channel in 2020). On the flip side, the multi-hypotheses refinement methods can be time-intensive – the CIR-based approach increases the inference time of GDRNPP by 6.03s per image on average (#1−#3).

Increased accuracy & speed: The third GDRNPP entry replaces the CIR-based refinement [32], which is used in the

top two entries, by a fast and simple depth-based adjustment of the 3D translation and still achieves impressive 80.5 AR_C in just 0.23s per image. In comparison, the best method in 2020 that took less than 1s per image is KoenigHybrid [27] (#24) with 63.9 AR_C and 0.63s per image.

RGB-only from 2022 beats RGB-D from 2020: The best method that relies only on RGB image channels at both training and test time is a variant of GDRNPP (#13). Without any pose refinement, this method achieves 72.8 AR_C which is +9.1 w.r.t. CosyPose that applies RGB-based pose refinement (#25) and +3.0 w.r.t. to the overall best method from 2020, *i.e.*, CosyPose with a depth-based ICP (#19).

Synthetic-to-real gap shrinks further: Another important result was achieved by the GDRNPP variant that is trained only on the provided synthetic PBR images rendered with BlenderProc [2, 3]. With 82.7 AR_C, this variant achieves the second highest accuracy. On datasets with real training images (T-LESS, YCB-V, TUD-L), the synthetically trained variant is only -2.5 AR_C on average behind the winning method that was trained on both PBR and real training images. In the RGB-only setting, the synthetic-to-real gap has been reduced on the three datasets from Δ15.8 AR_C (observed on CosyPose in 2020; #25−#29) to Δ6.2 AR_C (observed on GDRNPP in 2022; #13−#18). The BOP 2020 results [22] demonstrated the importance of training on PBR images over training on rasterized images with random backgrounds. The BOP 2022 results confirm this observation and also suggest that the synthetic-to-real gap monotonically shrinks as the accuracy of methods increases (see, *e.g.*, #25−#29, #14−#20, #5−#9, #1−#2 in Tab. 2).

Scalability in the number of objects: The advancement in the synthetic-to-real transfer is crucial for increasing the scope of applications. In addition, real world applications require methods whose computational and memory resources scale gracefully with the amount of target objects. The top four GDRNPP variants are all trained with at least one pose network per object. This means that the training and inference time complexity and the inference memory increase linearly with the number of target objects. When GDRNPP is trained with one pose network per BOP dataset containing 2–33 objects (Tab. 1), it achieves only 74.8 AR_C (#11) and is outperformed by, *e.g.*, Extended_FCOS+PFA [24] (#5) that reaches 78.7 AR_C with one pose network per dataset. This raises the question how the results would change if [24] was trained per object.

2D detection followed by 6D pose estimation: Almost all 6D object localization methods evaluated in 2022 start by detecting the object instances in RGB images by predicting their 2D bounding boxes. Some methods also predict 2D object masks in the detected regions at training time for loss calculation [24] or extra supervision [12], and some predict 2D masks at both training and inference time and use them to establish correspondences [11,43]. The only exception is RCVPose3D [54], which does not start by detecting object instance in the RGB image channels and instead segments the object instances in 3D point clouds calculated from the depth image channel.

Detector-agnostic results: Eleven methods use the default 2D object detections (*Default* in column *Det./seg.* in Tab. 3), which were provided to participants of the 2022 challenge and produced by Mask R-CNN [12] trained for the first stage of CosyPose [29] in 2020. Three of these methods use detections from Mask R-CNN trained only on PBR images, and eight use detections from Mask R-CNN trained on synthetic and real images (where the synthetic include PBR and additional images synthesized by the authors of [29]). Among the eight methods, GDRNPP is once again at the top with 79.8 AR_C (#4). We can therefore conclude that the pose estimation performance of the GDRNPP pipeline is performing best independent of the used detection method. However, the accuracy gap to other methods decreases with the default detections, *e.g.*, from +7.2 AR_C (#1−#8) to +3.3 AR_C (#4−#8) w.r.t. ZebraPose [43].

4.3. 2D Object Detection Results

As shown in Tab. 4, the YOLOX [8] detector from GDRNPP has the top performance of 77.3 AP_C. This detector employs a ConvNext [35] backbone and was trained with the Ranger optimizer [52] and strong data augmentation. Mask R-CNN [12] from CosyPose only achieves 60.5 AP_C (-16.8 AP_C), which explains the +3.9 AR_C gain in the pose accuracy (#1−#4 in Tab. 2). YOLOX is relatively insensitive to the image domain, improving only +3.5 AP_C (#1−#2 in Tab. 4) when trained also on real images. Mask R-CNN yields +4.8 AP_C (#6−#7) and FCOS [47] yields +5.4 AP_C (#3−#4) in such a comparison.

Although all 2D object detection methods rely only on RGB and ignore the depth channel, they work remarkably well even on the texture-less objects from T-LESS [18] (see the BOP website for per-dataset scores). However, detections from YOLOX on YCB-V [55] in Fig. 1 reveal a limitation of the RGB-only detection that fails to distinguish the two differently sized clamps. This detection failure can cause wrong pose estimates even though the rendered scene seems perfectly plausible. Depth data could help to disambiguate the object scale in such cases.

4.4. 2D Object Segmentation Results

We see an improvement from 40.5 AP_C achieved by the default masks from Mask R-CNN to 58.7 AP_C achieved by masks from ZebraPoseSAT [43] (+18.2 AP_C; #1−#7 in Tab. 5). Interestingly, ZebraPoseSAT predicts the high-quality masks in regions determined by the default detections from Mask R-CNN (#6 in Tab. 4) and would likely

#	Method	...based on	Year	Data	...type	AP_C	Time
1	GDRNPPDet	YOLOX	2022	RGB	PBR+real	77.3	.081
2	GDRNPPDet	YOLOX	2022	RGB	PBR	73.8	.081
3	Extended_FCOS	FCOS	2022	RGB	PBR+real	72.1	.030
4	Extended_FCOS	FCOS	2022	RGB	PBR	66.7	.030
5	DLZDet	DLZDet	2022	RGB	PBR	65.6	-
6	CosyPose	Mask R-CNN	2020	RGB	PBR+real	60.5	.054
7	CosyPose	Mask R-CNN	2020	RGB	PBR	55.7	.055
8	FCOS-CDPN	FCOS	2022	RGB	PBR	50.7	.047

Table 4. **2D object detection results.** The methods are ranked by the AP_C score defined in Sec. 2.1. The last column shows the average image processing time (in seconds).

#	Method	...based on	Year	Data	...type	AP_C	Time
1	ZebraPoseSAT	CosyPose+Zebra	2022	RGB	PBR+real	58.7	.080
2	ZebraPoseSAT	CDPNv2+Zebra	2022	RGB	PBR+real	57.8	.080
3	ZebraPoseSAT	CosyPose+Zebra	2022	RGB	PBR	53.8	.080
4	ZebraPoseSAT	CDPNv2+Zebra	2022	RGB	PBR	52.3	.080
5	DLZDet	DLZDet	2022	RGB	PBR+real	49.6	-
6	DLZDet	DLZDet	2022	RGB	PBR	42.9	-
7	CosyPose	Mask R-CNN	2020	RGB	PBR+real	40.5	.054
8	CosyPose	Mask R-CNN	2020	RGB	PBR	36.2	.055

Table 5. **2D object segmentation results.** Details as in Tab. 4.

achieve even higher segmentation accuracy if relying on detections from YOLOX trained for GDRNPP. As mentioned in Sec. 4.2, most 6D object localization methods evaluated in 2022 start by 2D object detection. Leveraging 2D object segmentation instead could improve results on objects with irregular shapes [56] which are included, *e.g.*, in the industrial ITODD dataset [6].

5. Awards

The following BOP Challenge 2022 awards were presented at the 7th Workshop on Recovering 6D Object Pose[7] organized at the ECCV 2022 conference. The awards are based on the 6D object localization results in Tab. 2, method properties in Tab. 3, the 2D object detection results in Tab. 4, and the 2D object segmentation results in Tab. 5.

The *GDRNPP* [34, 51] submissions were prepared by Xingyu Liu, Ruida Zhang, Chenyangguang Zhang, Bowen Fu, Jiwen Tang, Xiquan Liang, Jingyi Tang, Xiaotian Cheng, Yukang Zhang, Gu Wang, Xiangyang Ji; *Extended_FCOS+PFA* [24] by Yang Hai, Rui Song, Zhiqiang Liu, Jiaojiao Li, Mathieu Salzmann, Pascal Fua, Yinlin Hu; *ZebraPoseSAT* [43] by Yongzhi Su, Praveen Nathan, Torben Fetzer, Jason Rambach, Didier Stricker, Mahdi Saleh, Yan Di, Nassir Navab, Benjamin Busam, Federico Tombari, Yongliang Lin, Yu Zhang, *Coupled Iterative Refinement* [32] by Lahav Lipson, Zachary Teed, Ankit Goyal, and Jia Deng; and *RCVPose3D* [54] by Yangzheng Wu, Alireza Javaheri, Mohsen Zand, Michael Greenspan.

[7] cmp.felk.cvut.cz/sixd/workshop_2022

Awards for 6D object localization methods:

- **The Overall Best Method:** *GDRNPP-PBRReal-RGBD-MModel*
- **The Best RGB-Only Method:** *GDRNPP-PBRReal-RGB-MModel*
- **The Best Fast Method (less than 1s per image):** *GDRNPP-PBRReal-RGBD-MModel-Fast*
- **The Best BlenderProc-Trained Method:** *GDRNPP-PBR-RGBD-MModel*
- **The Best Single-Model Method** (trained per dataset)**:** *Extended_FCOS+PFA-MixPBR-RGBD*
- **The Best Open-Source Method:** *GDRNPP-PBRReal-RGBD-MModel*
- **The Best Method On Default Detections/Segment.:** *GDRNPP-PBRReal-RGBD-MModel-OfficialDet*
- **The Best Method on T-LESS, ITODD, YCB-V, HB:** *GDRNPP-PBRReal-RGBD-MModel*
- **The Best Method on LM-O:** *Extended_FCOS+PFA-MixPBR-RGBD*
- **The Best Method on TUD-L:** *Coupled Iterative Refinement (CIR)*
- **The Best Method on IC-BIN:** *RCVPose3D_SingleModel_VIVO_PBR*

Awards for 2D object detection/segmentation methods:

- **The Overall Best Detection Method:** *GDRNPPDet_PBRReal*
- **The Best BlenderProc-Trained Detection Method:** *GDRNPPDet_PBR*
- **The Overall Best Segmentation Method:** *ZebraPoseSAT-EffnetB4 (DefaultDetection)*
- **The Best BlenderProc-Trained Segment. Method:** *ZebraPoseSAT-EffnetB4 (DefaultDet+PBR_Only)*

6. Conclusions

In the BOP Challenge 2022, we witnessed another breakthrough in the 6D pose estimation accuracy, efficiency and synthetic-to-real transfer. Methods based on deep neural networks now clearly surpass the traditional methods based on point pair features in both accuracy and speed. Variations of the winning GDRNPP method [34, 51] allowed us to analyze the importance of different aspects related to training domains, modalities and run-time efficiency. Besides, we individually measured 2D detection and segmentation performance and could thereby determine sources of gains in the multi-stage pose estimation pipelines. Despite the progress, accuracy scores have not been saturated on most BOP datasets and we are already looking forward to insights from the next challenge. The online evaluation system at `bop.felk.cvut.cz` stays open and raw results of all methods will be made publicly available.

References

[1] Eric Brachmann, Alexander Krull, Frank Michel, Stefan Gumhold, Jamie Shotton, and Carsten Rother. Learning 6D object pose estimation using 3D object coordinates. *ECCV*, 2014. 4

[2] Maximilian Denninger, Martin Sundermeyer, Dominik Winkelbauer, Dmitry Olefir, Tomáš Hodaň, Youssef Zidan, Mohamad Elbadrawy, Markus Knauer, Harinandan Katam, and Ahsan Lodhi. BlenderProc: Reducing the reality gap with photorealistic rendering. *RSS Workshops*, 2020. 2, 3, 4, 7

[3] Maximilian Denninger, Martin Sundermeyer, Dominik Winkelbauer, Youssef Zidan, Dmitry Olefir, Mohamad Elbadrawy, Ahsan Lodhi, and Harinandan Katam. BlenderProc. *arXiv preprint arXiv:1911.01911*, 2019. 2, 3, 4, 7

[4] Maximilian Denninger, Dominik Winkelbauer, Martin Sundermeyer, Wout Boerdijk, Markus Wendelin Knauer, Klaus H Strobl, Matthias Humt, and Rudolph Triebel. Blenderproc2: A procedural pipeline for photorealistic rendering. *Journal of Open Source Software*, 8(82):4901, 2023. 3

[5] Andreas Doumanoglou, Rigas Kouskouridas, Sotiris Malassiotis, and Tae-Kyun Kim. Recovering 6D object pose and predicting next-best-view in the crowd. *CVPR*, 2016. 4

[6] Bertram Drost, Markus Ulrich, Paul Bergmann, Philipp Hartinger, and Carsten Steger. Introducing MVTec ITODD – A dataset for 3D object recognition in industry. *ICCVW*, 2017. 2, 4, 8

[7] Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3D object recognition. *CVPR*, 2010. 2, 5, 6

[8] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. YOLOX: Exceeding YOLO series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 5, 7

[9] Frederik Hagelskjær and Anders Glent Buch. PointPoseNet: Accurate object detection and 6 DoF pose estimation in point clouds. *arXiv preprint arXiv:1912.09057*, 2019. 5, 6

[10] Jon Hasselgren, Nikolai Hofmann, and Jacob Munkberg. Shape, light & material decomposition from images using monte carlo rendering and denoising. *NeurIPS*, 2022. 2

[11] Rasmus Laurvig Haugaard and Anders Glent Buch. SurfEmb: Dense and continuous correspondence distributions for object pose estimation with learnt surface embeddings. *CVPR*, 2022. 5, 6, 7

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. *ICCV*, 2017. 1, 7

[13] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab. Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. *ACCV*, 2012. 4

[14] Stefan Hinterstoisser, Vincent Lepetit, Paul Wohlhart, and Kurt Konolige. On pre-trained image features and synthetic images for deep learning. *ECCVW*, 2018. 2

[15] Tomáš Hodaň. Pose estimation of specific rigid objects. *PhD Thesis, Czech Technical University in Prague*, 2021. 4

[16] Tomáš Hodaň, Dániel Baráth, and Jiří Matas. EPOS: Estimating 6D pose of objects with symmetries. *CVPR*, 2020. 5, 6

[17] Tomáš Hodaň, Eric Brachmann, Bertram Drost, Frank Michel, Martin Sundermeyer, Jiří Matas, and Carsten Rother. BOP Challenge 2019. `https://bop.felk.cvut.cz/media/bop_challenge_2019_results.pdf`, 2019. 1

[18] Tomáš Hodaň, Pavel Haluza, Štěpán Obdržálek, Jiří Matas, Manolis Lourakis, and Xenophon Zabulis. T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects. *WACV*, 2017. 1, 2, 4, 7

[19] Tomáš Hodaň, Jiří Matas, and Štěpán Obdržálek. On evaluation of 6D object pose estimation. *ECCVW*, 2016. 3

[20] Tomáš Hodaň, Frank Michel, Eric Brachmann, Wadim Kehl, Anders Glent Buch, Dirk Kraft, Bertram Drost, Joel Vidal, Stephan Ihrke, Xenophon Zabulis, Caner Sahin, Fabian Manhardt, Federico Tombari, Tae-Kyun Kim, Jiří Matas, and Carsten Rother. BOP: Benchmark for 6D object pose estimation. *ECCV*, 2018. 1, 4

[21] Tomáš Hodaň, Frank Michel, Caner Sahin, Tae-Kyun Kim, Jiří Matas, and Carsten Rother. SIXD Challenge 2017. `http://cmp.felk.cvut.cz/sixd/challenge_2017/`, 2017. 1

[22] Tomáš Hodaň, Martin Sundermeyer, Bertram Drost, Yann Labbé, Eric Brachmann, Frank Michel, Carsten Rother, and Jiří Matas. BOP Challenge 2020 on 6D object localization. *ECCV*, 2020. 1, 2, 3, 7

[23] Tomáš Hodaň, Vibhav Vineet, Ran Gal, Emanuel Shalev, Jon Hanzelka, Treb Connell, Pedro Urbina, Sudipta Sinha, and Brian Guenter. Photorealistic image synthesis for object instance detection. *ICIP*, 2019. 2

[24] Yinlin Hu, Pascal Fua, and Mathieu Salzmann. Perspective flow aggregation for data-limited 6d object pose estimation. *arXiv preprint arXiv:2203.09836*, 2022. 5, 6, 7, 8

[25] Roman Kaskman, Sergey Zakharov, Ivan Shugurov, and Slobodan Ilic. HomebrewedDB: RGB-D dataset for 6D pose estimation of 3D objects. *ICCVW*, 2019. 2, 4

[26] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. SSD-6D: Making RGB-based 3D detection and 6D pose estimation great again. *ICCV*, 2017. 2

[27] Rebecca Koenig and Bertram Drost. A hybrid approach for 6dof pose estimation. *ECCVW*, 2020. 4, 5, 6, 7

[28] Abhijit Kundu, Yin Li, and James M Rehg. 3D-RCNN: Instance-level 3D object reconstruction via render-and-compare. *CVPR*, 2018. 5

[29] Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. CosyPose: Consistent multi-view multi-object 6D pose estimation. *ECCV*, 2020. 1, 4, 5, 6, 7

[30] Zhigang Li, Gu Wang, and Xiangyang Ji. CDPN: Coordinates-based disentangled pose network for real-time rgb-based 6-dof object pose estimation. *ICCV*, 2019. 5, 6

[31] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. *ECCV*, 2014. 3

[32] Lahav Lipson, Zachary Teed, Ankit Goyal, and Jia Deng. Coupled iterative refinement for 6d multi-object pose estimation. In *CVPR*, 2022. 5, 6, 8

[33] Jinhui Liu, Zhikang Zou, Xiaoqing Ye, Xiao Tan, Errui Ding, Feng Xu, and Xin Yu. Leaping from 2D detection to efficient 6DoF object pose estimation. *ECCVW*, 2020. 5, 6

[34] Xingyu Liu, Ruida Zhang, Chenyangguang Zhang, Bowen Fu, Jiwen Tang, Xiquan Liang, Jingyi Tang, Xiaotian Cheng, Yukang Zhang, Gu Wang, and Xiangyang Ji. GDRNPP. `https://github.com/shanice-l/gdrnpp_bop2022`, 2022. 2, 4, 5, 6, 8

[35] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. *CVPR*, 2022. 5, 7

[36] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting triangular 3d models, materials, and lighting from images. *CVPR*, 2022. 2

[37] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. *ISMAR*, 2011. 2, 3

[38] Kiru Park, Timothy Patten, and Markus Vincze. Pix2Pose: Pixel-wise coordinate regression of objects for 6D pose estimation. *ICCV*, 2019. 5, 6

[39] Carolina Raposo and Joao P Barreto. Using 2 point+normal sets for fast registration of point clouds with small overlap. *ICRA*, 2017. 5, 6

[40] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3D: Large-scale learning and evaluation of real-life 3D category reconstruction. In *ICCV*, 2021. 2

[41] Colin Rennie, Rahul Shome, Kostas E Bekris, and Alberto F De Souza. A dataset for improved RGBD-based object detection and pose estimation for warehouse pick-and-place. *RA-L*, 2016. 4

[42] Pedro Rodrigues, Michel Antunes, Carolina Raposo, Pedro Marques, Fernando Fonseca, and Joao Barreto. Deep segmentation leverages geometric pose estimation in computer-aided total knee arthroplasty. *Healthcare Technology Letters*, 2019. 5, 6

[43] Yongzhi Su, Mahdi Saleh, Torben Fetzer, Jason Rambach, Nassir Navab, Benjamin Busam, Didier Stricker, and Federico Tombari. ZebraPose: Coarse to fine surface encoding for 6DoF object pose estimation. *CVPR*, 2022. 5, 6, 7, 8

[44] Martin Sundermeyer, Maximilian Durner, En Yen Puang, Zoltan-Csaba Marton, Narunas Vaskevicius, Kai O Arras, and Rudolph Triebel. Multi-path learning for object pose estimation across domains. *CVPR*, 2020. 5, 6

[45] Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, and Rudolph Triebel. Augmented Autoencoders: Implicit 3D orientation learning for 6D object detection. *IJCV*, 2019. 5, 6

[46] Alykhan Tejani, Danhang Tang, Rigas Kouskouridas, and Tae-Kyun Kim. Latent-class hough forests for 3D object detection and pose estimation. *ECCV*, 2014. 4

[47] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: Fully convolutional one-stage object detection. In *ICCV*, 2019. 7

[48] Stephen Tyree, Jonathan Tremblay, Thang To, Jia Cheng, Terry Mosier, Jeffrey Smith, and Stan Birchfield. 6-DoF pose estimation of household objects for robotic manipulation: An accessible dataset and benchmark. *IROS*, 2022. 4

[49] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. In *CVPR*, 2022. 2

[50] Joel Vidal, Chyi-Yeu Lin, Xavier Lladó, and Robert Martí. A method for 6D pose estimation of free-form rigid objects using point pair features on range data. *Sensors*, 2018. 5, 6

[51] Gu Wang, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. GDR-Net: Geometry-guided direct regression network for monocular 6D object pose estimation. *CVPR*, 2021. 2, 4, 5, 6, 8

[52] Less Wright. Ranger: A synergistic optimizer. `https://github.com/lessw2020/Ranger-Deep-Learning-Optimizer`, 2019. 7

[53] Bojian Wu, Yang Zhou, Yiming Qian, Minglun Cong, and Hui Huang. Full 3D reconstruction of transparent objects. *ACM TOG*, 2018. 2

[54] Yangzheng Wu, Alireza Javaheri, Mohsen Zand, and Michael Greenspan. Keypoint cascade voting for point cloud based 6DoF pose estimation. *arXiv preprint arXiv:2210.08123*, 2022. 5, 6, 7, 8

[55] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. *RSS*, 2018. 2, 4, 7

[56] Lei Yang, Yan Zi Wei, Yisheng He, Wei Sun, Zhenhang Huang, Haibin Huang, and Haoqiang Fan. iShape: A first step towards irregular shape instance segmentation. *arXiv preprint arXiv:2109.15068*, 2021. 8

[57] Sergey Zakharov, Ivan Shugurov, and Slobodan Ilic. DPOD: 6D pose object detector and refiner. *ICCV*, 2019. 5, 6