



HAL
open science

FruitBin: a large-scale fruit bin picking dataset tunable over occlusion, camera pose and scenes for 6D pose estimation

Guillaume Duret, Mahmoud Ali, Nicolas Cazin, Alexandre Chapin, Florence Zara, Emmanuel Dellandréa, Jan Peters, Liming Chen

► To cite this version:

Guillaume Duret, Mahmoud Ali, Nicolas Cazin, Alexandre Chapin, Florence Zara, et al.. FruitBin: a large-scale fruit bin picking dataset tunable over occlusion, camera pose and scenes for 6D pose estimation. 2023. hal-04122072v1

HAL Id: hal-04122072

<https://hal.science/hal-04122072v1>

Preprint submitted on 8 Jun 2023 (v1), last revised 9 Jun 2023 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

FruitBin: a large-scale fruit bin picking dataset tunable over occlusion, camera pose and scenes for 6D pose estimation

Guillaume Duret^{1,3}
guillaume.duret@ec-lyon.fr

Mahmoud Ali¹
mahmoud.ali@ec-lyon.fr

Nicolas Cazin¹
nicolas.cazin@ec-lyon.fr

Alexandre Chapin¹
alexandre.chapin@ec-lyon.fr

Florence Zara²
florence.zara@liris.cnrs.fr

Emmanuel Dellandrea¹
emmanuel.dellandrea@ec-lyon.fr

Jan Peters³
jan.peters@tu-darmstadt.de

Liming Chen¹
liming.chen@ec-lyon.fr

¹Univ Lyon, Centrale Lyon, CNRS, INSA Lyon, UCBL, LIRIS, UMR5205, F-69130 Ecully, France

²Univ Lyon, UCBL, CNRS, INSA Lyon, LIRIS, UMR5205, F-69622 Villeurbanne, France

³Intelligent Autonomous Systems Lab, Technical University of Darmstadt, 64289 Darmstadt, Germany

Abstract

1 Bin picking is a widely spread application in industries and its automation through
2 robots generally requires object instance-level segmentation and 6D pose estimation.
3 State-of-the-art computer vision algorithms for these tasks are deep learning-based
4 and require large datasets of diversified annotated images at the instance level,
5 which are prohibitively expensive to acquire. In this paper, we make use of
6 PickSim, a newly developed Gazebo-based dynamically configurable open-source
7 pipeline, and introduce a dataset of simulated data, namely FruitBin, for the
8 challenging task of fruit bin picking. FruitBin contains more than 1M images
9 and 40M instance-level 6D pose annotations over both symmetric and asymmetric
10 fruits with or without texture. Rich annotations and metadata (including 6D pose,
11 segmentation mask, point cloud, 2D and 3D bounding boxes, occlusion rate) allow
12 the tuning of the proposed dataset for benchmarking the robustness of object
13 instance segmentation and 6D pose estimation models (with respect to variations in
14 terms of lighting, texture, occlusion, camera pose and scenes). We further propose
15 three scenarios presenting significant challenges of 6D pose estimation models:
16 new scene generalization; new camera viewpoint generalization; and occlusion
17 robustness. We show the results of these three scenarios for two 6D pose estimation
18 baselines making use of RGB or RGBD images. To the best of our knowledge,
19 FruitBin is the first dataset for the challenging task of fruit bin picking and the
20 biggest large-scale dataset for 6D pose estimation with the most comprehensive
21 challenges, tunable over scenes, camera poses and occlusions.



Figure 1: Example of 6 different points of view of a single scene of the FruitBin dataset

22 1 Introduction

23 Bin picking refers to the process of extracting objects from a bin or container. It is commonly
 24 used in various industries, such as manufacturing, logistics, and warehouse. Its automation through
 25 robotic systems involves using sensors, computer vision, and robotic arms to identify and retrieve
 26 objects from an unstructured and cluttered environment. State-of-the-art solutions are data-driven and
 27 imply robot perception to segment object instances and estimate their 6D pose (16; 10). As a result,
 28 they generally require large-scale datasets of diversified object instance-level annotations which are
 29 prohibitively expensive to acquire.

30 State-of-the-art benchmarks for 6D pose estimation are computer vision oriented. Therefore, they are
 31 not defined within robotic learning software or provide only a partial robotic environment, thereby
 32 hindering the latter stage of seamless robot learning for manipulation which further involves learning
 33 interactions between robots and objects (5). Furthermore, almost all of them only depict tabletop
 34 scenes with rigid objects and don't consider bin picking scenarios where objects typically occur in
 35 multiple instances and feature severe occlusions and clutter.

36 In this paper, we present FruitBin, a large-scale dataset of simulated data suitable for robot learning
 37 and specifically designed for the challenging task of fruit bin picking as example can be seen in the
 38 figures 1 or 3. And Table 1 compares FruitBin with the state of the art. It builds upon PickSim (7),
 39 a recently released open-source bin picking simulation pipeline that offers dynamic configuration
 40 capabilities within Gazebo (18), an open-source, 3D robotics simulation software widely used in
 41 the field of robotics research and development. FruitBin considers delicate objects, *e.g.*, apricot,
 42 banana, whose manipulation requires learning the appropriate force through haptic feedback and
 43 paves the way to robotic manipulation of deformable objects. It comprises over 1 million images
 44 and 40 million instance-level 6D pose annotations, encompassing both symmetric and asymmetric
 45 fruits with and without texture. The dataset includes comprehensive annotations and metadata,
 46 such as 6D pose, segmentation masks, point clouds, 2D and 3D bounding boxes, and occlusion
 47 rates. This rich set of annotations allows fine-tuning the proposed dataset in order to benchmark
 48 the robustness of object instance segmentation and 6D pose estimation models against variations
 49 in lighting, texture, occlusion, camera pose, and scene complexity. Furthermore, we propose three
 50 scenarios that present significant challenges for 6D pose estimation models: new scene generalization;
 51 new camera viewpoint generalization; and occlusion robustness. We evaluated the performance of
 52 two 6D pose estimation baselines, using either RGB or RGBD images, on these three scenarios.
 53 To the best of our knowledge, FruitBin is the first dataset specifically tailored for the demanding
 54 task of fruit bin picking and represents the largest-scale dataset for 6D pose estimation, offering
 55 comprehensive challenges that can be adjusted in terms of scenes, camera poses, and occlusions.

56 In summary, our contributions are as follows:

- 57 • release of FruitBin: the biggest large-scale dataset for fruit bin picking with 40M 6D pose
58 annotations over 1M images, depicting comprehensive challenges for 6D pose estimation;
- 59 • rich annotations for object instance-level segmentation and 6D pose estimation, including
60 2D and 3D bounding boxes, instance, segmentation masks, and occlusion rates;
- 61 • proposition of three benchmark scenarios to test robustness of 6D pose estimation models in
62 terms of occlusion, camera pose, and scene variety.
- 63 • baseline results over these three scenarios for two 6D pose estimation baselines making use
64 of RGB and RGB-D images, respectively.

65 In the following, Section 2 present previous work in 6D pose datasets. Section 3 presents the
66 generation of FruitBin with the software PickSim. Section 4 describes FruitBin with the specific
67 statistics of the large dataset and sub-datasets for the 3 scenarios. Section 5 gathers the result of
68 baselines of 6D estimation models through the 3 scenarios. Section 6 presents some limitations, and
69 Section 7 concludes this article with some perspectives.

70 2 Related work

71 Because data famine is the major roadblock for learning-based computer vision tasks and even
72 the more for robot learning. State-of-the-art datasets for 6D pose estimation (6; 11) is quickly
73 over-viewed in this section.

74 **6D pose datasets.** State of the art has featured a number of benchmarks for 6D pose estimation.
75 Table 1 lists these datasets and compares them in terms of different characteristics, including data
76 nature (real or synthetic), size (number of samples, number of scenes, number of 6D pose annotations),
77 as well as challenges, *e.g.*, occlusion, clutter, multiple instances, *etc.*

78 The proposed FruitBin dataset offers high-resolution images and Table 1 demonstrates that it distin-
79 guishes itself by incorporating all of the challenges into a single dataset. Notably, it stands out among
80 the limited number of datasets that have significantly increased the sample size, ranging from 2 to
81 1000 times more samples than existing 6D pose estimation datasets. This expansion in dataset size is
82 crucial as it addresses a critical challenge faced by deep learning models—generalizing to unknown
83 scenes. By incorporating a larger number of diverse scenes, we believe FruitBin enables models to
84 overcome this limitation. With over **40 million** 6D pose annotations, FruitBin not only surpasses
85 other datasets in terms of the number of scenes but also provides the largest number of annotations
86 available.

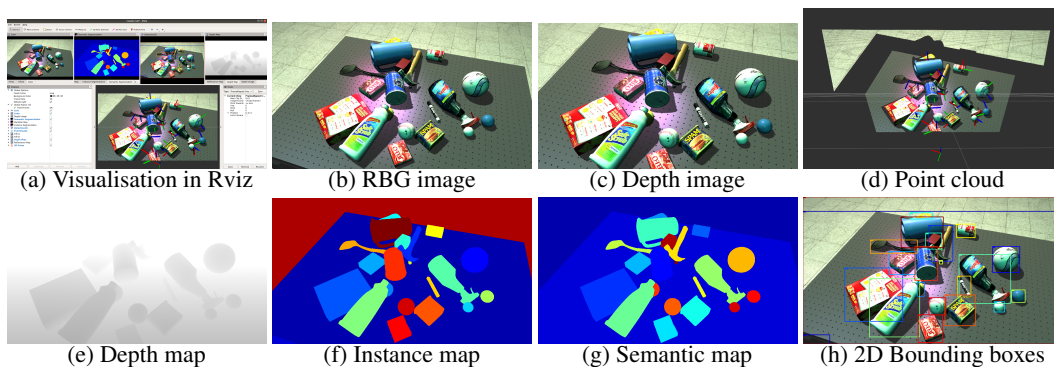


Figure 2: Examples of annotations

87 3 Raw data generation process of FruitBin using PickSim

88 This section introduces the generation of Fruitbin through the use of PickSim (7). This recent pipeline
89 offers extensive annotation generation capabilities as illustrated in the figure 2 and facilitates the

Dataset	Data	Labels	#Samples	#Scenes	#6D pose	Challenges
LINEMOD (12)	Real	Manual	18k	15	15k	C+TL
O-LINEMOD (1)	Real	Semi-auto	1214	15	120k	C+TL+O
APC (26)	Real	Semi-auto	10k	12	~240k	C+L
T-LESS (13)	Real	Semi-auto	49k	20	47k	C+TL+O+MI
YCB-V (29)	Real-S.	Semi-auto	133k	92	613k	O+C
FAT (24)	Synth.	Auto	60k	3	205k	C
BIN-P (17)	Real-S.	Semi-auto	206k	12	20M	SC+SO+MI+BP
CAMERA (28)	Real-S.	Semi-auto	308k	31	4M	O+C
ObjectSynth (14)	Synth.	Auto	600k	6	21M	O+C
HomebrewedDB (15)	Synth.	Auto	17.4k	13	56k	O+C+L
GraspNet-1B (3)	Real	Semi-Auto	97k	190	970k	O+C
RobotP (30)	Synth.	Semi-Auto	4k	-	-	O+TL+L
HOPE (25)	Real	Manual	2k	50	~30k	O+C+MI+L
MetaGraspNet (8)	Real-S.	Auto	217k	6.4k	3M	O+C+MI
SynPick (23)	Synth.	Auto	503k	300	10M	O
StereOBJ-1M (19)	Real	Semi-Auto	396k	183	1.5M	O+L
DoPose (9)	Real	Semi-Auto	3k	301	11k	O+C+BP
FruitBin	Synth.	Auto	1M	70k	40M	SO+SC+MI+BP+TL+L

Table 1: Comparison of 6D pose datasets with their different challenges (O: Occlusion, C: Clutter, SO: Severe Occlusion, SC: Severe Clutter, MI: Multiple Instance, BP: Bin Pickings, TL: Texture Less, L: Light).

90 utilization of the dataset for robotics learning. In order to create task-specific datasets for robotics, it
91 is crucial to develop and train tasks such as vision and manipulation within the same environment.
92 Utilizing robotic software, such as Gazebo, for generating computer vision synthetic data offers
93 numerous advantages. Firstly, it seamlessly integrates physical engines, resulting in more realistic
94 outcomes. Secondly, it facilitates the effortless integration of robots and sensors with native robot
95 control capabilities. Lastly, it unlocks the potential for creating datasets specifically tailored to
96 robotic tasks with various open-source libraries like MoveIt (4). Additionally, PickSim offers easy-to-
97 use setup files for domain randomization, dataset recording, and generation. Each step within the
98 pipeline can be executed effortlessly using a simple command, with parameters conveniently set in
99 user-friendly JSON or YAML files. In this section, we present the generation of FruitBin using the
100 PickSim (7) pipeline into four key steps.

101 **Pre-processing.** Raw meshes can be provided to enable mesh domain randomization, encompassing
102 textures, object properties, and more. For FruitBin, PickSim receives eight raw meshes representing
103 the fruits we aim to simulate, namely Apple, Apricot, Banana, Kiwi, Lemon, Orange, Peach, and Pear.
104 No randomization is applied to the mesh or textures in order to preserve the distinct characteristics of
105 each fruit. Through this automated process, sdf files are generated, which are essential for robotic
106 software simulation and contain key metadata such as category ID for the future dataset recording.

107 **Scene randomization.** PickSim (7) enables domain randomization (2; 20; 21). This functionality
108 is used to generate diverse scenes for fruit bin picking. Through configuration files, users can
109 easily customize the number of objects, cameras, and lighting conditions without writing additional
110 code, simplifying the creation of randomized Gazebo world files. In the FruitBin dataset, scene
111 randomization encompasses the bin, fruits, and lighting. The bin undergoes randomization with
112 rotations and color variations, while the lighting setup includes randomized positions, intensities, and
113 colors. This deliberate setup ensures significant lighting diversity. To maintain statistical consistency,
114 the number of instances for each category is randomly set between 0 and 30, ensuring a consistently
115 full bin. Examples of these scenes can be seen in Figure 3.

116 **Camera randomization.** The final aspect of randomization involves the camera settings, where
117 we employ an orbiter sampler included in PickSim to introduce randomness in the distance (ranging
118 from 0.55m to 1m) between the camera and the orbiter center, as well as different angles to ensure



Figure 3: 30 first scenes for a unique point of view

119 optimal viewpoints of the scene. This streamlined setup enables the generation of fully randomized
 120 scenes that are both physically realistic and ideal for fruit bin-picking scenarios. The impact of these
 121 camera parameters can be observed in Figure 1, showcasing five different viewpoints of a scene.

122 **Record data.** Simulations in Gazebo can be seamlessly launched using the generated world files.
 123 These simulations produce datasets with recorded annotations, including instance and semantic
 124 segmentation, bounding boxes, occlusion rates, 6D pose estimations, depth maps, point clouds, and
 125 normals.

126 4 FruitBin: a large scale dataset

127 4.1 Data and Statistics

128 For FruitBin with eight fruits, the randomization process is performed 10,000 times with 15 cameras,
 129 resulting in 150,000 data frames. This process is repeated seven times. Assembling the 7 parts
 130 represent Fruibin with over 1 million frames across 70,000 scenes and 105 camera viewpoints. The
 131 dataset is organized systematically, preserving information in metafiles. By dividing the dataset
 132 into seven parts, sub-datasets can be created for scene generalization, camera generalization, and
 133 occlusion robustness. A comprehensive dataset comparison can be found in Section 2.

134 To streamline the organization of data for learning 6D pose estimation, we have restructured the
 135 data based on features and categories. FruitBin incorporates metadata to store essential information
 136 such as scene ID, camera ID, category ID, and instance IDs. These unique IDs enable direct access
 137 to instance or category-based annotations such as masks, 2D/3D bounding boxes, 6D poses, and
 138 occlusion rates. This occlusion rate offers a notable advantage by enabling visible instance counting
 139 in images and facilitating data filtering based on different occlusion rates, as described in Section 4.
 140 Figure 4 depicts data statistics and provides insights into the distribution of 6D poses among different
 141 fruit categories. Notably, the complete randomization of scenes results in a Gaussian distribution of
 142 instance numbers in the images, and ensures a balanced representation of each fruit category.

143 4.2 A tunable large-scale dataset for fruit bin picking

144 Developing large-scale, diverse, and accurately annotated datasets for 6D pose estimation is a
 145 demanding and time-consuming task. The availability of comprehensive and well-annotated datasets
 146 plays a crucial role in advancing the state-of-the-art in 6D pose estimation. However, each dataset, as
 147 outlined in Table 1, presents unique challenges that necessitate specific datasets. These challenges
 148 include bin picking, scene diversity, point of view diversity, occlusion, and multi-instances, among
 149 others. Given that 6D pose estimation finds applications in various contexts, it becomes necessary
 150 to fine-tune different 6D models to cater to specific requirements. The suitability of FruitBin as a
 151 tunable dataset stems from its abundant annotations and extensive scale. Specifically, the large-scale
 152 dataset allows generating sub-datasets tailored to specific purposes or ablation studies. The tunable

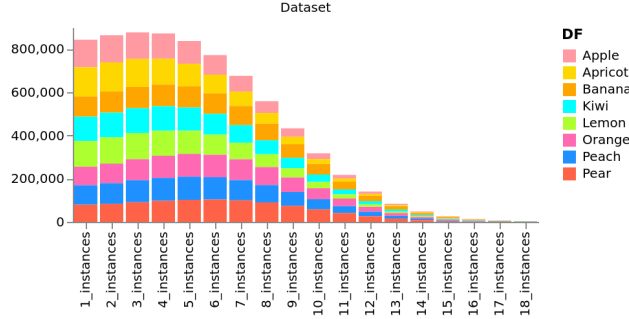


Figure 4: Statistics of the whole dataset with the number of instances for each category in the frames

153 capability of FruitBin is exemplified through its application in addressing three primary challenges
 154 in 6D pose estimation: scene generalization, camera point-of-view generalization, and occlusion
 155 robustness.

156 **Camera and scene generalization scenarios.** To explore scene generalization and point-of-view
 157 camera generalization, 2 scenarios of single-instance 6D pose estimation are created. The idea is
 158 to sample the FruitBin dataset to generate scenario oriented dataset. The sampling is from the first
 159 part of the dataset consisting of 10,000 distinct scenes and 15 identical camera points of view. In the
 160 scene scenario, data is sampled from the large dataset, resulting in 60% representing 6000 scenes
 161 with all 15 cameras for training, 20% for evaluation, and 20% for testing, with each part containing a
 162 different scene. A similar approach is applied to study the camera scenario, with nine initial points of
 163 view allocated for training, three for evaluation, and the last three for testing. During the filtering
 164 of the dataset, all the dataset samples are category based to already process the data for the purpose
 165 of future 6D pose estimation. During this sample, new metadata is recorded such as all the scenes,
 166 camera IDs, and all the category-based instance ID corresponding to including their occlusion rate.

167 **Occlusion robustness scenarios.** For the precise study of occlusion robustness, we considered
 168 studying as a parameter of the two previous. In practice for the two previous scenarios, 4 levels
 169 of difficulty are generated leveraging occlusion rate. During the previous sampling, an occlusion
 170 parameter is added; instead of taking all data, a filter will be applied based on the occlusion rate of
 171 the object. The first version focuses on objects with occlusion bellow 30%, followed by versions with
 172 occlusion going to 50%, 70%, and 90%, respectively, representing the most challenging scenarios.
 173 Additional splitting based on occlusion is performed during testing to evaluate occlusion-related
 174 performances. For an even more precise study of the impact of the occlusion, the testing part could
 175 be split according to a partition for getting an occlusion-aware performance overview as presented in
 176 the tables 3.

177 The figure 5 present the splitting of the different scenario for the 4 occlusion difficulties.

178 5 Experiments

179 **Baseline Methods** In order to estimate the challenges posed by our FruitBin dataset, we conducted
 180 a comprehensive assessment using two distinct state-of-the-art 6D pose estimation models employing
 181 different data modalities.

182 The first method, **PVNet** (22), utilizes a RGB image and information from 3D models of the objects
 183 as input to extract the 6D pose. This method processes the input image in two stages: firstly, it
 184 determines 2D keypoints locations of the objects using a series of convolution and deconvolution
 185 blocks, followed by a RANSAC-based voting scheme. The 6D pose is then obtained by solving an
 186 uncertainty-driven Perspective-n-Point (PnP) problem given the 2D key points and the 3D model.

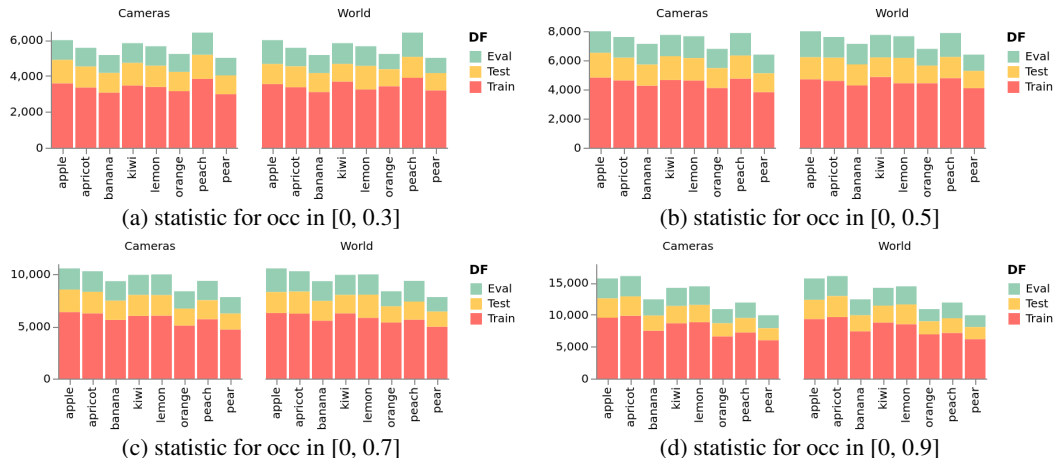


Figure 5: Statistics of the 4 occlusion ranges and two scenarios (scene (world) and camera) with the splitting: training, evaluating and testing.

187 The second method, **Densefusion** (27), takes as input RGB image and depth information (RGB-D
 188 input) along with a semantic mask of the scene. Generally, depth information methods tend to be
 189 more robust but require more computational resources. DenseFusion initially extracts a binary mask
 190 for each object, which is then used to crop the image and the point cloud within the region of interest
 191 (ROI). Each ROI is then used as input to a 2D feature extractor and a point cloud extractor to obtain
 192 color and geometry embeddings. These embeddings are then concatenated for each point and fused
 193 to extract 'local' and 'global' information, resulting in the final (dense) fused features. The 6D pose
 194 is estimated using a pose predictor model that iteratively refines the pose at each step.

195 Both models have been trained and evaluated on existing 6D pose estimation datasets: LINEMOD
 196 and YCB-Video, presented in section 2, and have shown state-of-the-art results.

197 **Metrics** The baseline models are evaluated using the ADD metric (12) (*average distance*) for
 198 non-symmetrical objects and ADD-S (29) (*average closest point distance*) for symmetrical objects.
 199 In the case of FruitBin, Apple, Apricot, Kiwi, Lemon, Orange and Peach objects are considered as
 200 symmetrical while Banana and Pear are non-symmetrical. In the following, ADD(-S) refer to both
 201 metrics.

202 ADD is the mean distance of the transformed 3d model points using the estimated pose $[\hat{R}|\hat{t}]$ and
 203 ground truth pose $[R|t]$. Based on the computed distance, the estimated pose is considered correct if
 204 the distance is less than 10% of the model's diameter. The diameter represents the longest distance
 205 between 2 points in the object.

206 **Experiments** Using the two baselines described in 5 and the metric presented in 5, we train each of
 207 them over the 8 scenarios presented in section 4.2. The result are shown in table 2 for Densefusion
 208 and PVNet. For instance, in these tables, "Scenes_occ[0.3,0.4]" refers to the "Scene" scenario with a
 209 level of occlusion ranging from 30% to 40%.

210 Densefusion, which relies on depth information, exhibits superior performance compared to PVNet,
 211 which solely utilizes RGB images. However, the effectiveness of Densefusion is heavily influenced
 212 by object occlusion. It achieves a success rate of 90% or higher when the object's occlusion is below
 213 30%. Conversely, as the occlusion in the data increases, up to 90%, the performance of Densefusion
 214 declines, reaching a success rate of 67%. It fails to meet the refinement threshold for this level of
 215 occlusion for the camera scenario. Additionally, it is important to note that Densefusion employs
 216 ground truth segmentation as an input for specific studies on 6D pose estimation. This observation
 217 is further substantiated by the results presented in Table 3, which illustrate the performance across
 218 various occlusion ranges. The performance remains satisfactory when objects are slightly occluded,

	Apple	Apricot	Banana	Kiwi	Lemon	Orange	Peach	Pear	Avg
Densefusion									
Scenes_occ[0.0,0.3]	0.997	0.993	0.490	0.991	0.996	1.0	1.0	0.674	0.899
Scenes_occ[0.0,0.5]	0.995	0.950	0.526	0.948	0.956	1.0	1.0	0.636	0.882
Scenes_occ[0.0,0.7]	0.981	0.950	0.414	0.894	0.933	0.997	0.998	0.570	0.849
Scenes_occ[0.0,0.9]	0.844	0.713	0.278	0.656	0.726	0.896	0.903	0.306	0.676
Cameras_occ[0.0,0.3]	0.983	0.872	0.588	0.968	0.957	1.0	0.999	0.669	0.888
Cameras_occ[0.0,0.5]	0.978	0.900	0.592	0.974	0.980	0.999	0.999	0.606	0.887
Cameras_occ[0.0,0.7]	0.983	0.922	0.530	0.887	0.864	0.995	0.997	0.553	0.850
Pvnet									
Scenes_occ[0.0,0.3]	0.505	0.422	0.858	0.501	0.486	0.572	0.640	0.762	0.593
Scenes_occ[0.0,0.5]	0.430	0.432	0.880	0.503	0.473	0.572	0.685	0.793	0.596
Scenes_occ[0.0,0.7]	0.533	0.431	0.879	0.475	0.492	0.581	0.649	0.763	0.600
Scenes_occ[0.0,0.9]	0.445	0.363	0.864	0.487	0.481	0.561	0.621	0.761	0.573
Cameras_occ[0.0,0.3]	0.590	0.516	0.952	0.631	0.594	0.701	0.784	0.862	0.704
Cameras_occ[0.0,0.5]	0.606	0.524	0.941	0.611	0.597	0.693	0.819	0.834	0.703
Cameras_occ[0.0,0.7]	0.577	0.475	0.935	0.602	0.588	0.748	0.773	0.810	0.688
Cameras_occ[0.0,0.9]	0.519	0.447	0.939	0.580	0.568	0.673	0.753	0.827	0.663

Table 2: Success rate result of Densefusion and Pvnet model trained on the scene and camera scenarios with the different levels of occlusion.

	Apple	Apricot	Banana	Kiwi	Lemon	Orange	Peach	Pear	Avg
Densefusion									
Scenes_occ[0.8, 0.9]	0.547	0.391	0.026	0.258	0.373	0.514	0.592	0.0	0.360
Scenes_occ[0.7, 0.8]	0.679	0.534	0.018	0.378	0.453	0.747	0.733	0.019	0.465
Scenes_occ[0.6, 0.7]	0.868	0.597	0.079	0.520	0.605	0.913	0.899	0.047	0.570
Scenes_occ[0.5, 0.6]	0.912	0.729	0.108	0.593	0.724	0.967	0.950	0.112	0.646
Scenes_occ[0.4, 0.5]	0.964	0.758	0.242	0.793	0.804	1.0	0.993	0.189	0.731
Scenes_occ[0.3, 0.4]	1.0	0.895	0.305	0.803	0.861	1.0	1.0	0.209	0.771
Scenes_occ[0.2, 0.3]	0.996	0.945	0.418	0.884	0.915	0.995	1.0	0.247	0.817
Scenes_occ[0.1, 0.2]	1.0	0.953	0.455	0.959	0.928	1.0	1.0	0.384	0.841
Scenes_occ[0.0, 0.1]	1.0	0.974	0.557	0.973	0.963	1.0	0.999	0.665	0.898
Pvnet									
Scenes_occ[0.8, 0.9]	0.258	0.220	0.896	0.478	0.557	0.558	0.489	0.723	0.522
Scenes_occ[0.7, 0.8]	0.248	0.286	0.898	0.519	0.498*	0.562	0.502	0.791	0.538
Scenes_occ[0.6, 0.7]	0.393	0.347	0.887	0.511	0.504	0.616	0.486	0.698	0.555
Scenes_occ[0.5, 0.6]	0.397	0.305	0.901	0.450	0.448	0.565	0.521	0.687	0.534
Scenes_occ[0.4, 0.5]	0.412	0.345	0.836	0.458	0.457	0.551	0.500	0.750	0.539
Scenes_occ[0.3, 0.4]	0.500	0.359	0.870	0.508	0.485	0.609	0.601	0.790	0.590
Scenes_occ[0.2, 0.3]	0.508	0.427	0.848	0.468	0.493	0.541	0.616	0.800	0.588
Scenes_occ[0.1, 0.2]	0.570	0.385	0.872	0.482	0.477	0.614	0.745	0.803	0.619
Scenes_occ[0.0, 0.1]	0.604	0.443	0.896	0.458	0.478	0.557	0.749	0.805	0.624

Table 3: Precise evaluation of Dense-fusion and PVnet trained on "Scenes_occ[0.0, 0.9]" over the testing partitions of level of occlusion

219 achieving a success rate of 90% with occlusion levels below 10%. However, the success rate drops
220 significantly to 30% when considering occlusion between 80% and 90%.

221 On the contrary, PVNet exhibits distinct characteristics. The method relies on keypoints, which re-
222 duces its dependence on occlusion. It demonstrates consistent performance across different scenarios,
223 yielding an average result of 64% (tables 2 and 3). Notably, there are variations in performance based
224 on the object, with notable peaks for pear, banana, and peach, which possess more intricate textures.
225 Conversely, other objects in the dataset can be considered texture-less, lacking any distinctive texture
226 specifications. Another important aspect to consider is that PVNet relies on pixel values, while our
227 dataset may exhibit varying distances between the camera and the objects. This can result in a limited
228 number of pixels available for inferring key points and subsequently estimating the 6D pose.

229 These experiments showcase the significant advantages of our dataset, enabling us to present
230 formidable challenges to state-of-the-art models that typically excel within the upper echelons
231 of standard datasets.

232 6 Limitation and future work

233 This work aims to introduce a benchmark for 6D pose estimation by presenting a dataset specifically
234 designed for this purpose. However, it is important to acknowledge that our dataset has limitations in
235 terms of fruit meshes. Given the nature of fruits, where each instance is unique, there is a need to
236 expand the dataset to include category 6D pose estimation. In order to address this, a logical step
237 would be to incorporate new vision annotations into the open-source software PickSim NOCS (28),
238 which is widely utilized for category 6D pose estimation. This addition would enhance the dataset
239 and enable comprehensive evaluation and analysis of category 6D pose estimation methods.

240 Furthermore, addressing the sim2real gap between our simulator and real fruits is an important aspect
241 to consider. We recognize the need for extensive studies on domain randomization techniques to
242 bridge this gap effectively, particularly in the context of robotics applications. By investigating
243 and refining domain randomization methods, we can enhance the simulation realism and improve
244 the transfer-ability of models trained in the simulator to real-world scenarios. These efforts will
245 contribute to advancing the field of robotics and promoting the seamless integration of simulated
246 environments with real-world applications.

247 7 Conclusion and discussion

248 In this paper, we present the largest dataset for fruit bin picking, comprising over 40 million 6D
249 pose annotations and 1 million images. The dataset constitutes significant challenges for 6D pose
250 estimation, as demonstrated in Section 5, including occlusion and texture-less objects. To address
251 different aspects of 6D pose estimation, we have created eight distinct datasets to evaluate scene
252 generalization, camera point of view generalization, and occlusion robustness. While the current
253 baseline model exhibits its own strengths, none of the baselines achieve satisfactory performance
254 across all benchmarks, offering the research community a challenging benchmark to tackle.

255 This dataset is curated specifically for 6D pose estimation, offering extensive annotations for enhanced
256 accessibility. Beyond its possible applications in 3D reconstruction, Nerf reconstruction, and multi-
257 view 6D pose estimation, it also holds significant value for robotics learning, highlighting one of its
258 key advantages. With its versatility, the dataset bridges the gap between computer vision and robotics,
259 fostering collaboration and innovation across various research areas. Our intention is for this dataset
260 to facilitate the improvement of existing 6D pose estimation models and drive further advancements
261 in the field of 6D pose for robotics.

262 Acknowledgments and Disclosure of Funding

263 This work was in part supported by the French Research Agency, l’ Agence Nationale de Recherche
264 (ANR), through the projects Learn Real (ANR-18-CHR3-0002-01), Chiron (ANR-20-IADJ-0001-01),
265 Aristotle (ANR-21-FAI1-0009-01). It was granted access to the HPC resources of IDRIS under the
266 allocation 2022-[AD011012172R1] made by GENCI.

267 References

- 268 [1] Eric Brachmann. ‘6d object pose estimation using 3d object coordinates, 2014. <https://heidata.uni-heidelberg.de/dataset.xhtml?persistentId=doi:10.11588/data/V4MUMX>.
- 269 [2] Xiaoyu Chen, Jiachen Hu, Chi Jin, Lihong Li, and Liwei Wang. Understanding domain randomization for
270 sim-to-real transfer. In *International Conference on Learning Representations, 2022*.
- 271 [3] Yuhao Chen, E. Zhixuan Zeng, Maximilian Gilles, and Alexander Wong. MetaGraspNet_v0: A Large-
272 Scale Benchmark Dataset for Vision-driven Robotic Grasping via Physics-based Metaverse Synthesis.
273 *ArXiv*, pages 1–7, dec 2021.
- 274 [4] D.M. Coleman, Ioan Alexandru Sucan, Sachin Chitta, and Nikolaus Correll. Reducing the barrier to entry
275 of complex robotic software: a moveit! case study. *ArXiv*, abs/1404.3785, 2014.
- 276 [5] Sudeep Dasari, Frederik Ebert, Stephen Tian, Suraj Nair, Bernadette Bucher, Karl Schmeckpeper, Sid-
277 dharth Singh, Sergey Levine, and Chelsea Finn. Robonet: Large-scale multi-robot learning. *CoRR*,
278 abs/1910.11215, 2019.
- 279

- 280 [6] Maximilian Denninger, Martin Sundermeyer, Dominik Winkelbauer, Youssef Zidan, Dmitry Olefir, Mo-
281 hamad Elbadrawy, Ahsan Lodhi, and Harinandan Katam. BlenderProc. *arXiv*, oct 2019.
- 282 [7] Guillaume Duret, Nicolas Cazin, Mahmoud Ali, Florence Zara, Emmanuel Dellandréa, Jan Peters, and
283 Liming Chen. PickSim: A dynamically configurable Gazebo pipeline for robotic manipulation. In
284 *Advancing Robot Manipulation Through Open-Source Ecosystems - 2023 IEEE International Conference*
285 *on Robotics and Automation (ICRA) Conference Workshop*, Londres, United Kingdom, May 2023. Adam
286 Norton, University of Massachusetts Lowell and Holly Yanco, University of Massachusetts Lowell and
287 Berk Calli, Worcester Polytechnic Institute and Aaron Dollar, Yale University.
- 288 [8] Maximilian Gilles, Yuhao Chen, Tim Robin Winter, E. Zhixuan Zeng, and Alexander Wong. MetaGrasp-
289 Net: A Large-Scale Benchmark Dataset for Scene-Aware Ambidextrous Bin Picking via Physics-based
290 Metaverse Synthesis. In *2022 IEEE 18th International Conference on Automation Science and Engineering*
291 *(CASE)*, volume 2022-Augus, pages 220–227. IEEE, aug 2022.
- 292 [9] Anas Gouda, Abraham Ghanem, and Christopher Reining. DoPose-6D dataset for object segmentation and
293 6D pose estimation. *ArXiv*, apr 2022.
- 294 [10] Matthieu Grard, Emmanuel Dellandréa, and Liming Chen. Deep multicameral decoding for localizing
295 unoccluded object instances from a single rgb image. *International Journal of Computer Vision*, 128, 05
296 2020.
- 297 [11] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan
298 Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam
299 Laradji, Hsueh Ti Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, Cengiz Oztireli, Etienne
300 Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S.M. Sajjadi, Matan Sela, Vincent Sitzmann,
301 Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and
302 Andrea Tagliasacchi. Kubric: A scalable dataset generator. In *Proceedings of the IEEE Computer Society*
303 *Conference on Computer Vision and Pattern Recognition*, volume 2022-June, pages 3739–3751. IEEE, jun
304 2022.
- 305 [12] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and
306 Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily
307 cluttered scenes. In *Computer Vision—ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon,*
308 *Korea, November 5-9, 2012, Revised Selected Papers, Part I 11*, pages 548–562. Springer, 2013.
- 309 [13] Tomas Hodan, Pavel Haluza, Stepan Obdrzalek, Jiri Matas, Manolis Lourakis, and Xenophon Zabulis.
310 T-LESS: An RGB-D Dataset for 6D Pose Estimation of Texture-Less Objects. In *2017 IEEE Winter*
311 *Conference on Applications of Computer Vision (WACV)*, pages 880–888. IEEE, mar 2017.
- 312 [14] Tomas Hodan, Vibhav Vineet, Ran Gal, Emanuel Shalev, Jon Hanzelka, Treb Connell, Pedro Urbina,
313 Sudipta N. Sinha, and Brian Guenter. Photorealistic Image Synthesis for Object Instance Detection. In
314 *2019 IEEE International Conference on Image Processing (ICIP)*, volume 2019-Sept, pages 66–70. IEEE,
315 sep 2019.
- 316 [15] Roman Kaskman, Sergey Zakharov, Ivan Shugurov, and Slobodan Ilic. HomebrewedDB : RGB-D Dataset
317 for 6D Pose Estimation of 3D Objects Technical University of Munich , Germany Siemens Corporate
318 Technology , Germany. *ICCV Workshop*, 2019.
- 319 [16] Kilian Kleeberger, Richard Bormann, Werner Kraus, and Marco F Huber. A Survey on Learning-Based
320 Robotic Grasping. *Current Robotics Reports*, 1(4):239–249, 2020.
- 321 [17] Kilian Kleeberger, Christian Landgraf, and Marco F. Huber. Large-scale 6D Object Pose Estimation
322 Dataset for Industrial Bin-Picking. In *2019 IEEE/RSJ International Conference on Intelligent Robots and*
323 *Systems (IROS)*, pages 2573–2578. IEEE, nov 2019.
- 324 [18] Nathan Koenig and Andrew Howard. Design and use paradigms for Gazebo, an open-source multi-
325 robot simulator. *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*,
326 3:2149–2154, 2004.
- 327 [19] Xingyu Liu, Shun Iwase, and Kris M. Kitani. StereOBJ-1M: Large-scale Stereo Image Dataset for 6D
328 Object Pose Estimation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages
329 10850–10859. IEEE, oct 2021.
- 330 [20] Samarth Mishra, Rameswar Panda, Cheng Perng Phoo, Chun-Fu (Richard) Chen, Leonid Karlinsky,
331 Kate Saenko, Venkatesh Saligrama, and Rogerio S. Feris. Task2sim: Towards effective pre-training and
332 transfer from synthetic data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern*
333 *Recognition (CVPR)*, pages 9194–9204, June 2022.
- 334 [21] Fabio Muratore, Fabio Ramos, Greg Turk, Wenhao Yu, Michael Gienger, and Jan Peters. Robot learning
335 from randomized simulations: A review. *Frontiers in Robotics and AI*, 9, 2021.
- 336 [22] Sida Peng, Yuan Liu, Qixing Huang, Hujun Bao, and Xiaowei Zhou. Pvnet: Pixel-wise voting network for
337 6dof pose estimation, 2018.
- 338 [23] Arul Selvam Periyasamy, Max Schwarz, and Sven Behnke. SynPick: A Dataset for Dynamic Bin Picking
339 Scene Understanding. In *2021 IEEE 17th International Conference on Automation Science and Engineering*
340 *(CASE)*, volume 2021-Augus, pages 488–493. IEEE, aug 2021.
- 341 [24] Jonathan Tremblay, Thang To, and Stan Birchfield. Falling Things: A Synthetic Dataset for 3D Object De-
342 tection and Pose Estimation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*
343 *Workshops (CVPRW)*, volume 2018-June, pages 2119–21193. IEEE, jun 2018.
- 344 [25] Stephen Tyree, Jonathan Tremblay, Thang To, Jia Cheng, Terry Mosier, Jeffrey Smith, and Stan Birchfield.
345 6-DoF Pose Estimation of Household Objects for Robotic Manipulation: An Accessible Dataset and

- 346 Benchmark. *IEEE International Conference on Intelligent Robots and Systems*, 2022-Octob:13081–13088,
 347 2022.
- 348 [26] Rutgers University. Rutgers apc rgb-d dataset, 2016. [https://robotics.cs.rutgers.edu/pracsys/
 349 rutgers-apc-rgb-d-dataset/](https://robotics.cs.rutgers.edu/pracsys/rutgers-apc-rgb-d-dataset/).
- 350 [27] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese.
 351 Densefusion: 6d object pose estimation by iterative dense fusion. In *2019 IEEE/CVF Conference on
 352 Computer Vision and Pattern Recognition (CVPR)*, pages 3338–3347, 2019.
- 353 [28] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J. Guibas. Normal-
 354 ized object coordinate space for category-level 6d object pose and size estimation. In *The IEEE Conference
 355 on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
- 356 [29] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. PoseCNN: A Convolutional Neural
 357 Network for 6D Object Pose Estimation in Cluttered Scenes. In *Robotics: Science and Systems XIV*.
 358 Robotics: Science and Systems Foundation, jun 2018.
- 359 [30] Honglin Yuan, Tim Hoogenkamp, and Remco C. Veltkamp. RobotP: A benchmark dataset for 6D object
 360 pose estimation. *Sensors (Switzerland)*, 21(4):1–26, 2021.

361 Checklist

362 The checklist follows the references. Please read the checklist guidelines carefully for information on
 363 how to answer these questions. For each question, change the default **[TODO]** to **[Yes]**, **[No]**, or
 364 **[N/A]**. You are strongly encouraged to include a **justification to your answer**, either by referencing
 365 the appropriate section of your paper or providing a brief inline description. For example:

- 366 • Did you include the license to the code and datasets? **[Yes]** See Section
- 367 • Did you include the license to the code and datasets? **[No]** The code and the data are
 368 proprietary.
- 369 • Did you include the license to the code and datasets? **[N/A]**

370 Please do not modify the questions and only use the provided macros for your answers. Note that the
 371 Checklist section does not count towards the page limit. In your paper, please delete this instructions
 372 block and only keep the Checklist section heading above along with the questions/answers below.

- 373 1. For all authors...
 - 374 (a) Do the main claims made in the abstract and introduction accurately reflect the paper’s
 375 contributions and scope? **[Yes]**
 - 376 (b) Did you describe the limitations of your work? **[Yes]** See Section 6.
 - 377 (c) Did you discuss any potential negative societal impacts of your work? **[N/A]**
 - 378 (d) Have you read the ethics review guidelines and ensured that your paper conforms to
 379 them? **[Yes]**
- 380 2. If you are including theoretical results...
 - 381 (a) Did you state the full set of assumptions of all theoretical results? **[N/A]**
 - 382 (b) Did you include complete proofs of all theoretical results? **[N/A]**
- 383 3. If you ran experiments (e.g. for benchmarks)...
 - 384 (a) Did you include the code, data, and instructions needed to reproduce the main experi-
 385 mental results (either in the supplemental material or as a URL)? **[Yes]** See supplemental
 386 material
 - 387 (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they
 388 were chosen)? **[Yes]** See section 6
 - 389 (c) Did you report error bars (e.g., with respect to the random seed after running experi-
 390 ments multiple times)? **[No]**
 - 391 (d) Did you include the total amount of compute and the type of resources used (e.g.,
 392 type of GPUs, internal cluster, or cloud provider)? **[Yes]** See Acknowledgement and
 393 supplemental material.

- 394 4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
- 395 (a) If your work uses existing assets, did you cite the creators? [Yes] We used the code of
- 396 PickSim
- 397 (b) Did you mention the license of the assets? [N/A]
- 398 (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
- 399
- 400 (d) Did you discuss whether and how consent was obtained from people whose data you're
- 401 using/curating? [N/A]
- 402 (e) Did you discuss whether the data you are using/curating contains personally identifiable
- 403 information or offensive content? [N/A]
- 404 5. If you used crowdsourcing or conducted research with human subjects...
- 405 (a) Did you include the full text of instructions given to participants and screenshots, if
- 406 applicable? [N/A]
- 407 (b) Did you describe any potential participant risks, with links to Institutional Review
- 408 Board (IRB) approvals, if applicable? [N/A]
- 409 (c) Did you include the estimated hourly wage paid to participants and the total amount
- 410 spent on participant compensation? [N/A]

411 A Example of Images from the dataset

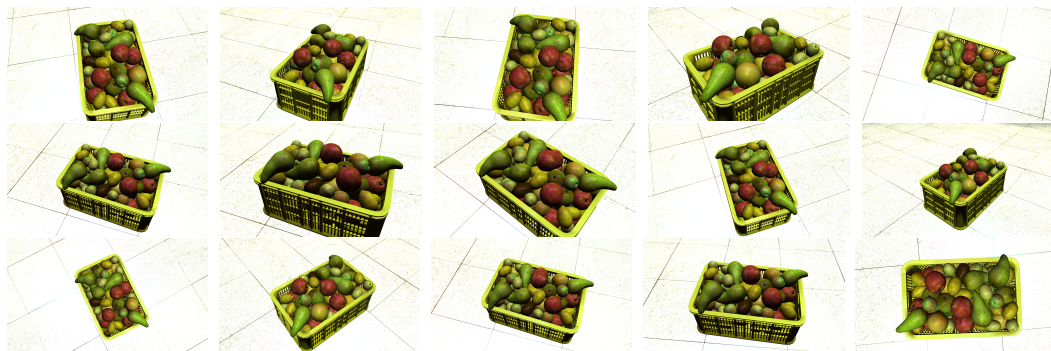


Figure 6: 15 point of view of a single scene of FruitBin

412 B Appendix

413 Include extra information in the appendix. This section will often be part of the supplemental material.
 414 Please see the call on the NeurIPS website for links to additional guides on dataset publication.

- 415 1. Submission introducing new datasets must include the following in the supplementary
- 416 materials:
- 417 (a) Dataset documentation and intended uses. Recommended documentation frameworks
- 418 include datasheets for datasets, dataset nutrition labels, data statements for NLP, and
- 419 accountability frameworks.
- 420 (b) URL to website/platform where the dataset/benchmark can be viewed and downloaded
- 421 by the reviewers.
- 422 (c) Author statement that they bear all responsibility in case of violation of rights, etc., and
- 423 confirmation of the data license.
- 424 (d) Hosting, licensing, and maintenance plan. The choice of hosting platform is yours, as
- 425 long as you ensure access to the data (possibly through a curated interface) and will
- 426 provide the necessary maintenance.

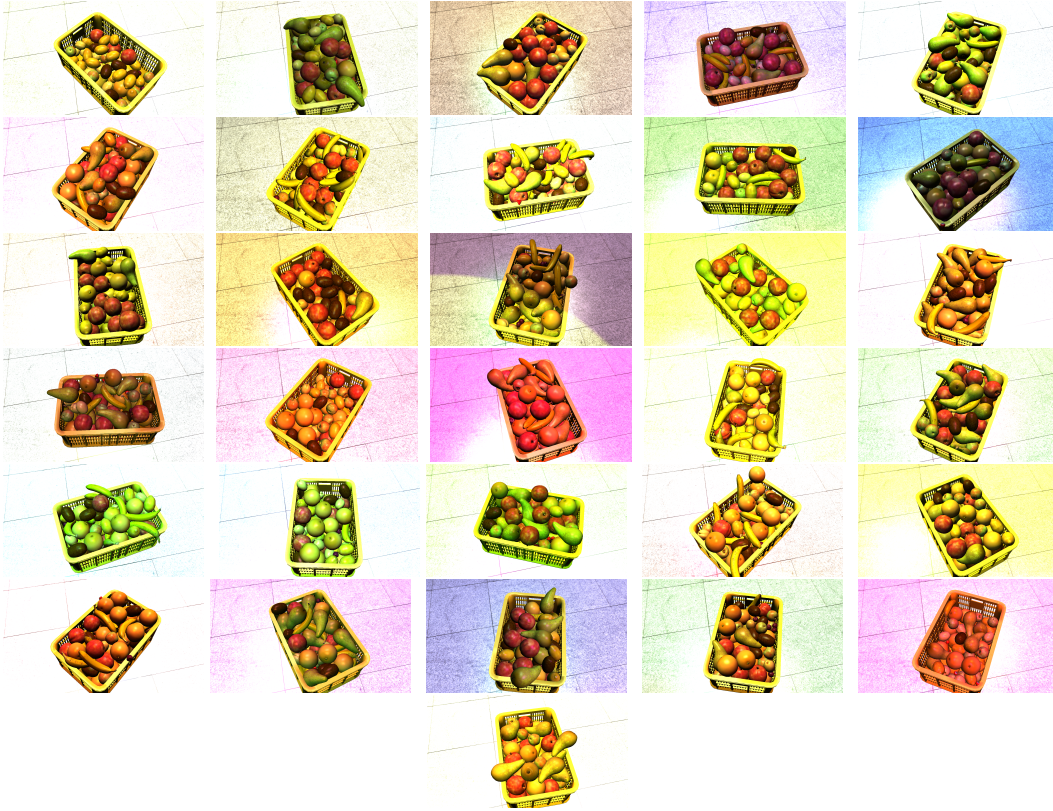


Figure 7: 31 first scenes for a unique point of view

- 427 2. To ensure accessibility, the supplementary materials for datasets must include the following:
- 428 (a) Links to access the dataset and its metadata. This can be hidden upon submission if the
- 429 dataset is not yet publicly available but must be added in the camera-ready version. In
- 430 select cases, e.g when the data can only be released at a later date, this can be added
- 431 afterward. Simulation environments should link to (open source) code repositories.
- 432 (b) The dataset itself should ideally use an open and widely used data format. Provide a
- 433 detailed explanation on how the dataset can be read. For simulation environments, use
- 434 existing frameworks or explain how they can be used.
- 435 (c) Long-term preservation: It must be clear that the dataset will be available for a long time,
- 436 either by uploading to a data repository or by explaining how the authors themselves
- 437 will ensure this.
- 438 (d) Explicit license: Authors must choose a license, ideally a CC license for datasets, or an
- 439 open source license for code (e.g. RL environments).
- 440 (e) Add structured metadata to a dataset's meta-data page using Web standards (like
- 441 schema.org and DCAT): This allows it to be discovered and organized by anyone. If
- 442 you use an existing data repository, this is often done automatically.
- 443 (f) Highly recommended: a persistent dereferenceable identifier (e.g. a DOI minted by
- 444 a data repository or a prefix on identifiers.org) for datasets, or a code repository (e.g.
- 445 GitHub, GitLab,...) for code. If this is not possible or useful, please explain why.
- 446 3. For benchmarks, the supplementary materials must ensure that all results are easily repro-
- 447 ducible. Where possible, use a reproducibility framework such as the ML reproducibility
- 448 checklist, or otherwise guarantee that all results can be easily reproduced, i.e. all necessary
- 449 datasets, code, and evaluation procedures must be accessible and documented.



Figure 8: 31 first scenes for a unique point of view

450
451

4. For papers introducing best practices in creating or curating datasets and benchmarks, the above supplementary materials are not required.