machines

MDPI

Article

Sim-to-Real Dataset of Industrial Metal Objects

Peter De Roovere [1,*], Steven Moonen [2], Nick Michiels [2] and Francis wyffels [1]

[1] IDLab-AIRO, Ghent University–imec, 9052 Ghent, Belgium; francis.wyffels@ugent.be
[2] Expertise Centre for Digital Media, Flanders Make, Hasselt University, 3590 Diepenbeek, Belgium; steven.moonen@uhasselt.be (S.M.); nick.michiels@uhasselt.be (N.M.)
* Correspondence: peter.deroovere@ugent.be

Abstract: We present a diverse dataset of industrial metal objects with unique characteristics such as symmetry, texturelessness, and high reflectiveness. These features introduce challenging conditions that are not captured in existing datasets. Our dataset comprises both real-world and synthetic multi-view RGB images with 6D object pose labels. Real-world data were obtained by recording multi-view images of scenes with varying object shapes, materials, carriers, compositions, and lighting conditions. This resulted in over 30,000 real-world images. We introduce a new public tool that enables the quick annotation of 6D object pose labels in multi-view images. This tool was used to provide 6D object pose labels for all real-world images. Synthetic data were generated by carefully simulating real-world conditions and varying them in a controlled and realistic way. This resulted in over 500,000 synthetic images. The close correspondence between synthetic and real-world data and controlled variations will facilitate sim-to-real research. Our focus on industrial conditions and objects will facilitate research on computer vision tasks, such as 6D object pose estimation, which are relevant for many industrial applications, such as machine tending. The dataset and accompanying resources are available on the project website.

Keywords: dataset; 6D object pose estimation; industrial robotics; sim-to-real; reflective materials

check for updates

Citation: De Roovere, P.; Moonen, S.; Michiels, N.; wyffels, F. Sim-to-Real Dataset of Industrial Metal Objects. *Machines* **2024**, *12*, 99. https://doi.org/10.3390/machines12020099

Academic Editor: Fugui Xie

Received: 8 December 2023
Revised: 13 January 2024
Accepted: 24 January 2024
Published: 1 February 2024

1. Introduction

6D object pose estimation is crucial for industrial robotics [1]. As the manufacturing industry moves towards High-Mix, Low-Volume (HMLV) production, robots must adapt to diverse products without human intervention. In the past, robots were limited to fixed repetitive actions, but HMLV manufacturing requires more flexibility [2]. 6D object pose estimation enables robots to interact with diverse objects in different compositions. This eliminates the need for costly and inflexible fixtures, resulting in more adaptable and precise systems. Despite its significance and the extensive history of research in this area, 6D object pose estimation remains an open problem and an active area of research [3].

Datasets are fundamental to academic research, especially in fields such as computer vision. They form the foundation for creating and validating new theories and technologies. Prominent benchmarks, like the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [4], Common Objects in Context (COCO) [5], the KITTI Vision Benchmark Suite (KITTI) [6], and the Benchmark for 6D Object Pose Estimation (BOP) [7], have become synonymous with progress in image recognition, object detection, and more recently, 6D object pose estimation. These benchmarks rely on real-world datasets to evaluate the effectiveness of different methods, measure progress, and compare advancements. In addition, learning-based approaches have outperformed traditional techniques in many computer vision tasks [8]. Datasets play a critical role in training these methods. The extent and depth of the data used in training are crucial factors in these systems' ability to generalize to new, real-world scenarios.

Creating high-quality datasets for 6D object pose estimation is a difficult task. Labeling the six degrees of freedom that define an object's pose is a complex, time-consuming process that requires great attention to detail. Moreover, it is challenging to ensure the accuracy of these labels. Synthetic data have emerged as a potential alternative to manual labeling. Recent advances in computer graphics make it feasible to generate large amounts of labeled data. Some 6D object pose estimation datasets adopt this strategy [9,10]. However, the resulting images often fail to capture essential aspects of their real-world counterparts, like object materials, scene compositions, lighting, and camera viewpoints. Variations between synthetic and real-world data can significantly hamper model performance—the so-called reality gap [11]. A proposed solution to the reality gap is domain randomization [12], in which parts of the data generation process are randomized. With enough variability, real-world data may appear to be another variation. However, generating *enough* variability to envelope real-world data points is non-trivial. Many variations are wasteful and can unnecessarily increase the task difficulty [13]. Matching the distribution of real-world data through synthetic data generation is a significant challenge. Additionally, generating high-quality synthetic data requires significant computational power and storage capabilities. Balancing the quality and quantity of data while keeping computational and storage costs in check requires careful planning and strategic decision-making. It is difficult to determine which variations are valuable, as existing datasets do not provide information about the variations used in the data generation process. Currently, no datasets exist that enable sim-to-real research on these variations for industrial settings.

In recent years, there has been significant progress in performance in the task of 6D object pose estimation and progress on current datasets is slowing [8]. Therefore, it is crucial to focus on more challenging conditions to maintain the momentum of technological progress and ensure that research remains relevant and capable of addressing complex real-world applications. Many existing datasets for estimating the pose of 6D objects have limitations, which make them less applicable to industrial scenarios. The Benchmark for 6D Object Pose Estimation (BOP) [7] unifies 12 datasets, of which the majority focus on household objects, which differ significantly from industrial ones [14]. Household objects are often non-symmetric with diverse textures, making them easier to identify and position. However, industrial objects are often symmetrical and lack discernible textures, making them more challenging to estimate accurately. Additionally, the reflectiveness of many industrial objects and environmental factors like scratches, dust, or dirt can further complicate this process. Other datasets incorporate textureless industrial objects [15,16] but focus on objects with limited reflectivity. However, reflective materials are widespread. For example, most high-precision machine components are made of shiny metals. These objects are particularly interesting as depth sensors cannot capture them correctly [17,18]. Secondly, most current 6D pose estimation datasets rely on single-view images, leading to appearance and depth perception issues [19]. However, in many industrial settings, multi-view approaches can be used to circumvent these ambiguities. Images can be captured from different viewpoints using multi-camera or eye-in-hand setups, enriching the information available.

To summarize, existing datasets present a significant sim-to-real gap, provide no information about the variations in data generation, focus on single-view scenes, and are irrelevant for industrial use cases. We aim to overcome these limitations by presenting the following contributions:

- A collection of over 30,000 real-world images depicting industrial metal objects. These images showcase objects of varying shapes, materials, composition types, carriers, and lighting conditions. They were captured using different cameras and from multiple angles;
- A collection of over 500,000 synthetic images. We provide synthetic counterparts for all real-world images, along with additional images that were generated by varying environmental conditions, such as lighting, object materials, object positions, and carriers in a controlled manner.

Figure 1 shows an example of a real-world image and its synthetic counterpart.

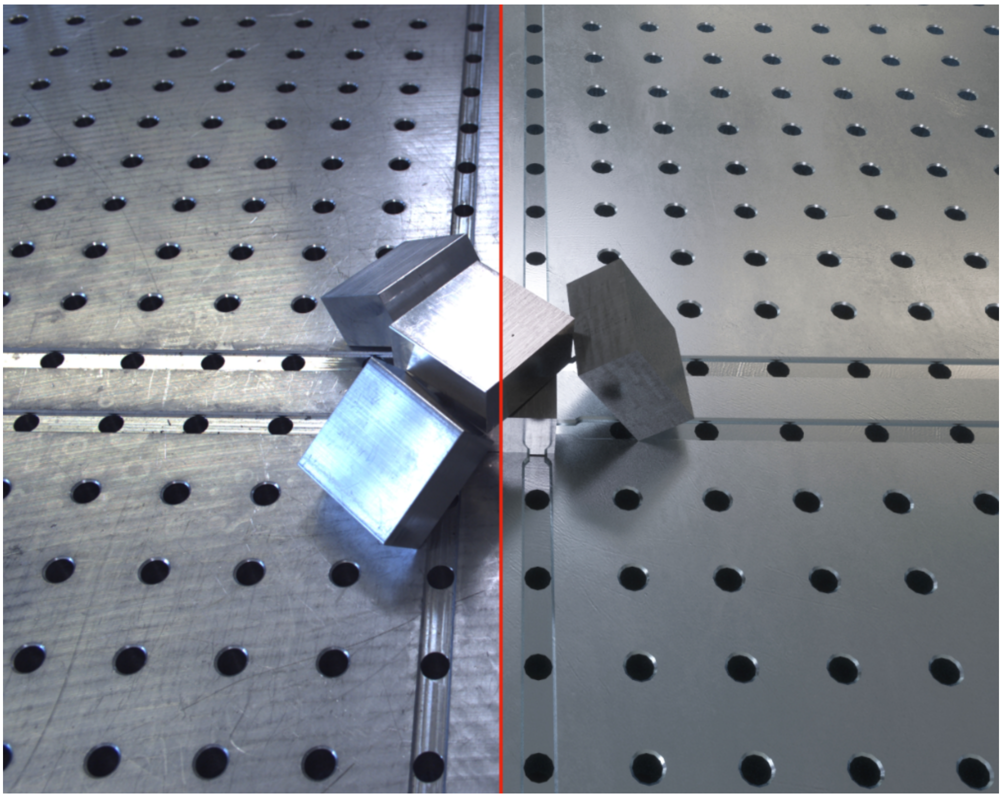


Figure 1. Side-by-side comparison of a real-world image (**left**) and its synthetic counterpart (**right**).

Figure 2 illustrates the challenges encountered in handling objects that are highly reflective and lack texture. Although 3D sensors and methods that are trained on large datasets can effectively deal with objects that have texture and are non-reflective, they face significant difficulties when it comes to dealing with industrial parts and backgrounds that lack texture and have high reflectivity.

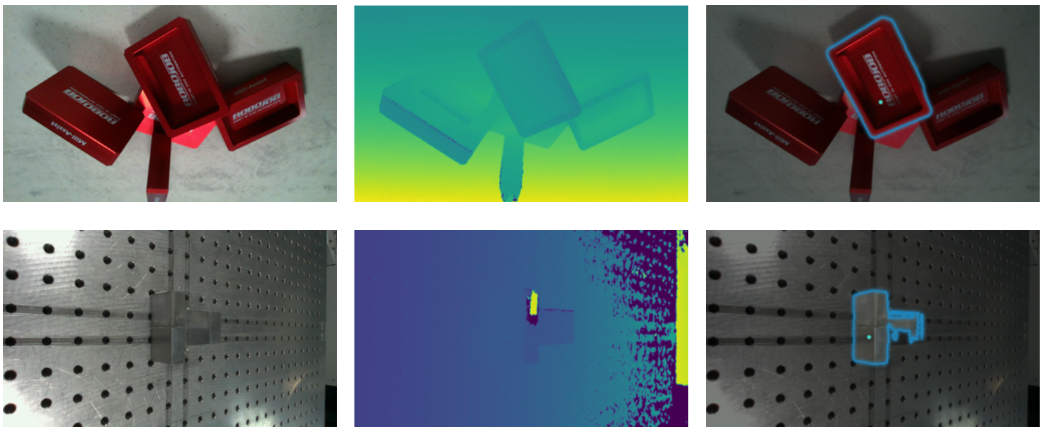


Figure 2. Depth sensors and methods trained on existing datasets perform poorly for reflective objects. RGB image (**left**), depth image (**center**), and object segmentation (**right**) for a scene with textured, non-reflective objects (**top**), and textureless, reflective objects (**bottom**). Depth images were recorded using a RealSense L515 Lidar camera. Object segmentation masks were obtained using Segment Anything [20], without any additional fine-tuning. The query point is shown (green) in addition to the obtained mask (blue).

In addition to our dataset, we provide an open-source tool that enables users to easily label 6D object poses for multi-view data. Correspondences between image pixels and object CAD points can be marked to calculate coarse object poses. The tool then overlays the object CAD file on the image, allowing users to fine-tune the 6D pose label using 3D position tools. It handles multi-view images, jointly labeling all views, improving accuracy, and reducing labeling effort.

Our dataset has significant value for various industry-relevant research problems. The congruity between synthetic and real-world data and controlled variations will facilitate sim-to-real research. Our dataset is conducive to researching several vital computer vision tasks involving industrial objects. We focus our discussion on 6D object pose estimation, but our dataset also applies to object detection, instance segmentation, novel view synthesis, 3D reconstruction, and active perception.

In addition to our dataset, we provide an open-source tool that enables users to easily label 6D object poses for multi-view data. Many specialized tools exist for labeling computer vision data. Prominent platforms, such as V7 [21], Labelbox [22], Scale AI Rapid [23], SuperAnnotate [24], Dataloop [25], Supervise.ly [26], and Segments.ai [27], have established themselves as key players in this field. Each platform offers a unique set of features tailored to specific annotation needs, such as classification, object detection, key points, and segmentation. However, it is noteworthy that these tools do not support labeling 6D object poses, which involves annotating the three-dimensional position and orientation of objects in space. This complex task requires a higher degree of precision and understanding of spatial relationships, which is not typically addressed by standard annotation tools. Furthermore, most of these platforms do not emphasize the consistency of labels across different viewpoints and sensor types, except for Segments.ai. This consistency, however, is crucial for developing robust computer vision models that can reliably interpret data from various perspectives and sensors, which is key for many real-world robotics applications.

Only a few tools are available that enable the precise labeling of 6D object poses, highlighting a niche but significant area in the field of data annotation for computer vision. Table 1 gives an overview of the tools for 6D object pose estimation.

Table 1. Comparison of tools for labeling 6D object poses.

Tool	Multi-View	Robustness to Reflections	Ease of Use [1]
6D-PAT		✓	**
Labelfusion	✓		***
DoPose		✓	*
Ours	✓	✓	****

[1] Ease of Use is rated from * to **** where * represents the lowest ease and **** indicates the highest. For a detailed explanation, refer to Appendix A.

6D-PAT [28] closely resembles the functionalities of our tool but does not support labeling multi-view images with consistent labels across views. DoPose [29] has limited capabilities and also lacks support for multi-view data. This feature is essential for creating datasets where the same object is observed from various angles, ensuring consistent and accurate labels in 3D, which, in turn, is crucial for robotics applications. Labelfusion [30] relies heavily on 3D reconstruction and Iterative Closest Point (ICP) algorithms. However, these methods are not suitable for dealing with images of reflective objects, which is a limitation that significantly impacts the tool's applicability in many industrial contexts where reflective surfaces are common.

Our tool addresses these specific gaps. It is unique in its capability of labeling multi-view-consistent 6D object poses, which is critical for achieving spatial accuracy in 3D. Additionally, it introduces a wide range of unique features that streamline the labeling process, making it more versatile and efficient for complex 6D pose labeling tasks. This is especially important in scenarios where multi-view consistency and precision are paramount, such as in high-precision robotics tasks.

2. Materials and Methods

First, we introduced a data format that is based on the BOP format [7] and is specifically designed for multi-view 6D object pose data with controlled conditions. Next, we established a system that allowed us to collect real-world data with different camera and conditions. With this system, we gathered a significant real-world dataset of industrial metal objects. To label the collected dataset, we developed a specialized tool that is dedicated to the annotation of 6D object poses. Finally, we detailed our approach for producing high-fidelity synthetic data that closely replicates real-world conditions, along with controlled variants.

2.1. Data Format

We adopted and extended the BOP format [7]. The goal of BOP is to capture the state-of-the art in 6D object pose estimation. The benchmark provides 12 datasets in a unified format. The format incorporates camera intrinsics and extrinsics. However, for multi-view datasets, the provided per-image object labels often do not match in a shared world frame. In addition, no information is provided about the conditions in which scenes were captured. We extended this format to include this information. Object labels in a joint world frame were added, along with information about the lighting, carrier, and object composition type. Additionally, we included drawings that can be used to construct each of the objects, so they can be used in real-world experiments. The resulting folder structure is shown in Figure 3. All distances are recorded in mm, coordinate systems are right-handed, and matrices and vectors are flattened, following the BOP convention.

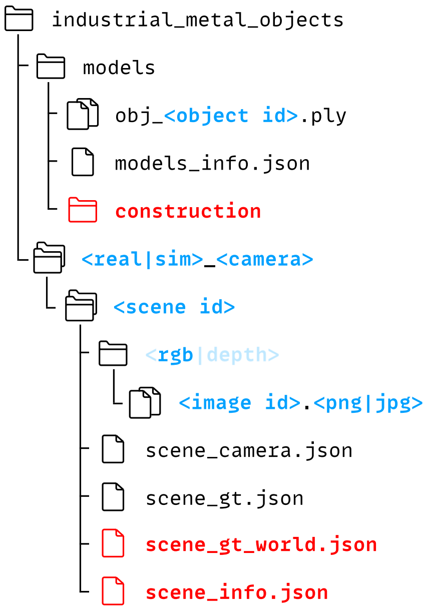


Figure 3. Folder structure of our dataset. Our dataset contains a sub-folder with object CAD models and sub-folders for each camera (real and virtual). These camera folders contain sub-folders for each scene. Each scene folder contains a sub-folder with multi-view images and json files with accompanying information. We adopted the BOP format and extended it (marked in red).

Object CAD models are stored in PLY format in `models`. `models_info.json` contains information about the size and symmetries of each object, as shown in Listing 1. Continuous symmetries are described by axis and offset values, discrete symmetry by transformation matrices.

Listing 1. `models_info.json` contains information about the CAD models.

```
{
    "<object id>": {
        "diameter": 300.0,
        "min_x": −12.5,
        "min_y": −12.5,
        "min_z": −150.0,
        "size_x": 25.0,
        "size_y": 25.0
        "size_z": 300.0,
        "symmetries_continuous": [
            {
                "axis": <(3,)>,
                "offset": <(3,)>
            }, ...
        ],
        "symmetries_discrete": [
            <(4,4)>, ...
        ]
    }, ...
}
```

Technical drawings that can be used for manufacturing the various parts are contained in `construction`. Scenes are organized by camera type (both real and virtual) and identified by a unique id. `scene_camera.json` contains camera intrinsics and extrinsics for each image, as shown in Listing 2.

Listing 2. `scene_camera.json` contains camera intrinsics and extrinsics for each image.

```
{
    "<image id>": {
        "cam_K": <(3,3)>,
        "cam_R_c2w": <(4,4)>,
        "cam_t_c2w": <(3,)>
    }, ...
}
```

Object poses are recorded in `scene_gt.json` and `scene_gt_world.json`, as shown in Listings 3 and 4. For each image, `scene_gt.json` contains object poses relative to the camera reference frame. As poses are consistent between different viewpoints, we also record object poses in a shared global reference frame (`scene_gt_world.json`).

`scene_info.json` contains information about the data generation process, as presented in Listing 5. For each image, we record which lighting conditions apply, which carrier is used, the type of composition (heterogeneous or homogeneous), and the camera viewpoint. This information allows selecting sub-sets on the data based on specific variations.

Listing 3. `scene_gt.json` contains object poses (relative to the camera) for each image.

```
{
    "<image id>": [{
            "cam_K": <(3,3)>,
            "cam_R_m2c": <(4,4)>,
            "cam_t_m2c": <(3,)>,
            "obj_id": 2,
        }, ...]
}
```

Object poses (relative to a global world reference) are contained in

Listing 4. `scene_gt_world.json`. These poses are valid for all images.

```
{[
    {
        "cam_K": <(3,3)>,
        "cam_R_m2w": <(4,4)>,
        "cam_t_m2w": <(3,)>,
        "obj_id": 2
    }, ...
]}
```

Information about the data generation process for each image is contained in

Listing 5. `scene_info.json`.

```
{
    "<image id>": {
            "light": <lightmap_id>,
            "carrier": <carrier_id>,
            "parts": <0 (mix) | obj_id>,
            "viewpoint" <viewpoint_id>,
        }, ...
}
```

2.2. Real-World Data

We used different camera sensors (Table 2) mounted on an industrial robot to capture multi-view images of diverse scenes. Figure 4 shows the setup for collecting real-world data. Figure A1 shows images of each of the sensors. A custom-designed end-effector attached all cameras to a Fanuc M20ia robot, as shown in Figure 5. With a repeatability of 0.1 mm, the robot was used to change camera viewpoints accurately. We further stabilized camera poses by canceling backlash using the Fanuc instruction `IRVBKLSH`. We calibrated all cameras using ChArUco targets, undistorted all images, and performed hand–eye calibration [31]. We used high-precision aluminum-LDPE calibration targets with `DICT_5x5` dictionary codes and 15mm-wide checkers. ChArUco detection and calibration were implemented using OpenCV [32]. Each scene was captured with every camera from 13 different viewpoints. Figure A2 shows images of each of the viewpoints. These viewpoints were spread out on a hemisphere and oriented towards the center of the object carrier, as shown in Figure 6.

Table 2. The different cameras used for capturing real-world image data.

Camera	Manufacturer	Lens	Type	Resolution
GO-5000-PGE	JAI (Copenhagen, Denmark)	KOWA LM12HC-V	RGB	2560 × 2048
mvBlueFOX3-2124rG-1112	Balluff (Neuhausen auf den Fildern, Germany)	MV-O1218-10M-KO	Grayscale	4064 × 3044
RealSense L515	Intel (Santa Clara, CA, USA)		RGB, LiDAR	1920 × 1080
RealSense D415	Intel (Santa Clara, CA, USA)		RGB, Active IR Stereo	1920 × 1080

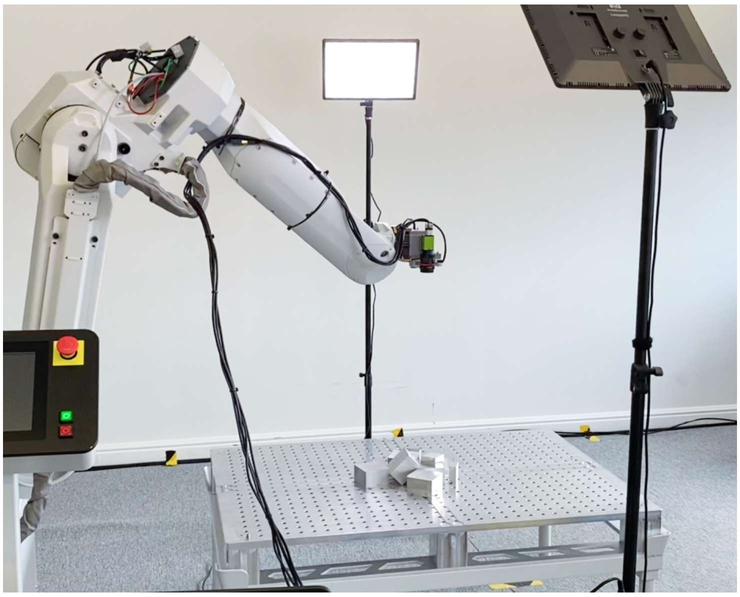


Figure 4. Oursetup for collecting real-world images. We captured multi-view images of diverse scenes using different camera sensors mounted on an industrial robot.

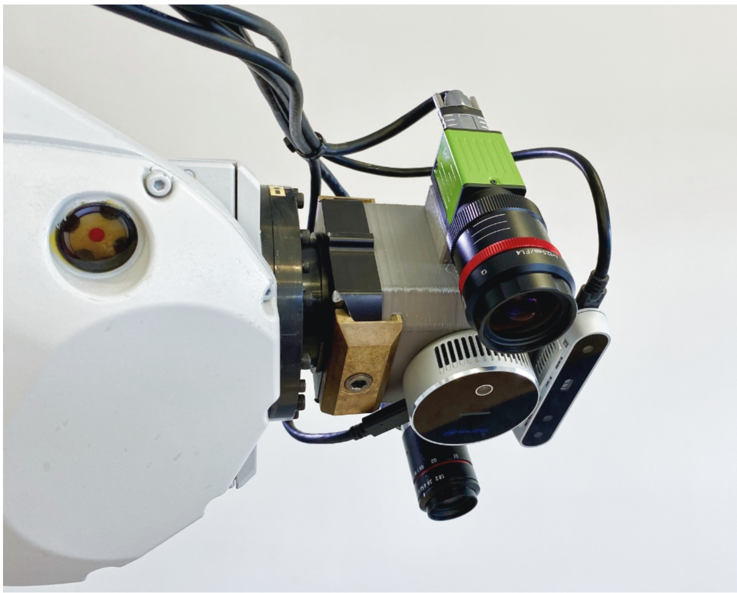


Figure 5. Different camera sensors were attached to the end-effector of an industrial robot.

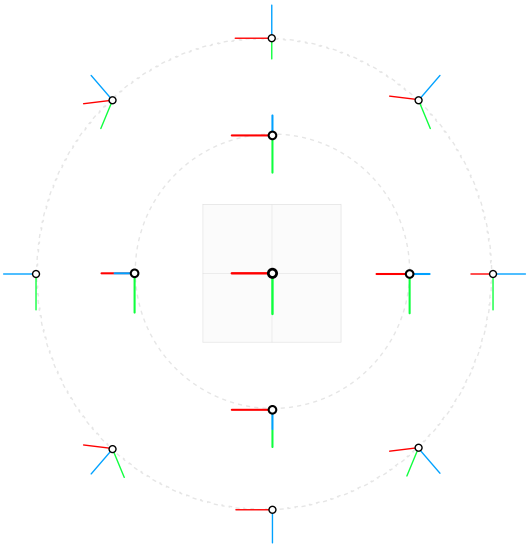
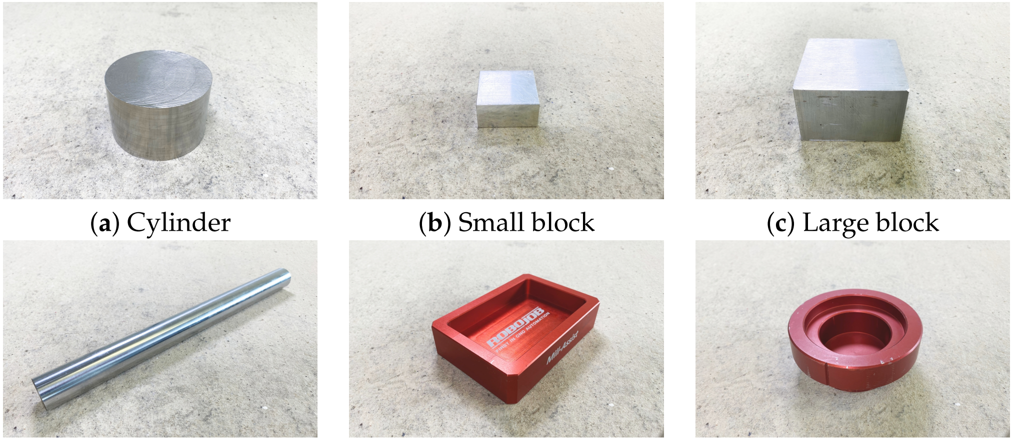


Figure 6. Top-viewvisualization of all 13 camera viewpoints. These viewpoints were spread out in a hemisphere, centered around and oriented towards where objects were placed.

Varying object shapes, materials, carriers, compositions, and lighting created diverse scenes. Figure A3 shows example images with different objects, carriers, compositions, and lighting types. The selected variations were meant to imitate realistic variations that occur in industrial settings, specifically in metal manufacturing factories. We used six object categories, shown in Figure 7, with different shapes and material properties. Most objects (Figure 7a–d) were highly reflective and exhibited real-world features like scratches and saw patterns. These objects represent raw or half-finished workpieces. We also included less reflective (Figure 7e,f) and textured objects (Figure 7e) for comparison. These objects represent finished workpieces. We used three object carriers, as shown in Figure 8. In manufacturing environments, objects are usually transported in bins or on pallets, with or without cardboard. Parts are stacked in various compositions with different levels of occlusion. Lighting conditions were varied by controlling Nanguan Luxpad 43 lights. We replicated both daylight and nighttime conditions, with and without factory lighting.



(**a**) Cylinder (**b**) Small block (**c**) Large block

(**d**) Shaft (**e**) Red block (**f**) Red cylinder

Figure 7. Object categories. We used objects from 6 categories with different shapes and material properties. Most objects (**a**–**d**)were highly reflective and exhibited real-world features like scratches. For comparison, we also included less reflective and textured objects (**e**,**f**).

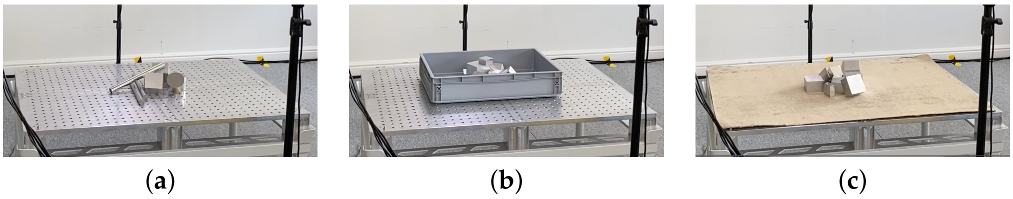
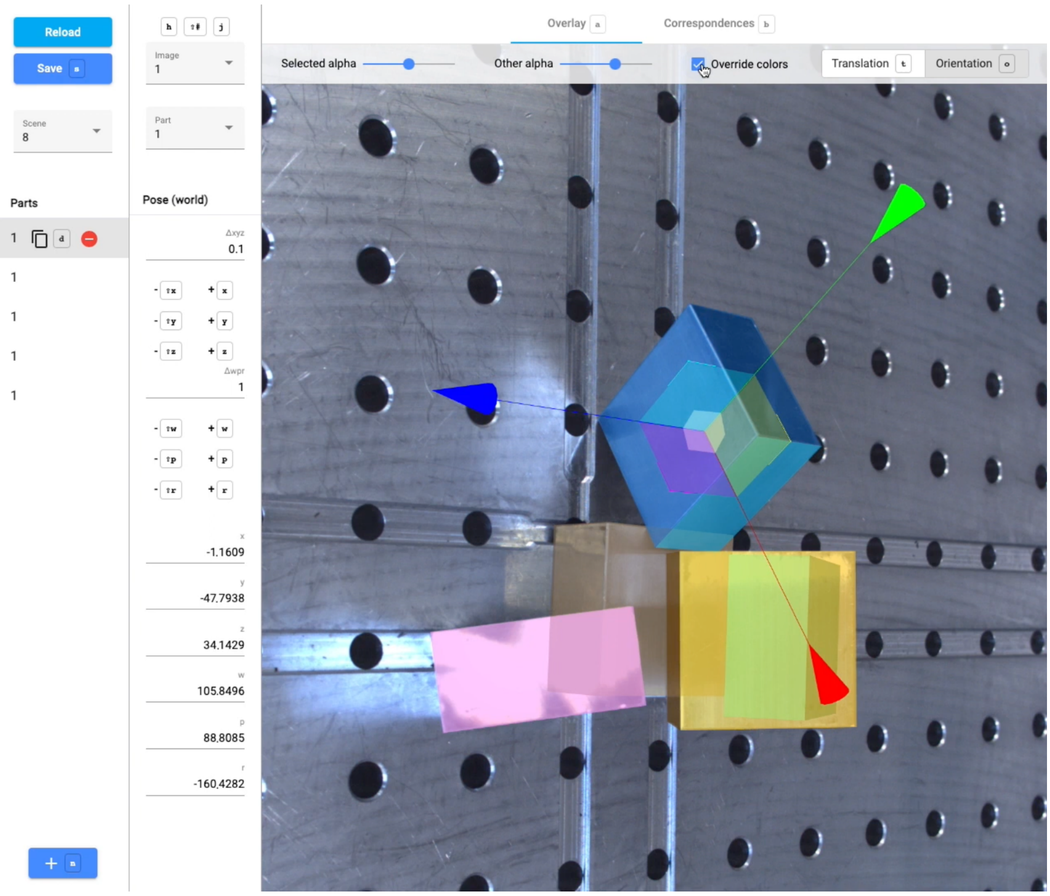


Figure 8. Objects were placed on three different types of carriers: (**a**) pallet; (**b**) bin; (**c**) cardboard.

2.3. Labeling Tool

To label the recorded real-world data and, more generally, the datasets stored in the presented format, we developed a tool for the quick and accurate labeling of 6D object poses that are consistent across multiple views. Labeling 6D object poses on 2D images can be time-consuming and tedious. An object's six degrees of freedom interact in intricate ways, making it difficult for annotators to bridge the gap between the 2D image and the object's actual 3D position and orientation. In addition, labelers must understand 3D perspectives and correctly handle object occlusions. Achieving 3D accuracy is crucial for industrial robotics applications, where sub-millimeter precision is often required. However, minor errors made during labeling in pixel space can lead to significant errors in 3D space. For example, moving objects along the camera's z-axis can lead to a big difference in their 3D position, with minimal pixel changes. We address these challenges by providing an intuitive user interface and allowing for the joint labeling of multi-view images. Figure 9 shows the labeling tool.



Scene 8

Image 1

Part 1

Δxyz 0.1

Δwpr 1

x -1.1609

y -47.7938

z 34.1429

w 105.8496

p 88.8085

r -160.4282

☑ Override colors

Figure 9. Screenshot ofthe labeling tool. Users can reload or save the data on the left, scroll through the available scenes, and manage object labels. Next, users can scroll through images of the selected scene, assign the correct part type, and manually edit the pose. In the center, a visualization of the labeled objects is shown on top of the selected image, along with the controls used to interact with the pose of the selected object. Users can change the opacity of the objects overlaid on the image.

2.3.1. Workflow

Figure 10 shows the workflow of our labeling tool.

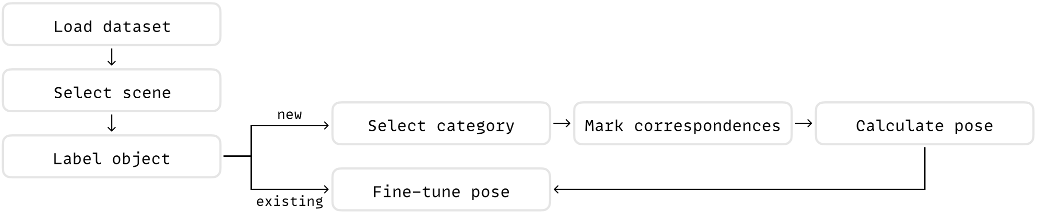
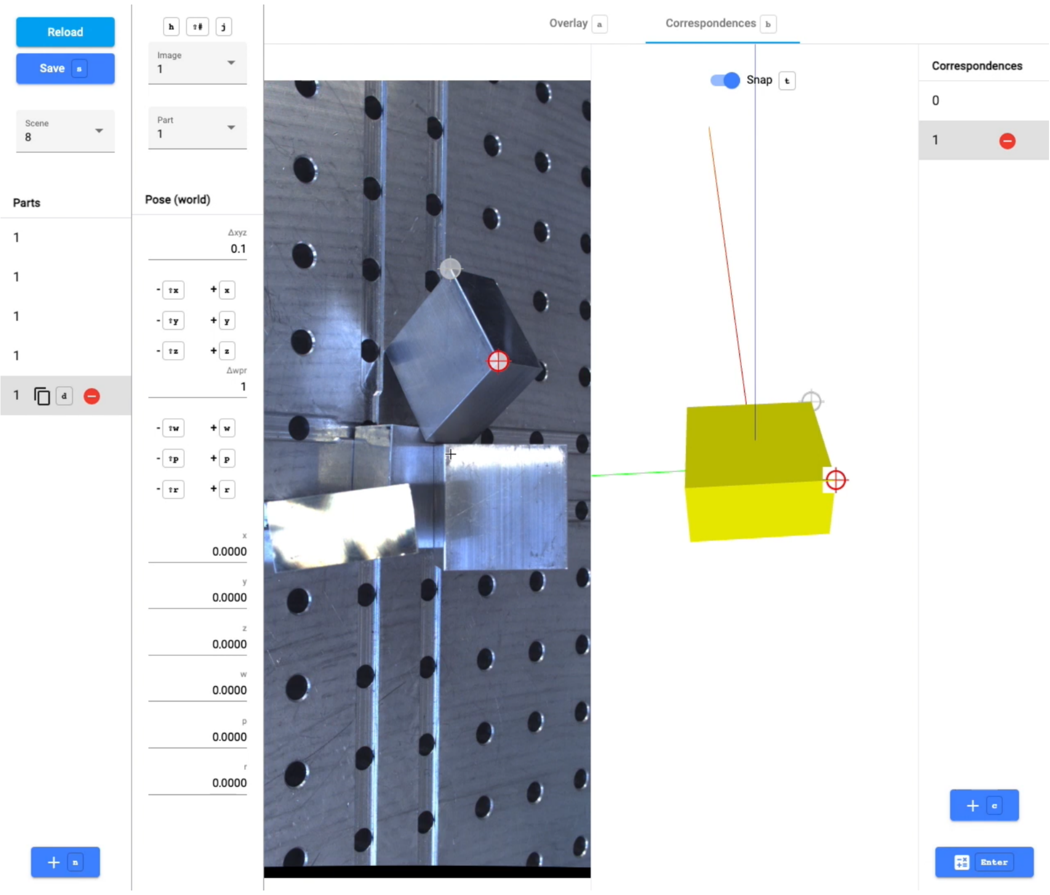


Figure 10. Labeling tool workflow.

After loading a dataset, users can browse the available scenes and select one. To create a new object label, users first select the object category. Then, an initial pose can be calculated after marking at least four correspondences between image pixels and points on the object CAD model. Figure 11 shows the interface for marking correspondences.



Scene
8

Image
1

Part
1

Δxyz
0.1

Δwpr
1

x
0.0000

y
0.0000

z
0.0000

w
0.0000

p
0.0000

r
0.0000

Figure 11. After definingcorrespondences between image pixels and points on the CAD model of the object, the 6D pose of the object can be calculated.

Once this initial pose has been calculated, users can fine-tune it using the interactive controls provided by the visualization. These controls allow for changing the 6D object pose, including translation and rotation along all axes or planes. Additionally, users can incrementally update the pose by moving fixed steps along all axes. The alpha and color of each object can be changed to improve the visualization.

The entire dataset can be saved at any moment.

It is essential to note that object poses are configured on a per-scene level. This means that users can adjust the pose of the same object across different views, significantly improving the 3D accuracy of the 6D object poses. This feature also speeds up the labeling process for multi-view data.

2.3.2. Technical Details

The tool is a web-based application that uses Python for the back end and Angular for the front end. The back end runs on a RESTful API and relies on OpenCV and Scipy for all calculations regarding 6D object poses. For all 3D visualizations and interactions, ThreeJS is used. To ensure a perfect match between the provided RGB images and the 3D visualizations, high-quality object models and camera intrinsics and extrinsics are necessary.

2.4. Synthetic Data

Using a Unity project, we generated high-fidelity synthetic data. Real-world conditions were carefully simulated. We adopted camera intrinsics and extrinsics and model objects and carriers. Environment maps of the different real-world lighting conditions were created by combining bracketed images. We used a Rico Theta S 360° camera to capture the environment with different exposures. The resulting images were combined into a single HDRI environment map to light the virtual scene. We mimicked real-world features like scratches and saw patterns by synthesizing textures from real-world images of example objects. These images were relighted and cropped to be used as albedo textures. A normal map was created by estimating normal vectors from these textures. A texture synthesis algorithm was then used to create new variations of the captured textures. This texture synthesis algorithm works in two steps. First, random patches are taken from an example image. Next, to reduce seams between copied patches, pixels search their neighborhood for similar pixels and update their value accordingly. This step is repeated multiple times, reducing the radius of the neighborhood with each turn until the neighborhood consists of a single pixel. This procedure generates unique and realistic appearances for every virtual object. Figure 12 shows a close-up of multiple generated object textures. Our synthetic data generation setup resulted in synthetic images that resemble their real-world counterparts.

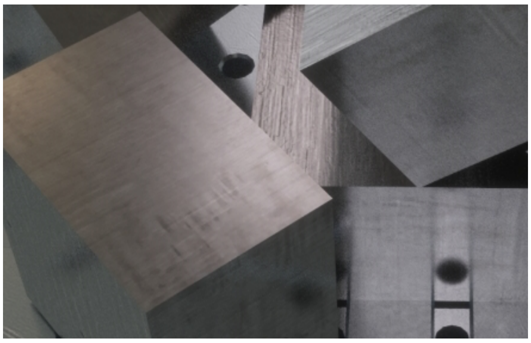


Figure 12. Close-up of a synthetic image. Our texture synthesis algorithm leads to realistic object textures, exhibiting real-world features like scratches and saw patterns. The use of path tracing leads to realistic reflections.

A large amount of additional data were generated by varying object material, composition, and lighting types. Object materials were generated using the texture synthesis algorithm described before, using additional example images. We used Unity's physics engine to generate physically plausible object poses, starting from random initializations. Lighting was varied by randomly selecting an environment map from a predefined set. As we were in total control of the data generation process, labels were free. Generating 1 scene took 2 min using an NVIDIA GeForce 2080 Ti.

3. Results

3.1. Real-World Data

In total, we recorded 600 real-world scenes (Table 3), resulting in 31,200 real-world images (Table 4). Table 3 shows an overview of the variations in object compositions. Table 4 shows the resulting scene variations. All images were manually labeled using the labeling tool presented before. As we carefully calibrated each camera, object labels correspond across images from different viewpoints and for different cameras.

Table 3. Composition variations in real-world data. We used six object types. For each object type, we recorded two compositions with a single instance and eight with multiple instances (homogeneous). In addition, we created 40 compositions with objects of different types (heterogeneous). This resulted in 100 different compositions.

Category	# Variations
Object types	6
Single object	2
Multiple objects (same type)	8
Homogeneous	$6 \times (2+8) = 60$
Heterogeneous	40
Compositions	$60 + 40 = 100$

Table 4. Scene variations in real-world data. All composition variants were recorded for each carrier (3) and lighting condition (2), resulting in 600 scene variations. Using four different cameras, each recording from 13 different viewpoints, resulted in 31,200 images.

Category	# Variations
Compositions	100
Carriers	3
Lighting	2 (light or dark)
Scenes	$100 \times 3 \times 2 = 600$
Camera's	4
Viewpoints	13
Images	600 × 4× 13 = 31,200

3.2. Labeling Tool

To assess the accuracy of our labeling pipeline, we manually labeled ten synthetic scenes for which exact groundtruth poses were available with both multi-view and single-view 6D object poses. We measured pose errors using the Maximum Symmetry-Aware Surface Distance (MSSD):

$$e_{\text{MSSD}} = \min_{\mathbf{S} \in S_O} \max_{\mathbf{x} \in V_O} \|\hat{\mathbf{P}}\,\mathbf{x} - \bar{\mathbf{P}}\mathbf{S}\mathbf{x}\|_2,$$

with $\hat{\mathbf{P}}$ denoting the estimated pose, $\bar{\mathbf{P}}$ the ground truth pose, S_O a set of symmetry transformations, and V_O a set of vertices of the object CAD model. Table 5 shows the results. Labeling multi-view images resulted in labels with significantly lower MSSD error scores. In addition, the speed of labeling increased as multi-view images were labeled together. Using our dataset, with 13 images per scene, led to a 13-fold reduction in labeling time.

Table 5. Labeling accuracy. Labeling multi-view data significantly reduced the MSSD error.

Category	$e_{\mathbf{MSSD}}$
Single-view	4.53
Multi-view	0.267

3.3. Synthetic Data

In total, 42,600 synthetic scenes were generated, resulting in 553,800 synthetic images (Table 4). Table 6 shows the synthetic variations for each real-world scene. Figure 13 shows an example of a real-world image and synthetic variants with different lighting conditions.

Table 6. Synthetic variations of a real-world scene. First, each real-world scene was replicated by mimicking object poses and lighting conditions. Next, lighting conditions were varied while retaining object poses. Next, object poses were varied, while using the original lighting conditions. Finally, both lighting conditions and object poses were jointly varied. Object textures were regenerated for each recording. This resulted in 71 synthetic variations for each real-world scene.

Pose	Lighting	# Variations
Original	Original	1
Original	Randomized	15
Randomized	Original	15
Randomized	Randomized	40
		71

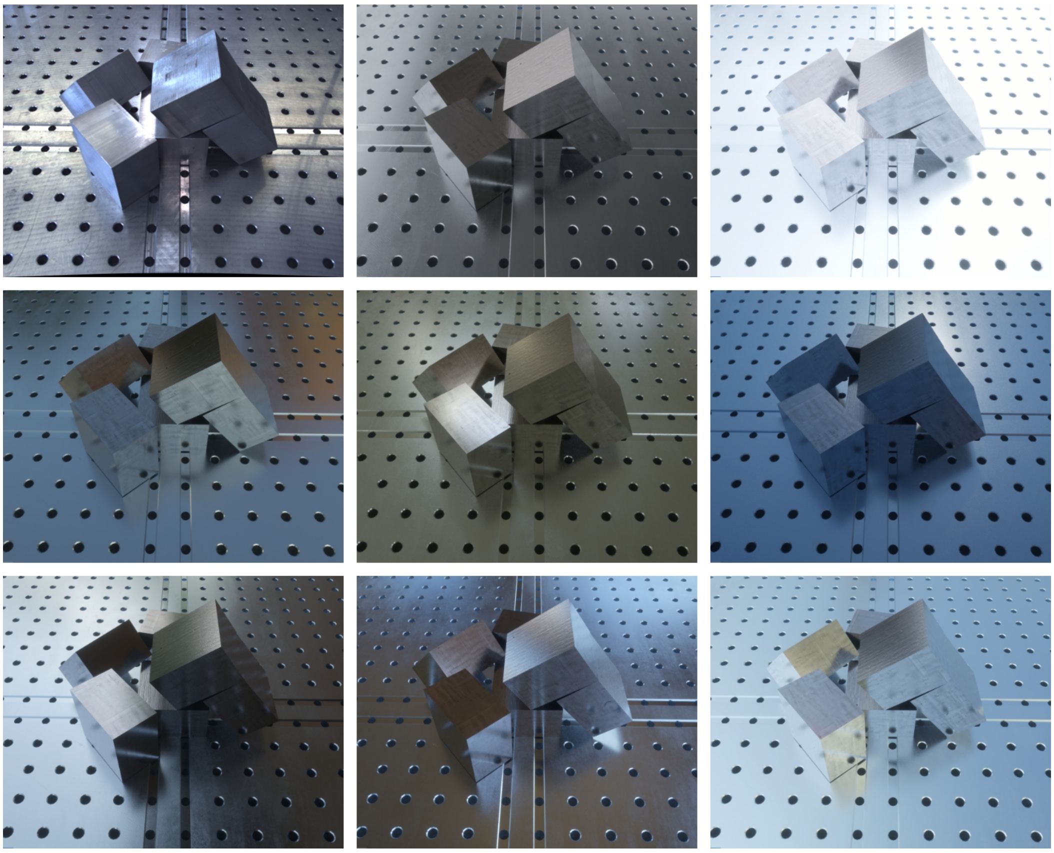


Figure 13. Real-world image (top-left) and corresponding synthetic images with varying lighting conditions.

3.4. Usage

The high fidelity of our synthetic images, their close correspondence with their real-world counterparts, and the controlled variations in the data generation process will be valuable for sim-to-real research. Recently, our dataset was used to train object detection models with synthetic data [33]. The authors trained a model on various sub-sets of our synthetic data to determine which variations improved model performance. They learned that modeling real-world lighting conditions and realistic object poses enhanced model performance significantly.

Moreover, our dataset is highly useful for various computer vision tasks, including 6D pose estimation, object detection, instance segmentation, novel view synthesis, 3D reconstruction, and active perception. In a recent study, our dataset was used for 6D object pose estimation [34]. The authors first trained PVNet [35], a popular 6D object pose estimation method, on our dataset. Although this model works well on many existing pose estimation datasets, it did not perform well on our dataset. The authors then trained a different model that was designed to deal with reflections and to use information from multiple viewpoints. This significantly increased performance. However, there is still room for improvement.

4. Conclusions

We present a diverse dataset of industrial, reflective objects with real-world and synthetic data. Real-world data were obtained by recording multi-view images of scenes with varying object shapes, materials, carriers, compositions, and lighting conditions. This resulted in over 31,200 accurately labeled images using a new public tool that reduces the time and effort needed to label 6D object poses for multi-view data. Synthetic data were obtained by carefully simulating real-world and varying environmental conditions in a controlled and realistic way. This results in over 553,800 synthetic images.

The close resemblance of our synthetic images to their real-world counterparts, along with the controlled variations in the data generation process, make our dataset valuable for sim-to-real research.

Our dataset is also useful for several computer vision tasks, including 6D pose estimation, object detection, instance segmentation, novel view synthesis, 3D reconstruction, and active perception. These tasks are crucial for various industrial applications such as automated assembly, bin picking, quality control, welding, painting, (de-)palletizing, and machine tending.

In future work, we aim to broaden the adoption of our dataset within key academic benchmarks like BOP. This will enhance its visibility and utility for comparative studies. Additionally, we plan to explore methodologies that utilize multi-view data, a relatively under-explored area in current research. Given the complexities in many industrial settings, this multi-view approach holds significant promise for developing more robust and capable solutions. By focusing on these two strategic areas, we intend not only to elevate the dataset's academic prominence but also unlock new potential in practical applications.

Author Contributions: Conceptualization, P.D.R., N.M. and F.w.; methodology: P.D.R., N.M. and F.w.; real-world data collection: P.D.R.; synthetic data creation: P.D.R., S.M. and N.M.; development of labeling tool: P.D.R.; writing—original draft preparation: P.D.R.; writing—review & editing: P.D.R. and F.w.; visualization: P.D.R.; supervision: N.M. and F.w. All authors have read and agreed to the published version of the manuscript.

Funding: The authors disclose the receipt of the following financial support for the research, authorship, and/or publication of this article: this work was supported by VLAIO Baekeland Mandate HBC.2019.2162 and Flanders Make's PILS SBO project (R-9874).

Data Availability Statement: The dataset is available on the project website: https://pderoovere.github.io/dimo (accessed on 30 September 2023) along with a utility script for loading data and information about the labeling tool.

Acknowledgments: We would like to thank Cedric Verschueren and Hube Van Loey for their help in annotating the real-world data and RoboJob NV for their overall support.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

HMLV	High-mix, low-volume
BOP	Benchmark for 6D object pose estimation

Appendix A. Usability Evaluation

The rubric in Table A1 was used to evaluate the ease of use of the labeling tools described in Table 1.

Table A1. Rubric for evaluating the ease of use of 6D object pose labeling tools.

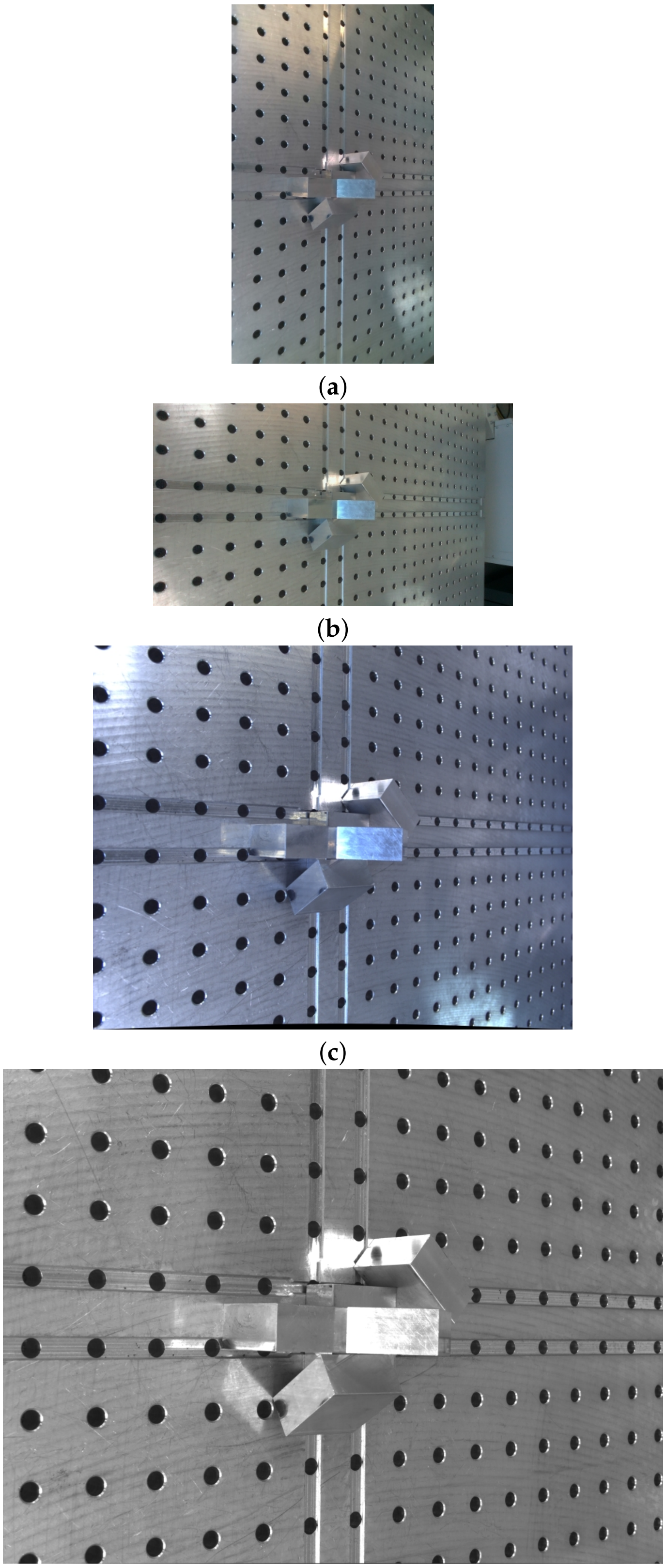
Criteria	Poor *	Adequate **	Good ***	Excellent ****
Initial Pose Labeling	Only a default value is used for the initial pose.	A default value or an estimated initial pose based on image-CAD correspondences is available.	Adds the option to snap to CAD vertices for precise key point identification.	Includes duplicating pose labels for previously labeled objects to efficiently label similar objects in the same plane.
Pose Fine-Tuning	The pose can be changed manually.	Step-wise pose adjustments are available for more refined control.	Incorporates a gizmo control for manipulating an overlay of the CAD file on the image.	Pose fine-tuning is automated (e.g., using ICP).
Visualization	An overlay of the CAD model is shown on the image.	Adds zoom and pan functionalities for the image and the overlay.	Enables changing the opacity level of the overlay for better discernment.	Introduces color overlays for distinguishing between different objects.
UX	Navigation is challenging and confusing with an unattractive or cluttered design. The app experiences noticeable lags and lacks accessibility features.	Navigation is functional with some intuitive elements and an average design. Acceptable response times with limited accessibility features.	Navigation is straightforward with a pleasing design. The app responds well with minor delays and includes basic accessibility features.	Navigation is effortless with immediate access to features and a visually striking, fully responsive design. Comprehensive accessibility features are integrated, including keyboard shortcuts for important actions.

Table A2. Comparison of ease of use [1] of tools for labeling 6D object poses.

Criteria	6D-PAT	Labelfusion	DoPose	Ours
Initial Pose Labeling	**	***	*	****
Pose Fine-Tuning	***	****	**	***
Visualization	***	***	*	****
UX	**	**	*	****
Overall	**	***	*	****

[1] Ease of Use is rated from * to **** where * represents the lowest ease and **** indicates the highest. For a detailed explanation, refer to Table A1.

Appendix B. Examples



(**a**)

(**b**)

(**c**)

(**d**)

Figure A1. Images from real-world cameras scaled to reflect their differing resolutions. The images present challenges due to occlusions and reflections. (**a**) RealSense D415; (**b**) RealSense L515; (**c**) JAI GO-5000-PGE; (**d**) mvBlueFOX3-2124rG-1112.

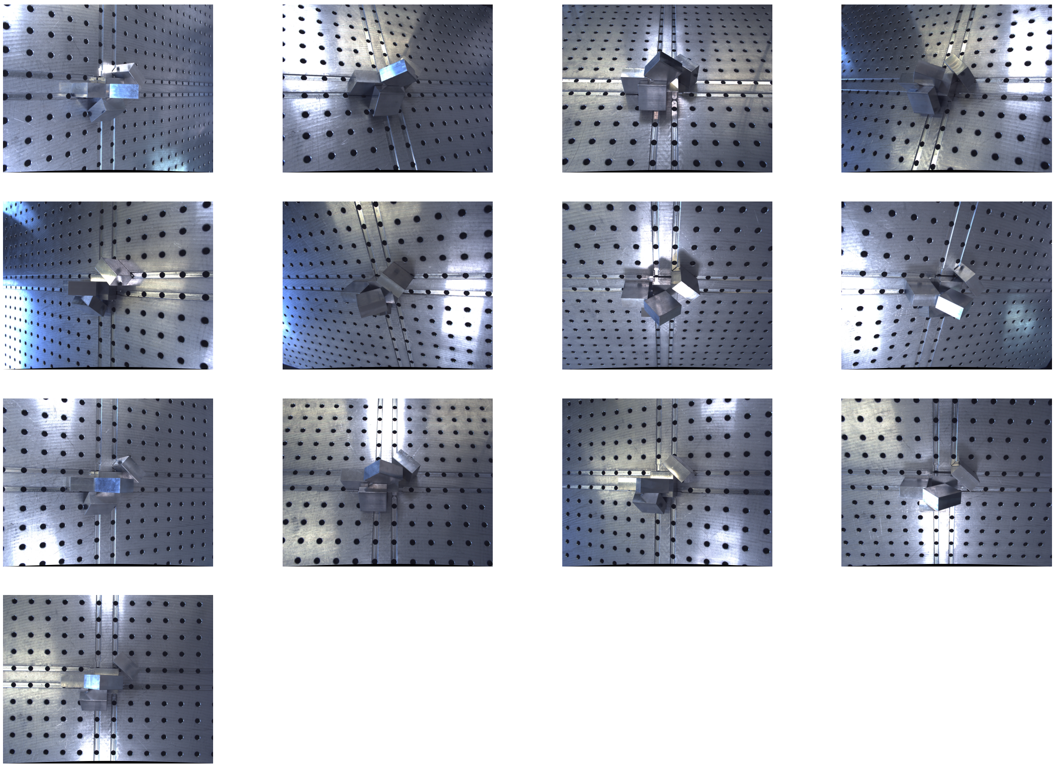


Figure A2. Real-world multi-view images of a scene captured with the JAI GO-5000-PGE camera. The images present challenges due to occlusions and reflections.

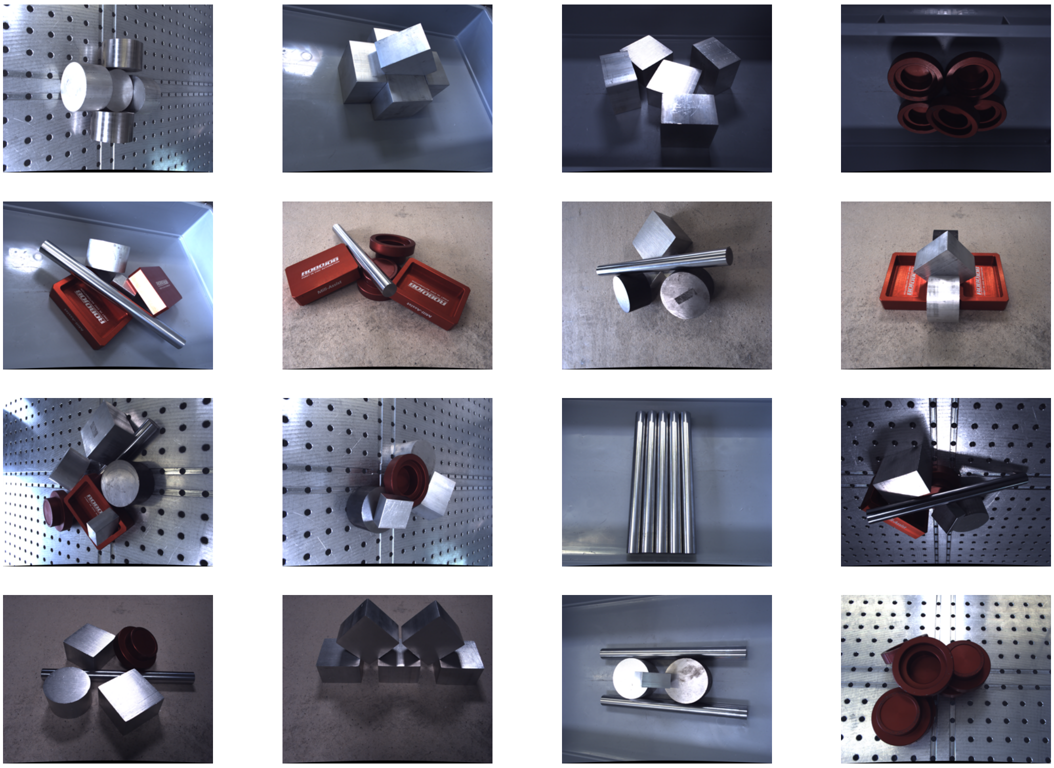


Figure A3. Real-world images captured with the JAI GO-5000-PGE camera. A diverse set of objects was captured in a diverse set of conditions (carrier, lighting, and composition).

References

1. Du, G.; Wang, K.; Lian, S.; Zhao, K. Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: A review. *Artif. Intell. Rev.* **2021**, *54*, 1677–1734. [CrossRef]
2. Directorate-General for Internal Market, Industry, Entrepreneurship and SMEs (European Commission). *A Vision for the European Industry Until 2030: Final Report of the Industry 2030 High Level Industrial Roundtable*; Publications Office of the European Union: Luxembourg, 2019. [CrossRef]
3. Marullo, G.; Tanzi, L.; Piazzolla, P.; Vezzetti, E. 6D object position estimation from 2D images: A literature review. *Multimed. Tools Appl.* **2023**, *82*, 24605–24643. [CrossRef]
4. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009*; IEEE: New York, NY, USA, 2009; pp. 248–255.
5. Lin, T.; Maire, M.; Belongie, S.J.; Bourdev, L.D.; Girshick, R.B.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. *arXiv* **2014**, arXiv:1405.0312.
6. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets Robotics: The KITTI Dataset. *Int. J. Robot. Res.* **2013**, *32*, 1231–1237. [CrossRef]
7. Hodan, T.; Michel, F.; Brachmann, E.; Kehl, W.; Buch, A.G.; Kraft, D.; Drost, B.; Vidal, J.; Ihrke, S.; Zabulis, X.; et al. BOP: Benchmark for 6D Object Pose Estimation. In *Proceedings of the Computer Vision—ECCV 2018—15th European Conference, Munich, Germany, 8–14 September 2018*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Proceedings, Part X—Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2018; Volume 11214, pp. 19–35. [CrossRef]
8. Sundermeyer, M.; Hodaň, T.; Labbe, Y.; Wang, G.; Brachmann, E.; Drost, B.; Rother, C.; Matas, J. Bop challenge 2022 on detection, segmentation and pose estimation of specific rigid objects. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 2784–2793.
9. Denninger, M.; Sundermeyer, M.; Winkelbauer, D.; Zidan, Y.; Olefir, D.; Elbadrawy, M.; Lodhi, A.; Katam, H. BlenderProc. *arXiv* **2019**, arXiv:1911.01911.
10. Kleeberger, K.; Landgraf, C.; Huber, M.F. Large-scale 6D object pose estimation dataset for industrial bin-picking. In *Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019*; IEEE: New York, NY, USA, 2019; pp. 2573–2578.
11. Sünderhauf, N.; Brock, O.; Scheirer, W.J.; Hadsell, R.; Fox, D.; Leitner, J.; Upcroft, B.; Abbeel, P.; Burgard, W.; Milford, M.; et al. The limits and potentials of deep learning for robotics. *Int. J. Robot. Res.* **2018**, *37*, 405–420. [CrossRef]
12. Tobin, J.; Fong, R.; Ray, A.; Schneider, J.; Zaremba, W.; Abbeel, P. Domain randomization for transferring deep neural networks from simulation to the real world. In *Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017, Vancouver, BC, Canada, 24–28 September 2017*; IEEE: New York, NY, USA, 2017; pp. 23–30. [CrossRef]
13. Hodan, T.; Vineet, V.; Gal, R.; Shalev, E.; Hanzelka, J.; Connell, T.; Urbina, P.; Sinha, S.N.; Guenter, B. Photorealistic Image Synthesis for Object Instance Detection. In *Proceedings of the 2019 IEEE International Conference on Image Processing, ICIP 2019, Taipei, Taiwan, 22–25 September 2019*; IEEE: New York, NY, USA, 2019; pp. 66–70. [CrossRef]
14. He, Z.; Li, Q.; Zhao, X.; Wang, J.; Shen, H.; Zhang, S.; Tan, J. ContourPose: Monocular 6-D Pose Estimation Method for Reflective Textureless Metal Parts. *IEEE Trans. Robot.* **2023**, *39*, 4037–4050. [CrossRef]
15. Hodan, T.; Haluza, P.; Obdrzálek, S.; Matas, J.; Lourakis, M.I.A.; Zabulis, X. T-LESS: An RGB-D Dataset for 6D Pose Estimation of Texture-Less Objects. In *Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017, Santa Rosa, CA, USA, 24–31 March 2017*; IEEE Computer Society: New York, NY, USA, 2017; pp. 880–888. [CrossRef]
16. Drost, B.; Ulrich, M.; Bergmann, P.; Härtinger, P.; Steger, C. Introducing MVTec ITODD—A Dataset for 3D Object Recognition in Industry. In *Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2017, Venice, Italy, 22–29 October 2017*; IEEE Computer Society: New York, NY, USA, 2017; pp. 2200–2208. [CrossRef]
17. Byambaa, M.; Koutaki, G.; Choimaa, L. 6D Pose Estimation of Transparent Objects Using Synthetic Data. In *Proceedings of the International Workshop on Frontiers of Computer Vision, Hiroshima, Japan, 21–22 February 2022*; Sumi, K., Na, I.S., Kaneko, N., Eds.; Springer: Cham, Switzerland, 2022; pp. 3–17.
18. Sajjan, S.; Moore, M.; Pan, M.; Nagaraja, G.; Lee, J.; Zeng, A.; Song, S. Clear Grasp: 3D Shape Estimation of Transparent Objects for Manipulation. In *Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020*; IEEE: New York, NY, USA, 2020. [CrossRef]
19. Labbé, Y.; Carpentier, J.; Aubry, M.; Sivic, J. CosyPose: Consistent Multi-view Multi-object 6D Pose Estimation. In *Computer Vision—ECCV 2020*; Springer International Publishing: Cham, Switzerland, 2020; pp. 574–591. [CrossRef]
20. Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A.C.; Lo, W.Y.; et al. Segment anything. *arXiv* **2023**, arXiv:2304.02643.
21. V7 Labs. V7 Labs. 2023. Available online: www.v7labs.com (accessed on 6 December 2023).
22. Labelbox, Inc. Labelbox. 2023. Available online: www.labelbox.com (accessed on 6 December 2023).
23. Scale AI, Inc. Rapid. Scale AI. 2023 Available online: www.scale.com/rapid (accessed on 6 December 2023).
24. SuperAnnotate AI. SuperAnnotate AI. 2023. Available online: www.superannotate.com (accessed on 6 December 2023).
25. Dataloop Ltd. Dataloop. 2023. Available online: www.dataloop.ai (accessed on 6 December 2023).
26. Supervisely. Supervisely. 2023. Available online: www.supervisely.com (accessed on 6 December 2023).

27. Segments.ai. Segments.ai. 2023. Available online: www.segments.ai (accessed on 6 December 2023).
28. Blume, F. 6D-PAT. GitHub Repository. 2023. Available online: https://github.com/florianblume/6d-pat (accessed on 6 December 2023).
29. Gouda, A. Chair of Materials Handling and Warehousing DoPose. GitHub Repository. 2023. Available online: https://github.com/FLW-TUDO/3d_annotation_tool (accessed on 6 December 2023).
30. Marion, P.; Florence, P.R.; Manuelli, L.; Tedrake, R. Label Fusion: A Pipeline for Generating Ground Truth Labels for Real RGBD Data of Cluttered Scenes. In *Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018*; IEEE: New York, NY, USA, 2018; pp. 3235–3242.
31. Tsai, R.Y.; Lenz, R. A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. *IEEE Trans. Robot. Autom.* **1989**, *5*, 345–358. [CrossRef]
32. Bradski, G. The OpenCV Library. *Dr. Dobbs J. Softw. Tools* **2000**, *25*.
33. Vanherle, B.; Moonen, S.; Reeth, F.V.; Michiels, N. Analysis of Training Object Detection Models with Synthetic Data. In *Proceedings of the 33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, 21–24 November 2022*; BMVA Press: London, UK, 2022.
34. De Roovere, P.; Daems, R.; Croenen, J.; Bourgana, T.; de Hoog, J.; wyffels, F. CenDerNet: Center and Curvature Representations for Render-and-Compare 6D Pose Estimation. In *Proceedings of the Computer Vision–ECCV 2022 Workshops, Tel Aviv, Israel, 23–27 October 2022*; Karlinsky, L., Michaeli, T., Nishino, K., Eds.; Springer: Cham, Switzerland, 2023; pp. 97–111.
35. Peng, S.; Liu, Y.; Huang, Q.; Zhou, X.; Bao, H. Pvnet: Pixel-wise voting network for 6DoF pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4561–4570.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.