Dataset of Industrial Metal Objects

Peter De Roovere¹, Steven Moonen², Nick Michiels², Francis wyffels¹

Abstract
We present a diverse dataset of industrial metal objects. These objects are symmetric, textureless and highly reflective, leading to challenging conditions not captured in existing datasets. Our dataset contains both real-world and synthetic multi-view RGB images with 6D object pose labels. Real-world data is obtained by recording multi-view images of scenes with varying object shapes, materials, carriers, compositions and lighting conditions. This results in over 30,000 images, accurately labelled using a new public tool. Synthetic data is obtained by carefully simulating real-world conditions and varying them in a controlled and realistic way. This leads to over 500,000 synthetic images. The close correspondence between synthetic and real-world data, and controlled variations, will facilitate sim-to-real research. Our dataset’s size and challenging nature will facilitate research on various computer vision tasks involving reflective materials. The dataset and accompanying resources are made available on the project website https://pderoovere.github.io/dimo.

Keywords
sim-to-real, robotic manipulation, reflective materials, 6D object pose estimation, industrial robotics

Introduction
Identifying objects and estimating their 6D poses are crucial aspects of many robotics applications (Du et al. 2021). Manipulating physical objects requires knowledge about their position and orientation. Estimating these poses from image data is a longstanding problem, which remains an active area of research (Rad and Lepetit 2017; Xiang et al. 2018; Tekin et al. 2018; Li et al. 2020; Zhou et al. 2019).

Existing object pose estimation datasets fail to capture important aspects of real-world industrial use-cases. The Benchmark for 6D Object Pose Estimation (BOP) (Hodan et al. 2018) unifies 12 datasets, of which the majority focus on household objects. These objects are non-symmetric, with rich textures that lead to distinct visual features. In contrast, most industrial objects are symmetric and textureless. Material properties such as colour and reflectance dominate their visual appearance. Other datasets do incorporate textureless industrial objects (Hodan et al. 2017; Drost et al. 2017) but focus on objects with limited reflectivity. However, reflective materials are widespread. Most high precision machine components, for example, are made of shiny metals. Existing depth sensors are unable to capture these parts correctly. 6D pose estimation from RGB images is thus a fundamental problem for reflective objects. This task is challenging, as these objects literally reflect their surroundings, ambiguating their appearance. This leads to widely varying visual appearances. A diverse dataset is needed to develop and evaluate new solutions.

As object pose labels are costly to obtain, synthetic data generation is a promising approach to creating diverse pose estimation datasets. Recent advances in computer graphics make it feasible to generate large amounts of labelled data. This has been applied to pose estimation datasets (Denninger et al. 2019; Kleeberger et al. 2019). However, the resulting images 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

¹IDLab-AIRO – Ghent University – imec
²Hasselt University – tUL – Flanders Make, Expertise Centre for Digital Media

Corresponding author:
Peter De Roovere. IDLab-AIRO – Ghent University – imec, iGent Tower - Technologiepark-Zwijnaarde 126, B-9052 Ghent, Belgium
Email: peter.deroovere@ugent.be
hamper model performance – the so-called reality gap (Sünderhauf et al. 2018). A proposed solution to the reality gap is domain randomization (Tobin et al. 2017), in which parts of the data generation process are randomized. With enough variability, real-world data may then appear to be just 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 (Hodan et al. 2019). This problem is amplified for reflective objects, as their visual appearance is highly dependent on their surroundings.

We present a diverse dataset of industrial, reflective objects with real-world data and corresponding synthetic data with controlled variations. Scenes with varying object shapes, materials, carriers, compositions, and lighting are recorded with different cameras and from multiple viewpoints. Our dataset consists of more than 30,000 real-world images labelled using a new open source labeling tool. In addition, we generate over 500,000 synthetic images by carefully mimicking real-world conditions and adding controlled variations in object materials, compositions and lighting. We use modern physically-based rendering techniques to simulate realistic reflections and procedurally generate real-world artefacts like scratches.

Our dataset will be of value to many different research problems. The close correspondence between synthetic and real-world data, along with controlled variations, will facilitate sim-to-real research. While many existing datasets are insufficiently large (Du et al. 2021), the large number of labelled images presented in this work can be used for training robust models. Our dataset enables research on many important computer vision tasks involving reflective objects. We focus our discussion on 6D object pose estimation, which is at the root of many robotics use cases. However, our dataset also applies to object detection, instance segmentation, novel view synthesis, 3D reconstruction and active perception.

Methodology

We built a setup for easily collecting and labelling real-world data and created a system for generating synthetic data by simulating real-world conditions.

Real-world data

We use different camera sensors (Table 1) mounted on an industrial robot, to capture multi-view images of diverse scenes. A custom-designed end-effector is used to attach all cameras to a Fanuc M20ia robot, as shown in Figure 3. The robot, with repeatability of 0.1mm, is used to accurately change camera viewpoints. We further stabilize camera poses by cancelling backlash using the Fanuc IRVBKLSH instruction. We calibrate all cameras using ChArUco targets, undistort all images and perform hand-eye calibration (Tsai and Lenz 1989). We use high-precision aluminium-LDPE calibration targets with DICT_5x5 dictionary codes and 15mm wide checkers. ChArUco detection and calibration are implemented using OpenCV (Bradski 2000). Each scene is captured with every camera from 13 different viewpoints. These viewpoints are spread out on a hemisphere and oriented towards the center of the object carrier, as shown in Figure 4.

Figure 2. Our setup for collecting real-world images. We capture multi-view images of diverse scenes using different camera sensors mounted on an industrial robot.

Figure 3. Different camera sensors are attached to the end-effector of an industrial robot.

Diverse scenes are created by varying object shapes, materials, carriers, compositions, and lighting. We use 6 object categories, shown in Figure 5, with different shapes and material properties. Most objects (5a, 5b, 5c, 5d) are highly reflective and exhibit real-world artefacts like scratches and saw patterns. We also include less reflective (5e, 5f) and textured objects (5e) for comparison. We use 3 object carriers, shown in Figure 6. Parts are stacked in various compositions with different levels of occlusion. Lighting conditions are varied by controlling Nanguan Luxpad 43 lights. These variations lead to 600 scenes (Table 2) and 31,200 images (Table 3).

We carefully label all real-world data using a new open source tool. Precise calibration and consistent camera viewpoints allow joint labelling of all images of one scene. We developed a new open source tool to facilitate this task, available at https://github.com/
around and oriented towards where objects are placed. These viewpoints were spread out on a hemisphere, centred around and oriented towards where objects are placed. These viewpoints were spread out on a hemisphere, centred around and oriented towards where objects are placed.

De Roovere, Moonen, Michiels and wyffels

Table 1. The different cameras used for capturing real-world image data.

| Camera            | Lens                  | Type       | Resolution   |
|-------------------|-----------------------|------------|--------------|
| JAI GO-5000-PGE   | KOWA LM12HC-V         | RGB        | 2560 × 2048  |
| mvBlueFOX3-2124rG-1112 | MV-O1218-10M-KO     | Grayscale  | 4064 × 3044  |
| RealSense L515    |                       | RGB, LiDAR | 1920 × 1080  |
| RealSense D415    |                       | RGB, Active IR Stereo | 1920 × 1080 |

Figure 4. Top-view visualisation of all 13 camera viewpoints. These viewpoints were spread out on a hemisphere, centred around and oriented towards where objects are placed.

Table 2. Composition variations in real-world data. We use 6 object types. For each object type, we record 2 compositions with a single instance and 8 compositions with multiple instances (homogeneous). In addition, we create 40 compositions with objects of different types (heterogeneous). This leads to 100 different compositions.

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

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

| Category       | # Variations |
|----------------|--------------|
| Compositions   | 100          |
| Carriers       | 3            |
| Lighting       | 2 (light or dark) |
| Scenes         | 100 × 3 × 2 = 600 |
| Camera’s       | 4            |
| Viewpoints     | 13           |
| Images         | 600 × 4 × 13 = 31,200 |

All poses are defined relative to a global coordinate system, ensuring multi-view consistency. To assess the accuracy of our labelling pipeline, we manually label 10 synthetic scenes, where exact ground-truth poses are available. We measure pose errors using Maximum Symmetry-Aware Surface Distance (MSSD):

\[ e_{\text{MSSD}} = \min_{S_O} \max_{x} \|P_x - \hat{P}_S\|_2, \]

with \( P \) the estimated pose, \( \hat{P} \) the ground truth pose, \( S_O \) a set of symmetry transformations, and \( V_O \) a set of vertices of the object CAD model. We obtain an \( e_{\text{MSSD}} \) of 0.267 mm, demonstrating highly accurate manual annotations.

Synthetic data

Using a Unity project, we generate high-fidelity synthetic data. Real-world conditions are carefully simulated. We adopt camera intrinsics and extrinsics, and model objects and carriers. Environment maps of the different real-world lighting conditions are created by combining bracketed images. We use a Rico Theta S 360° camera to capture the environment with different exposures. The resulting images are combined into a single HDRI environment map used to light the virtual scene. We mimic real-world artefacts like scratches and saw patterns by synthesizing textures from real-world images of example objects. These images are relighted and cropped so they can be used as albedo textures. A normal map is created by estimating normal vectors from these textures. A texture synthesis algorithm is 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 neighbourhood for similar pixels and update their value accordingly. This step is repeated multiple times, reducing the radius of the neighbourhood with each turn, until the neighbourhood consists of a single pixel. This procedure generates unique and realistic appearances for every virtual object. Figure 7 shows a close-up of multiple generated object textures. Our synthetic data generation setup leads to synthetic images that strongly resemble their real-world counterparts.

A large amount of additional data is generated by varying object materials, compositions and lighting. Object materials are generated using the texture synthesis algorithm described before, using additional example images. We use Unity’s physics engine to generate physically plausible object poses, starting from random initialisations. Lighting is varied by randomly selecting an environment map from a predefined set. Table 4 gives an overview of all variations.

Combining these variations leads to 42,600 scenes. To focus our resources on generating environmental variations, we generate synthetic data for 1 camera type (JAI GO-5000-PGE). In total, 553,800 images are generated.
Figure 5. Object categories. We use objects of 6 categories, with different shapes and material properties. Most objects (5a, 5b, 5c, 5d) are highly reflective and exhibit real-world artefacts like scratches. For comparison, we also include less reflective and textured objects (5e, 5f).

(a) Cylinder  
(b) Small block  
(c) Large block  
(d) Shaft  
(e) Red block  
(f) Red cylinder

Figure 6. Objects are placed on 3 different types of carriers.

(a) Pallet  
(b) Bin  
(c) Cardboard

Figure 7. Close up of a synthetic image. Our texture synthesis algorithm leads to realistic object textures, exhibiting real-world artefacts like scratches and saw patterns. The use of path tracing leads to realistic reflections.

Table 4. Synthetic variations of a real-world scene. First, each real-world scene is replicated by mimicking object poses and lighting conditions. Next, lighting conditions are varied while retaining object poses. Next, object poses are varied, while using the original lighting conditions. Finally, both lighting conditions and object poses are jointly varied. Object textures are regenerated for each recording. This leads to 71 synthetic variations for each real-world scene.

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

Dataset

As we are in total control of the data generation process, labels are free. Generating 1 scene takes 2 minutes using a NVIDIA GeForce 2080 Ti.

Our dataset contains 31,200 images of 600 real-world scenes and 553,800 images of 42,600 synthetic scenes, stored in a unified format. In addition to camera intrinsics and extrinsics, object models and object pose labels, we store information about the data generation process.
Format

We adopt and extend the BOP format (Hodan et al. 2018). The folder structure is shown in Figure 8. All distances are recorded in mm. Matrices and vectors are stored as flattened lists.

Figure 8. Folder structure of our dataset. Our dataset contains a subfolder with object CAD models and subfolders for each camera (real and virtual). These camera folders contain subfolders for each scene. Each scene folder contains a subfolder with multi-view images and json files with accompanying information. We adopt the BOP format and extend it (marked in red).

models contains object CAD models, stored in the PLY format. models_info.json contains additional information about the size and symmetries of each object, as shown in Listing 1. Each continuous symmetry is described by an axis and offset value, each discrete symmetry by a transformation matrix. construction contains technical drawings that can be used for manufacturing the parts.

Scenes are organised by camera type (both real and virtual) and identified by a unique id. Each scene is available for each camera. For real cameras, 13 images are available for each scene – one for each viewpoint. For virtual cameras, 71 scenes are available for each real-world scene – 1 mimicking the corresponding real conditions and 70 generated with varying conditions.

scene_camera.json contains camera intrinsics and extrinsics for each image, as shown in Listing 2.

Object poses are recorded in scene_gt.json and scene_gt_world.json, as shown in Listing 3 and Listing 4. For each image, scene_gt.json contains objects 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 camera viewpoint. This information allows selecting subsets on the data based on specific variations.

Project website

The dataset is made available on the project website https://pderoovere.github.io/dimo, along with a utility script for loading data and information about the labeling tool.
Figure 9. Real-world image (left) and corresponding synthetic images with varying lighting conditions.

Listing 4: `scene_gt_world.json` contains object poses (relative to a global world reference). These poses are valid for all images.

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

Listing 5: `scene_info.json` contains information about the data generation process for each image.

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

Conclusion

We present a diverse dataset of industrial, reflective objects with both real-world and synthetic data. Real-world data is obtained by recording multi-view images of scenes with varying object shapes, materials, carriers, compositions and lighting conditions. This leads to over 31,200 images, accurately labelled using a new public tool. Synthetic data is obtained by carefully simulating real-world conditions and varying environmental conditions in a controlled and realistic way. This results in over 553,800 synthetic images. We hope our dataset can facilitate both sim-to-real research and research on computer vision tasks involving reflective objects.

Acknowledgements

We thank Cedric Verschueren and Hube Van Loey for their help in annotating real-world data and RoboJob NV for their overall support.

Declaration of conflicting interests

The Authors declare that there is no conflict of interest.

Funding

The authors disclosed 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.

References

Bradski G (2000) The OpenCV Library. *Dr. Dobb’s Journal of Software Tools*.

Denninger M, Sundermeyer M, Winkelbauer D, Zidan Y, Olefir D, Elbadrawy M, Lodhi A and Katam H (2019) Blenderproc. *CoRR* abs/1911.01911. URL http://arxiv.org/abs/1911.01911.

Drost B, Ulrich M, Bergmann P, Härtinger P and Steger C (2017) Introducing mvtec ITOOD - A dataset for 3d object recognition in industry. In: 2017 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2017, Venice, Italy, October 22-29, 2017. IEEE Computer Society, pp. 2200–2208. DOI:10.1109/ICCVW.2017.257. URL https://doi.org/10.1109/ICCVW.2017.257.

Du G, Wang K, Lian S and Zhao K (2021) Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review. *Artificial Intelligence Review* 54(3): 1677–1734. DOI:10.1007/s10462-020-09888-5. URL https://doi.org/10.1007/s10462-020-09888-5.

Hodan T, Haluza P, Obdrzálek S, Matas J, Lourakis MIA and Zabulis X (2017) T-LESS: an RGB-D dataset for 6d pose estimation of texture-less objects. In: Ferrarì V, Hebert M, Sminchisescu C and Weiss Y (eds.) Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part X, Lecture Notes in Computer Science, volume 11214. Springer, pp. 880–888. DOI:10.1007/978-3-030-01249-6_2. URL https://doi.org/10.1007/978-3-030-01249-6_2.

Hodan T, Michel F, Brachmann E, Kehl W, Buch AG, Kraft D, Drost B, Vidal J, Ihrke S, Zabulis X, Sahin C, Manhardt F, Tombari F, Kim T, Matas J and Rother C (2018) BOP: benchmark for 6d object pose estimation. In: Ferrari V, Hebert M, Sminchisescu C and Weiss Y (eds.) Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part X, Lecture Notes in Computer Science, volume 11214. Springer, pp. 19–35. DOI:10.1007/978-3-030-01249-6_2. URL https://doi.org/10.1007/978-3-030-01249-6_2.

Hodan T, Vineet V, Gal R, Shalev E, Hanzelka J, Connell T, Urbina P, Sinha SN and Guenter B (2019) Photorealistic image synthesis for object instance detection. In: 2019 IEEE International Conference on Image Processing, ICIP 2019, Taipei, Taiwan, September 22-25, 2019. IEEE, pp. 66–70. DOI: 10.1109/ICIP.2019.8803821. URL https://doi.org/10.1109/ICIP.2019.8803821.
Kleeberger K, Landgraf C and Huber MF (2019) Large-scale 6d object pose estimation dataset for industrial bin-picking. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 2573–2578.

Li Y, Wang G, Ji X, Xiang Y and Fox D (2020) Deepim: Deep iterative matching for 6d pose estimation. International Journal of Computer Vision 128(3): 657–678. DOI:10.1007/s11263-019-01250-9. URL https://doi.org/10.1007/s11263-019-01250-9.

Rad M and Lepetit V (2017) BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. IEEE Computer Society, pp. 3848–3856. DOI:10.1109/ICCV.2017.413. URL https://doi.org/10.1109/ICCV.2017.413.

Sünderhauf N, Brock O, Scheirer WJ, Hadsell R, Fox D, Leitner J, Upcroft B, Abbeel P, Burgard W, Milford M and Corke P (2018) The limits and potentials of deep learning for robotics. The International Journal of Robotics Research 37(4-5): 405–420. DOI:10.1177/0278364918770733. URL https://doi.org/10.1177/0278364918770733.

Tekin B, Sinha SN and Fua P (2018) Real-time seamless single shot 6d object pose prediction. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. Computer Vision Foundation / IEEE Computer Society, pp. 292–301. DOI:10.1109/CVPR.2018.00038. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Tekin_Real-Time_Seamless_Single.CVPR_2018_paper.html.

Tobin J, Fong R, Ray A, Schneider J, Zaremba W and Abbeel P (2017) Domain randomization for transferring deep neural networks from simulation to the real world. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017, Vancouver, BC, Canada, September 24-28, 2017. IEEE, pp. 23–30. DOI:10.1109/IROS.2017.8202133. URL https://doi.org/10.1109/IROS.2017.8202133.

Tsai RY and Lenz R (1989) A new technique for fully autonomous and efficient 3d robotics hand/eye calibration. IEEE Transactions on Robotics and Automation 5(3): 345–358. DOI:10.1109/70.34770. URL https://doi.org/10.1109/70.34770.

Xiang Y, Schmidt T, Narayanan V and Fox D (2018) Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. Robotics: Science and Systems (RSS).

Zhou X, Wang D and Krähenbühl P (2019) Objects as points. CoRR abs/1904.07850. URL http://arxiv.org/abs/1904.07850.