# Introducing MVTec ITODD - A Dataset for 3D Object Recognition in Industry

Bertram Drost Markus Ulrich Paul Bergmann Philipp Härtinger
Carsten Steger

MVTec Software GmbH
Munich, Germany
http://www.mvtec.com

## Abstract

*We introduce the MVTec Industrial 3D Object Detection Dataset (MVTec ITODD), a public dataset for 3D object detection and pose estimation with a strong focus on objects, settings, and requirements that are realistic for industrial setups. Contrary to other 3D object detection datasets that often represent scenarios from everyday life or mobile robotic environments, our setup models industrial bin picking and object inspection tasks that often face different challenges. Additionally, the evaluation citeria are focused on practical aspects, such as runtimes, memory consumption, useful correctness measurements, and accuracy. The dataset contains 28 objects with different characteristics, arranged in over 800 scenes and labeled with around 3500 rigid 3D transformations of the object instances as ground truth. Two industrial 3D sensors and three high-resolution grayscale cameras observe the scene from different angles, allowing to evaluate methods that operate on a variety of different modalities. We initially evaluate 5 different methods on the dataset. Even though some show good results, there is plenty of room for improvement. The dataset and the results are publicly available*[1]*, and we invite others to submit results for evaluation and for optional inclusion in the result lists on the dataset's website.*

## 1. Introduction

Public datasets are a vital tool for the computer and machine vision research community. For researchers, they allow a fair and easy comparison with prior art without the need to either acquire one's own dataset or to evaluate all prior art methods oneself. For users, datasets help to quickly get an overview over the state of the art in a particular field.

As large-scale datasets become increasingly available, evaluation against them and obtaining reasonable results becomes increasingly important for the publication of new methods. Therefore, indirectly, datasets and their evaluation criteria can steer the direction of research and shape the requirements for new methods. It is thus even more important to have datasets that represent realistic scenarios, with evaluation criteria that focus not only on overall performance, but also on practical issues such as parameter selection and computational costs.

Quite often, the objects and setups of previous datasets for 3D object detection model environments from households, offices, or mobile robot applications such as warehouse navigation. While these scenarios are important from both a research and application point of view, we found that industrial applications, such as bin picking or surface and defect inspection, have quite different characteristics that are not modeled by the existing datasets. This includes different 3D shapes, different kinds of sensors and modalities, and different kinds of object placements. As a result, methods that perform well on existing datasets sometimes show quite different results when applied to industrial scenarios.

Because of the above-mentioned shortcomings, we introduce a new dataset, the *MVTec Industrial 3D Object Detection Dataset* for the detection and pose estimation of 3D objects, which strongly focuses on industrial scenarios. The dataset contains 28 rigid objects with different shapes and surface characteristics, arranged in over 800 scenes, labeled with their rigid 3D transformation as ground truth. The scenes are observed by two industrial 3D sensors and three grayscale cameras, allowing to evaluate methods that work on 3D, image, or combined modalities. Grayscale cameras were chosen since they are much more prominent in industrial setups. The objects sometimes are observed alone and sometimes in a heap to simulate bin picking.

For the evaluation procedure, we focus on properties that are important for practical applications. This includes comparing the full 3D rigid transformation instead of just a bounding box, in a symmetry-aware manner, as well as explicitly including computational costs in the form of train-

[1] www.mvtec.com/company/research/datasets/

ing runtime, model size, detection runtime, and memory consumption.

The dataset is available for download. While some of the ground truth transformations are also available, most are not made publicly available to prevent overfitting methods with excessive parameter tuning. For evaluation, the results can be uploaded and optionally be included in the result list of the website.

## 2. Related Work

Several datasets for 3D object detection were introduced in the past. For a comprehensive review over RGB-D related datasets, please refer to the work of Firman [5]. Another discussion of a subset of those datasets that are especially relevant to 3D pose estimation can be found in the Work of Hodaň *et al.* [7].

Since the introduction of the Primesense sensor family, especially in the form of Microsoft's Kinect v1 and Asus Xtion, several datasets were acquired using these sensors. We belive that even though this sensor class allows an inexpensive and rapid acquisition of RGB-D data and was important for the progress of 3D and multimodal object detectors, its characteristics are less relevant for industrial scenarios, where typically different kinds of 3D sensors are used. Additionally, the RGB-camera uses a Bayer filter, which makes accurate and sub-pixel precise edge extraction difficult due to demosaicing effects.

Recently, Hodaň *et al.* [7] introduced the T-LESS dataset, a challenging dataset of textureless objects, arranged in close proximity, and acquired with a Primesense and an RGB sensor. T-LESS has a similar focus as the dataset introduced in this work and is similar in design and evaluation. Contrary to it, our dataset features objects with wider characteristics (especially regarding planarity, size and complexity) and sensors with a stronger industrial focus.

## 3. The MVTec ITODD Dataset

The overall target was to realistically cover as many applications as possible. For this, multiple sensors and objects were selected, and objects were arranged in different ways to cover single-instance (conveyor belt, surface inspection), multiple-instance (bin picking), and cluttered scenarios.

**Sensors** Each scene is observed by two industrial stereo 3D cameras and three grayscale cameras. All sensors were arranged such that their field of view was approximately the same and calibrated to obtain their intrinsic parameters as well as their relative poses.

- **High-Quality 3D**: A multi-shot, wide-baseline 3D stereo sensor, providing a range (Z) image, X and Y images, as well as a grayscale image with the same viewpoint as the range image. The sensor uses multiple random projected patterns and reconstructs the scene using a spacetime stereo approach with an accuracy of around 100 $\mu$m.

- **Low-Quality 3D**: Similar to the High-Quality 3D sensor, but with a shorter baseline, a wider field of view, and fewer shots per scene. Because of this, the reconstruction is noisier, with an accuracy of around 1–2 mm. While data of higher quality is always desirable, economic constraints often lead to setups where compromises between cost and data quality must be made.

- **Cameras**: Three high-resolution cameras ($\approx$ 8 MP, $f = 50\text{mm}$) capturing grayscale images. Each scene was captured twice, once with and once without projecting a random pattern. The images with the projected patterns can be used for stereo reconstruction.

Fig. 1 shows the images of an example scene.

Methods can use any combination of the sensors. This allows to evaluate methods that operate on intensity images, on range data, on 3D point clouds, or on multimodal data alike.

**Calibration** The two 3D sensors were factory-calibrated. Additionally, the two 3D sensors and the three grayscale cameras were calibrated as described in [11], yielding a calibration error of less than 0.2 px.

**Objects** A total of 28 objects were selected, with diameters ranging from 2.4 cm to 27 cm. The objects were selected such that they cover a range of different values w.r.t. surface reflectance, symmetry, complexity, flatness, detail, compactness, and size. Fig. 3 shows the used objects, along with their names. Tab. 1 lists some of the key properties of the different objects. Multiple instances of each object are available, and for each object, scenes with only a single and scenes with multiple instances are available. For all objects, manually created CAD models are available for training the detection methods. Note that the dataset does not provide any other training data in form of range or intensity images.

**Acquisition Protocol** The objects were captured in three types of scenes: scenes containing only a single instance of the object, without clutter; scenes containing multiple instances of the object, without additional clutter; and scenes containing both multiple instances of the target object and clutter.

Each scene was acquired once with each of the 3D sensors, and twice with each of the grayscale cameras: once

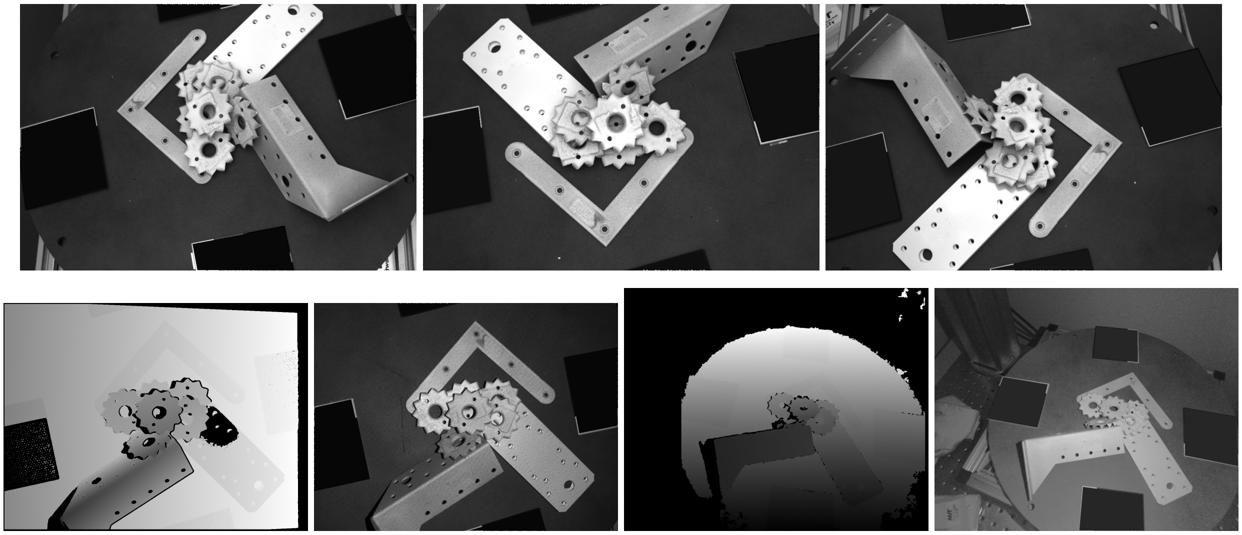


Figure 1. Example scene of the dataset from all sensors. Top row: Grayscale cameras. Bottom row: Z and grayscale image of the High-Quality (left) and Low-Quality (right) 3D sensor.

| **Model** | **Diameter [cm]** | **Symmetries** | **No. scenes** | **No. instances** | **Size ratio** |
|---|---|---|---|---|---|
| adapter_plate_square | 6.4 | no | 27 | 87 | 0.35 |
| adapter_plate_triangular | 5.1 | yes | 30 | 117 | 0.31 |
| box | 14.2 | yes | 25 | 75 | 0.49 |
| bracket_big | 13.9 | yes | 48 | 161 | 0.72 |
| bracket_planar | 15.9 | yes | 39 | 177 | 0.02 |
| bracket_screw | 8.5 | no | 30 | 132 | 0.37 |
| cap | 3.9 | yes | 33 | 180 | 0.93 |
| car_rim | 6.9 | yes | 34 | 131 | 0.59 |
| clamp_big | 9.5 | yes | 27 | 63 | 0.49 |
| clamp_small | 5.6 | no | 21 | 144 | 0.30 |
| connector_planar | 13.8 | yes | 27 | 75 | 0.02 |
| cylinder | 10.8 | yes | 18 | 105 | 0.40 |
| engine_part_bearing | 12.8 | no | 27 | 72 | 0.41 |
| engine_part_cooler_round | 10.3 | yes | 36 | 132 | 0.47 |
| engine_part_cooler_square | 11.4 | no | 33 | 96 | 0.83 |
| engine_part_cover | 19.3 | no | 21 | 30 | 0.35 |
| filter | 7.8 | yes | 30 | 72 | 0.67 |
| fuse | 10.8 | yes | 35 | 100 | 0.52 |
| handle | 12.1 | yes | 30 | 177 | 0.08 |
| injection_pump | 12.2 | no | 30 | 72 | 0.39 |
| multi_bracket | 17.1 | no | 39 | 129 | 0.37 |
| punched_rail | 26.7 | no | 23 | 65 | 0.08 |
| screw | 5.7 | yes | 9 | 48 | 0.30 |
| screw_black | 6.5 | yes | 24 | 105 | 0.50 |
| star | 4.9 | yes | 54 | 381 | 0.12 |
| tee_connector | 6.7 | no | 39 | 186 | 0.39 |
| thread | 5.6 | yes | 24 | 72 | 0.38 |
| washer | 2.4 | yes | 6 | 306 | 0.08 |

Table 1. **Object list with properties**. A list of the 28 objects in the dataset, along with some of their properties. **Size ratio** is the ratio of the smallest to the largest side of an axis-aligned bounding box around the model, indicating the planarity of the object.

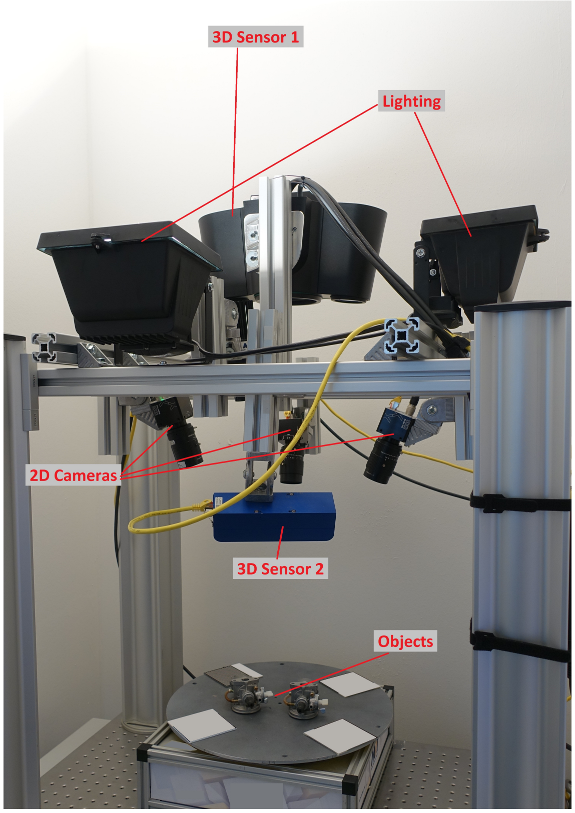


Figure 2. **Annotated picture of the setup**. The top shows the two 3D sensors and the three grayscale cameras that were used for acquisition. The sensors were static and their relative position calibrated. Below, the object placement area is visible. A calibrated turn table was used to transfer the ground truth between scenes by using a total of three turntable rotations per placement.

with and once without a random projected pattern. The objects were arranged on a turntable with calibrated movement. Multiple scenes were acquired for each arrangement by turning the table. This allowed the transfer of ground truth between the rotations.

**Ground Truth** The ground truth was labeled using a semi-manual approach based on the 3D data of the high-quality 3D sensor. Each object instance was approximately segmented by hand, followed by several iterations of manually running ICP[2], tuning its parameters and refining the start pose were performed until both a good score and a visually correct result was obtained. The corresponding ground truth poses were transfered to the scenes obtained by rotating the turntable by using the calibrated turn table positions.

## 4. Evaluation Criteria

**Pose** Industrial manipulation and inspection tasks usually require an exact rigid 3D transformation between the scene and the model. In order to evaluate the practical usefulness of the results, we refrain from using bounding box or surface overlaps as correctness measure and use a pose-based evaluation instead. When comparing a detected pose with a ground truth pose, we use the maximum distance a point on the model surface is away from its ground truth location, normalized by the size of the model.

Formally, given a model represented by a set of points $M \subset \Re^3$ and its diameter $\operatorname{diam}(M) = \max_{v_1,v_2 \in M} |v_1 - v_2|_2$, two transformations $T_1$ and $T_2$ have the distance

$$d^P(T_1, T_2) = \frac{\max_{x \in M} |T_1 x - T_2 x|_2}{\operatorname{diam}(M)} \tag{1}$$

This is similar to the average distance measure in [6], but invariant against different samplings and internal complexity of the model. Due to the normalization, it is also invariant w.r.t. scalings and model size, allowing a comparison of the quality of matches between different objects.

We additionally measure the accuracy of the translation of the model's center point $c_M$ as

$$d^T(T_1, T_2) = |T_1 c_M - T_2 c_M|_2 \tag{2}$$

and the error in the rotation as

$$d^R(T_1, T_2) = \angle(T_1^{-1} T_2) \tag{3}$$

where $\angle(T)$ is the rotation angle of the rotational part of $T$.

Different applications have different requirements w.r.t the accuracy of the detected poses. For example, surface defect detection requires a very accurate pose, while grasping an object with a vacuum suction actuator might work even if the pose is somewhat off. To account for this, we use different thresholds of $d^P$ when classifying the correctness of results.

**Symmetries** Since some of the objects exhibit strong rotational or discrete symmetries that the object detection methods cannot possibly detect, we exclude such symmetries from the evaluation. If a model $M$ has a set of symmetry transformations $S_M$, with $I \in S_M$ and

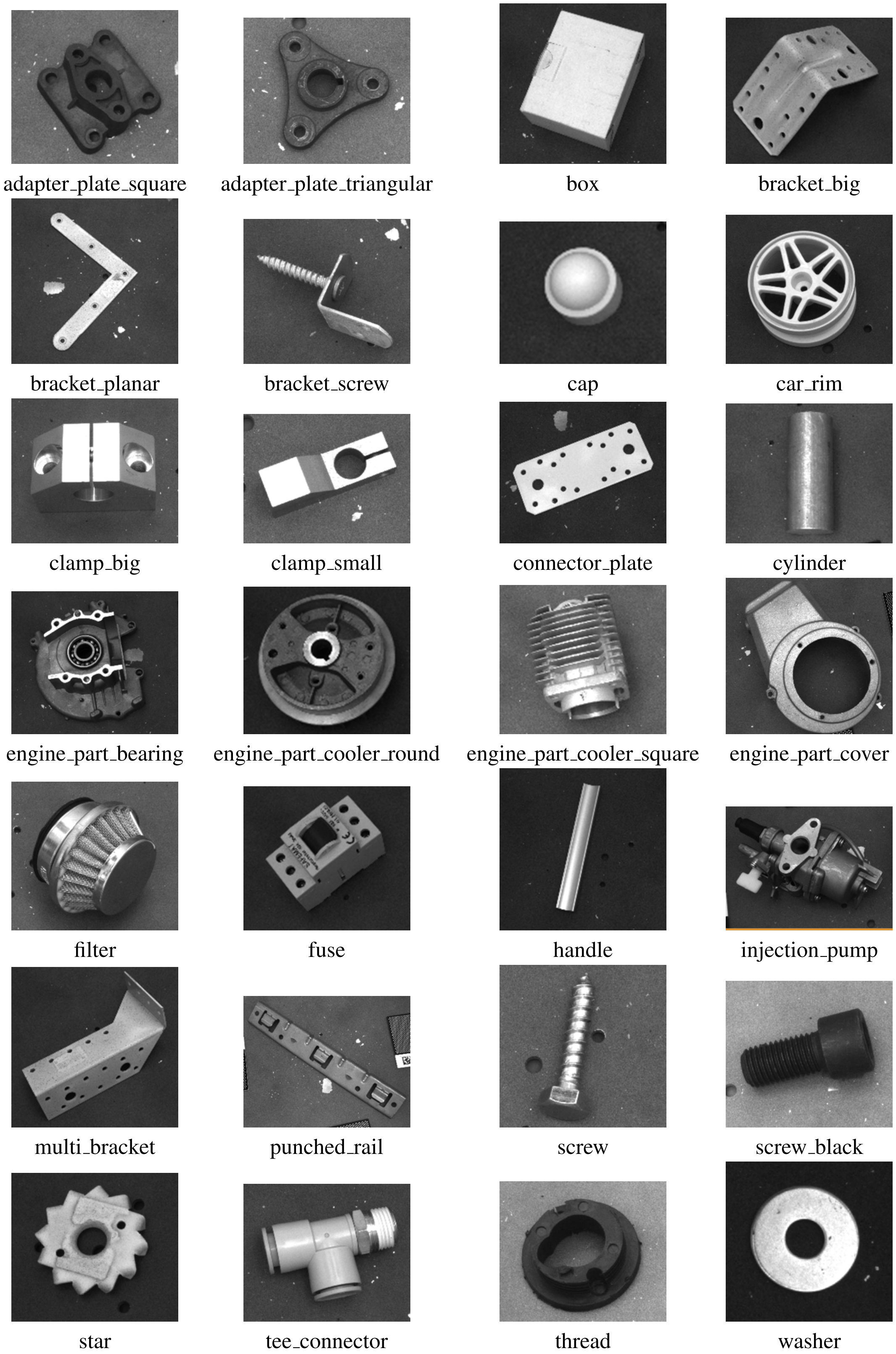
$$\forall T \in S_M : M \approx TM, \tag{4}$$

the distance measure becomes

$$d^P(T_1, T_2) = \operatorname*{arg\,min}_{S \in S_M} \frac{\max_{x \in M} |T_1 x - T_2 S x|_2}{\operatorname{diam}(M)} \tag{5}$$

The errors in translation and rotation are handled accordingly. We model two kind of symmetries: continous rotational symmetries, for objects such as cylinder or cap, and

[2] We used the implementation available in the CloudCompare library, `http://www.danielgm.net/cc/`, which is different from the implementation used in the methods we evaluated to avoid any bias.



adapter_plate_square
adapter_plate_triangular
box
bracket_big
bracket_planar
bracket_screw
cap
car_rim
clamp_big
clamp_small
connector_plate
cylinder
engine_part_bearing
engine_part_cooler_round
engine_part_cooler_square
engine_part_cover
filter
fuse
handle
injection_pump
multi_bracket
punched_rail
screw
screw_black
star
tee_connector
thread
washer

Figure 3. Images of the 28 objects used in the dataset. The objects were selected to show different characteristics in terms of **surface** (reflecting vs. lambertian), **symmetry** (no vs. full rotational symmetry), **complexity** (primitive shapes vs. complex objects), **flatness** (flat vs. voluminous), **details** (no vs. very fine details on surface), **compactness** (long vs. compact), and **size** (diameters from 2.4 cm to 27 cm)

sets of discrete symmetric poses for objects such as box or car_rim. Objects that are almost symmetric and where sensors were unable to distinguish the symmetries, such as screw and screw_black, are also modeled to be rotationally symmetric.

Contrary to the evaluation criteria proposed in [8], which measures relative surface overlaps of detection and ground truth, we decided to include symmetry explicitly. We believe that for applications such as bin picking, detecting an incorrect pose, even if almost indistinguishable from the correct point for the particular viewpoint, is dangerous from an application perspective. It also allows methods that operate on the data of multiple sensors to take advantage of those different viewing directions to resolve such ambiguity.

**Detection Rate** To compute the detection rate given a set of results $R$, a set of ground truth transformations $GT$, and a threshold $t$ for the distance $d^P$, we first search, for each result transformation $T_R \in R$, the best matching ground truth $T_{GT} \in GT$ where $d^P(T_R, T_{GT}) < t$. If multiple ground truth transformations match this criterion, the one with the smallest distance is used. Each ground truth transformation is assigned to at most one result transformation, again the one with the smallest distance. Because of this, if $R$ contains duplicate results, only the best is classified as correct, while all others are false positives.

The **detection rate** used in the evaluation is then computed as the ratio of correctly matched transformations in $R$ vs. the total number of ground truth transformations, $|GT|$. The **false positive rate** is the number of unmatched result transformations vs. the total number of result transformations $|R|$.

**Computational Costs and Metadata** Since runtime is an important factor in real-world applications, for all evaluated methods, we also measure the training and detection times, model size, and memory requirements during detection. Since runtimes can heavily depend on the system, the used components (CPU vs. GPU) and the effort spent for the implementation, we also provide a free-form text field where the implementation and the used system can be summarized.

**Priors, Parameters, and Evaluation Rules** To enable an evaluation that is as realistic and fair as possible, evaluations on the dataset should obey the following rules regarding their priors and parametrization.

- **Per-Model Parameters**: Parameters may be set on a per-model basis. All parameters that are not shared between models must be summarized in text form to obtain an overview of the usability of the method.
- **Per-Scene Parameters**: Parameters may not be tuned on a per-scene basis, i.e., the detection parameters must be constant for a particular object. The only prior allowed on a per-scene basis is the number of instances contained in the scene, which is provided along with the dataset. The background plane may be removed from scenes, if documented accordingly.
- **Provided Parameters**: In addition to the CAD models of the objects and the number of instances per scene, the distance range (i.e., the range of $z$-values of the model centers within the scene) are provided to allow training methods that require, for example, synthetic renderings of the object. Note that the rotation range is not limited.

Of course, evaluations can be valuable even when not strictly adhering to the rules above. Such cases, however, should be summarized, and the corresponding descriptions will be published along with the results on the website.

# 5. Evaluation

Along with the dataset, this work also provides evaluations of several methods on the dataset. This allows a first insight into the difficulty of the dataset given state of the art detection methods. Note that additional and more detailed results will be found on the dataset's website.

## 5.1. Evaluated Methods

**Shape-Based 3D Matching** *(S2D)* An optimized implementation of [12], which detects 3D objects in 2D images. A template-based matching approach is used, where the object is rendered from multiple viewpoints to create templates for different orientations. This method does not use any 3D image data. Poses where flat objects are seen from the side are excluded during the training to avoid degenerated views. Additionally, the image contrast and the number of trained image levels were adapted on a per-object basis.

**Point-Pair Voting** *(PP3D)* An optimized implementation of [4], which detects objects in 3D point clouds by using a local Hough transform and point pairs as features. The method was augmented with a point-to-plane ICP [2]. Identical parameters were used for all models, both during training and evaluation. For detection, the background plane was removed from the scenes.

**Point-Pair Voting with 3D edges** *(PP3D-E)* Based on [4], we implemented a method that, similar to [3], performs the voting not only for pairs of surface points, but also for pairs of surface and edge points. This allows the detector to optimize both the surface overlap and the alignment of 3D edges. Identical parameters were used for all models, both during training and evaluation.

**Point-Pair Voting with 3D edges and 2D refinement** *(PP3D-E-2D)* As another extension of the previous method, we extended the refinement (ICP) step such that it not only optimizes the 3D point-to-plane distances between scene and model, but also the alignment of reprojected model edges and 2D image edges, i.e., a multimodal refinement. Identical parameters were used for all models, both during training and evaluation.

**Efficient RANSAC** *(RANSAC)* We evaluated the publicly available variant of [9, 10], using the 3D data only. For the evaluation, the background plane was removed to obtain reasonable runtimes. The method also includes an ICP refinement. Identical parameters were used for all models and scenes.

For RANSAC, we used the publicly available C++-implementation. For the other methods, the optimized implementations of the HALCON machine vision library [1], version 13.0.1, were used.

### 5.2. Results

As a main result, Tab. 2 shows the Top-1 detection rate of the different methods, evaluated on different thresholds between detected transformations and ground truth. Tab. 3 shows the corresponding detection rate of the first n results, where n is the number of labeled instances per scene. Note that the Top-n-rate is significantly lower than the Top-1-rate, indicating that it is much easier to find *any* instance instead of *all* instances.

Tab. 4 shows the mean error of translation and rotation for all transformations labeled as correct, using different thresholds.

Fig. 4 shows the Top-1 detection rate vs. the average detection time per labeled instance. Note that even though the PP3D-E method has a good performance, it also has a rather high runtime, making it less qualified for real-world applications. Note also that all methods used the CPU only.

When comparing the performance of S2D to that of the other methods, it should be noted that it is the only evaluated approach that does not use 3D input data. Furthermore, we noticed that although many results of the S2D seemed to be correct when projecting them into the images, they resulted in large values for $d^P$, and hence were classified as false positives. The main reason is that because of the large focal lengths, a small error in the estimated object scale in the image or a small error in the size of the CAD model result in large errors in the z coordinate.

## 6. Discussion

This paper introduces the MVTec Industrial 3D Object Detection Dataset (MVTec ITODD) for 3D object detection and pose estimation. The extensive dataset is focused on modelling industrial applications: Setup, sensors, objects and evaluation criteria were selected to most closely match scenarios found in real-world applications.

A first evaluation on five different methods shows their characteristics, strengths, and weaknesses for different object classes. It also highlights that the dataset is not yet maxed out by existing methods, and that there is plenty of

| **Method** | $< 1\%$ | $< 3\%$ | $< 5\%$ | $< 10\%$ |
|---|---|---|---|---|
| PP3D | 0.07 | 0.48 | 0.66 | 0.75 |
| PP3D-E | 0.11 | 0.51 | 0.68 | 0.77 |
| PP3D-E-2D | 0.04 | 0.42 | 0.66 | 0.81 |
| S2D | 0.02 | 0.20 | 0.34 | 0.47 |
| RANSAC | 0.07 | 0.23 | 0.33 | 0.43 |

Table 2. **Top-1 detection rate**. For each method, the best result (according to each method's confidence score) for each object and scene is used and compared against the ground truth. The match is classified as correct, if the smallest distance $d^P$ to any of the ground truth transformations is closer than the given threshold. Different thresholds simulate different requirements on the accuracy of the match. This simluates a *pick any* strategy, where for further processing, at least one instace must be detected.

| **Method** | $< 1\%$ | $< 3\%$ | $< 5\%$ | $< 10\%$ |
|---|---|---|---|---|
| PP3D | 0.04 | 0.29 | 0.45 | 0.53 |
| PP3D-E | 0.05 | 0.34 | 0.50 | 0.59 |
| PP3D-E-2D | 0.02 | 0.26 | 0.46 | 0.61 |
| S2D | 0.01 | 0.10 | 0.17 | 0.25 |
| RANSAC | 0.03 | 0.13 | 0.19 | 0.27 |

Table 3. **Top-N detection rate**. For each object and scene, the first N results are compared against the ground truth, where N is the number of labeled object instances in the scene (see Tab. 2 for details).

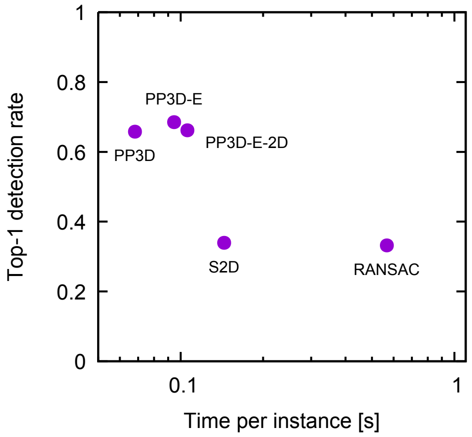


Figure 4. **Matching time vs. average detection rate.** The average detection time per instance is plotted against the Top-1 detection rate. A threshold of $d^P < 5\%$ was used for classifying a result as correct.

| Method | $d^P < 1\%$ | | $d^P < 3\%$ | | $d^P < 5\%$ | | $d^P < 10\%$ | |
|---|---|---|---|---|---|---|---|---|
| | $d^T$ | $d^R$ | $d^T$ | $d^R$ | $d^T$ | $d^R$ | $d^T$ | $d^R$ |
| PP3D | 0.55% | 0.31° | 1.28% | 0.58° | 1.61% | 0.72° | 1.95% | 0.91° |
| PP3D-E | 0.58% | 0.32° | 1.20% | 0.54° | 1.54% | 0.69° | 1.83% | 0.87° |
| PP3D-E-2D | 0.59% | 0.40° | 1.25% | 0.75° | 1.68% | 0.97° | 2.06% | 1.25° |
| S2D | 0.56% | 0.31° | 1.41% | 0.58° | 1.89% | 0.81° | 2.64% | 1.12° |
| RANSAC | 0.48% | 0.31° | 1.06% | 0.54° | 1.39% | 0.83° | 1.99% | 1.52° |

Table 4. **Top-1 pose error**. For each method, the first result for each object and scene is used and compared against the ground truth, yielding the relative translation error $d^T$ and the rotation error $d^R$. For most methods, the main source of error comes from the translation, not from the rotation.

| Model | PP3D | PP3D-E | PP3D-E-2D | S2D | RANSAC |
|---|---|---|---|---|---|
| adapter_plate_square | 0.58 | 0.58 | 0.56 | 0.06 | 0.11 |
| adapter_plate_triangular | 0.64 | 0.63 | 0.44 | 0.01 | 0.07 |
| box | 0.69 | 0.78 | 0.80 | 0.39 | 0.44 |
| bracket_big | 0.53 | 0.65 | 0.65 | 0.38 | 0.51 |
| bracket_planar | 0.15 | 0.23 | 0.24 | 0.37 | 0.20 |
| bracket_screw | 0.02 | 0.04 | 0.02 | 0.00 | 0.05 |
| cap | 0.82 | 0.80 | 0.69 | 0.06 | 0.00 |
| car_rim | 0.48 | 0.49 | 0.36 | 0.07 | 0.42 |
| clamp_big | 0.46 | 0.39 | 0.38 | 0.51 | 0.33 |
| clamp_small | 0.28 | 0.31 | 0.26 | 0.09 | 0.00 |
| connector_planar | 0.24 | 0.37 | 0.38 | 0.30 | 0.46 |
| cylinder | 0.72 | 0.73 | 0.74 | 0.24 | 0.86 |
| engine_part_bearing | 0.85 | 0.87 | 0.75 | 0.34 | 0.00 |
| engine_part_cooler_round | 0.81 | 0.84 | 0.67 | 0.74 | 0.26 |
| engine_part_cooler_square | 0.44 | 0.52 | 0.39 | 0.00 | 0.12 |
| engine_part_cover | 0.85 | 0.83 | 0.82 | 0.43 | 0.67 |
| filter | 0.10 | 0.10 | 0.05 | 0.01 | 0.24 |
| fuse | 0.07 | 0.54 | 0.41 | 0.82 | 0.00 |
| handle | 0.60 | 0.67 | 0.73 | 0.03 | 0.21 |
| injection_pump | 0.71 | 0.74 | 0.60 | 0.08 | 0.33 |
| multi_bracket | 0.65 | 0.78 | 0.77 | 0.45 | 0.53 |
| punched_rail | 0.37 | 0.38 | 0.39 | 0.06 | 0.36 |
| screw | 0.30 | 0.17 | 0.03 | 0.00 | 0.00 |
| screw_black | 0.30 | 0.16 | 0.19 | 0.07 | 0.32 |
| star | 0.36 | 0.60 | 0.64 | 0.25 | 0.29 |
| tee_connector | 0.78 | 0.66 | 0.55 | 0.16 | 0.07 |
| thread | 0.33 | 0.30 | 0.33 | 0.12 | 0.36 |
| washer | 0.04 | 0.03 | 0.04 | 0.00 | 0.00 |

Table 5. **Top-N detection rates per object**. A threshold of $d^P < 5\%$ was used for classifying a result as correct.

room for improvement.

We hope that this dataset encourages others to consider industrial challenges during the design and development of new methods, and that it helps to identify well-performing existing methods.

**Acknowledgements** The authors would like to thank Maximilian Fichtl, Fabian Hainzl, and Clarissa Siegfarth for their help in the creation of the dataset.

## References

[1] `https://www.mvtec.com/products/halcon/`. 7

[2] Y. Chen and G. Medioni. Object modelling by registration of multiple range images. *Image and vision computing*, 10(3):145–155, 1992. 6

[3] C. Choi, Y. Taguchi, O. Tuzel, M.-Y. Liu, and S. Ramalingam. Voting-based pose estimation for robotic assembly using a 3d sensor. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1724–1731. IEEE,

2012. 6

[4] B. Drost, M. Ulrich, N. Navab, and S. Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 998–1005. Ieee, 2010. 6

[5] M. Firman. RGBD datasets: Past, present and future. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 19–31, 2016. 2

[6] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian conference on computer vision*, pages 548–562. Springer, 2012. 4

[7] T. Hodan, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, and X. Zabulis. T-LESS: An RGB-D dataset for 6d pose estimation of texture-less objects. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 880–888. IEEE, 2017. 2

[8] T. Hodaň, J. Matas, and Š. Obdržálek. On evaluation of 6d object pose estimation. In *Computer Vision–ECCV 2016 Workshops*, pages 606–619. Springer, 2016. 6

[9] C. Papazov and D. Burschka. An efficient ransac for 3d object recognition in noisy and occluded scenes. In *Asian Conference on Computer Vision (ACCV)*, pages 135–148. Springer, 2010. 7

[10] C. Papazov, S. Haddadin, S. Parusel, K. Krieger, and D. Burschka. Rigid 3d geometry matching for grasping of known objects in cluttered scenes. *The International Journal of Robotics Research*, 31(4):538–553, 2012. 7

[11] C. Steger, M. Ulrich, and C. Wiedemann. *Machine Vision Algorithms and Applications*. Wiley-VCH, Weinheim, 2007. 2

[12] M. Ulrich, C. Wiedemann, and C. Steger. Combining scale-space and similarity-based aspect graphs for fast 3d object recognition. *IEEE transactions on pattern analysis and machine intelligence*, 34(10):1902–1914, 2012. 6