Illustrative visualizations of key parts of our method are shown in Section 1, including a step-wise example showing the corresponding pose scoring. Extending upon the results in the main text, we provide further ablations with respect to convergence, initialization error and segmentation quality in Section 2. Additionally, a formal definition of the used evaluation metrics is given in Section 3.

1. Additional Visualizations

The following visualizations aim to illustrate key concepts of SporeAgent by qualitative examples.

1.1. Visualizing the Scene Representation

Figure 1 shows the target point clouds for a frame in YCB-VIDEO (YCBV) [2] in the scene representation. Critical points for a queried object (the coffee can, shown in gray) are indicated. Under initial poses (left), the object would intersect with the plane and a neighboring object (shown red). The supported points (cyan) would span a supporting polygon sufficient for static stability. But we define feasibility, i.e., non-intersecting and non-floating, as a precondition for plausibility, and hence the object pose is considered implausible under its current pose with the scene. The remaining objects in the scene are processed analogously.

After refinement using SporeAgent (Figure 1 right), these implausibilities are resolved. Objects are resting on the supporting plane and no longer intersect (subject to slack parameter $\varepsilon$). Contacts (green) of the queried object with the object resting on-top of it are considered non-supported, since the surface normals of the neighboring object near the contacts are pointing in gravity direction.

1.2. Visualizing the Pose Scoring

The scoring of pose estimates with respect to the observed frame is illustrated in Figure 2. The estimates are visualized by outlines (blue) and the corresponding per-pixel score as a heatmap. For the observation, the depth and normal image are shown (right).

As seen in the left-most column in Figure 2, only the initial pose estimate for the sugar box (yellow, front) results in close alignment, indicated by a warm color in the heatmap. Already within the first steps of refinement using SporeAgent, alignment is significantly increased. The best poses after the full refinement result in an even finer alignment, also resolving the large initial pose errors for the driller and the clamp (top right in the frame).

1.3. Canonical Object Frame

As shown in the main results, representing the target objects in a canonical frame and considering their symmetries significantly improves accuracy on YCBV. We propose to align symmetry axes as to simplify the annotation of geometric symmetries, as illustrated in Figure 3.

By centering the axes, the different symmetrical poses of an object reduce to rotations. This, in addition, allows to define the expert policy for a given estimate as resolving the smallest rotation error with respect to one of these rotations.

1.4. Visualizing the Segmentation Augmentation

The augmentation of the instance segmentation during training is visualized in Figure 4. The foreground (blue) and background (purple) are determined using the ground-truth visibility mask, with the background restricted to the bounding box around the mask. Both regions are pre-
2. Additional Experiments

Given that we use Reinforcement Learning (RL) and several data augmentations that depend on random values, the training of SporeAgent is dependent on the random seed value. During inference, the performance of our method is moreover affected by the number of refinement iterations and the pose initialization.

sampled to an equal number of points. Per sample, the augmentation randomly selects one of the foreground pixels (cross) and determines its nearest neighbors in image space. Depending on a uniformly-random fraction $p$ and a total number of points to sample $n$, the $\lceil pn \rceil$ nearest neighbors in the foreground and $\lfloor (1-p)n \rfloor$ nearest neighbors in the background are sampled. This results in a coherent foreground patch, simulating occlusion or a too small mask for $p < 1$. The background patch will consist of a part that is coherent with the foreground patch (too large mask, bleeding-out into the surrounding of the object) and a part that fades farther into the surrounding (outliers). As shown in the experiments on YCBV, where training and test scenes are different, this approach allows to learn which points to ignore rather than learning specific scene surroundings.

2.1. Convergence with Varying Random Seed

Figure 5 shows the training convergence of SporeAgent on the LINEMOD (LM) [1] dataset for 5 different random seeds. The reported AD recalls are obtained by evaluation on the test set using the current weights after the corresponding training epoch. After 50 epochs, the final performance for the $0.10d$ threshold is already achieved and the recall is within $1\%$ for the $0.05d$ threshold. The standard deviation for the last epoch is $0.02, 0.2$ and $0.3\%$ for the $0.10, 0.05$ and $0.02d$ threshold, respectively.

Figure 4. Segmentation augmentation during training with $p = 50\%$ foreground samples. Selected center (cross), sampled foreground (blue) and sampled background (purple). Background samples are limited to a bounding box around the target object.
Recall [%]

3. Definition of Metrics

Hinterstoisser et al. [1] propose two metrics for the evaluation of estimated object poses of a given 3D model. The Average Distance of Model Points (ADD) is defined as the mean distance between corresponding model points \( m \in M \) under estimated pose \( \hat{T} \) and under ground-truth pose \( T \). To deal with symmetrical objects, the Average Distance of Model Points with Indistinguishable Views (ADI) computes the mean distance between the nearest neighbors under either pose. Formally, the metrics are defined as

\[
ADD = \frac{1}{|M|} \sum_{m \in M} ||\hat{T}m - Tm||_2,
\]

\[
ADI = \frac{1}{|M|} \sum_{m_1 \in M} \min_{m_2 \in M} ||\hat{T}m_1 - Tm_2||_2.
\]

Additionally, we abbreviate a mixed usage of both metrics with \( AD \). For \( AD \), objects considered symmetrical are evaluated using the ADI metric and using ADD otherwise.

These metrics are computed for all \( N \) test samples and the recall for a given precision threshold \( th \) is defined as

\[
AD_{th} = \frac{1}{N} \sum_{i=1}^{N} \left\{ \begin{array}{ll} 0, & AD_i > th \\ 1, & AD_i \leq th. \end{array} \right. \]

For the Area Under the precision-recall Curve (AUC), \( th \) is varied within a range of precision thresholds and the area under resulting curve of recall values is reported.

References
