Explaining the Ambiguity of Object Detection and 6D Pose From Visual Data

Fabian Manhardt[1,*]
fabian.manhardt@tum.de

Diego Martin Arroyo[1,*]
martin.arroyo@tum.de

Christian Rupprecht[2]
chrisr@robots.ox.ac.uk

Benjamin Busam[1,3]
benjamin.busam@huawei.com

Tolga Birdal[4]
tbirdal@stanford.edu

Nassir Navab[1]
nassir.navab@tum.de

Federico Tombari[1,5]
tombari@in.tum.de

[1]Technical University of Munich   [2]University of Oxford   [3]Huawei   [4]Stanford University   [5]Google

Abstract

3D object detection and pose estimation from a single image are two inherently ambiguous problems. Oftentimes, objects appear similar from different viewpoints due to shape symmetries, occlusion and repetitive textures. This ambiguity in both detection and pose estimation means that an object instance can be perfectly described by several different poses and even classes. In this work we propose to explicitly deal with these ambiguities. For each object instance we predict multiple 6D pose outcomes to estimate the specific pose distribution generated by symmetries and repetitive textures. The distribution collapses to a single outcome when the visual appearance uniquely identifies just one valid pose. We show the benefits of our approach which provides not only a better explanation for pose ambiguity, but also a higher accuracy in terms of pose estimation.

1. Introduction

Driven by deep learning, image-based object detection has recently made a tremendous leap forward in both accuracy as well as efficiency [39, 16, 31, 38]. An emerging research direction in this field is the estimation of the object's pose in 3D space over the existing 6-Degrees-of-Freedom (DoF) rather than on the 2D image plane [24, 37, 46, 51, 34, 29, 49, 33]. This is motivated by a strong interest in achieving robust and accurate monocular 6D pose estimation for applications in the field of robotic grasping, scene understanding and augmented/mixed reality, where the use of a 3D sensor is not feasible [36, 26, 50, 45].

Nevertheless, 6D pose estimation from RGB is a challenging problem due to the intrinsic ambiguity caused by visual appearance of objects under different viewpoints and occlusion. Indeed, most common objects exhibit shape ambiguities and repetitive patterns that cause their appearance

---

* The first two authors contributed equally to this work.
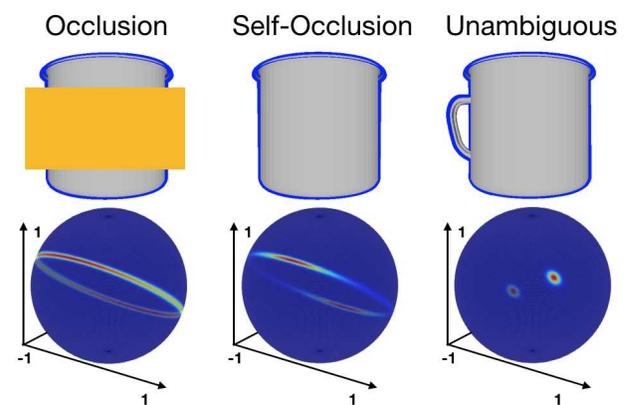
Figure 1: **Pose ambiguities**. External or self-occlusion can cause the 6DoF pose of an object to become ambiguous. Our method is able to detect and predict these ambiguities automatically without additional supervision. The antipodally symmetric Bingham distributions show that the model has understood the full range of valid poses.

to be very similar under different viewpoints, thus rendering pose estimation a problem with multiple correct solutions. Furthermore, also occlusion (from the same object or from others) can cause pose ambiguity.

For example, as illustrated in Figure 1, the cup is identical from every viewpoint in which the handle is not visible. Thus, from a single image, it is impossible to univocally estimate the current object pose. Moreover, object symmetry can also induce visual ambiguities leading to multiple poses with the same visual appearance. However, most datasets do not reflect this ambiguity, as the ground truth pose annotations are mostly uniquely defined at each frame. This is problematic for a proper optimization of the rotation, since a visually correct pose still results in a high loss. Thus, many recent 3D detectors avoid regressing the rotation directly and, instead, explicitly model the solution space in an unambiguous fashion [37, 24].

Essentially, in [24], the authors train their convolutional

Figure 2: **Overview.** We predict M hypotheses for the pose to approximate the distribution in the solution space. Each hypothesis is visually identical from the current viewpoint.

neural network (CNN) by mapping all possible pose solutions for a certain viewpoint onto an unambiguous arc on the view sphere. Rad *et al*. [37] employ a separate CNN solely trained to classify the symmetry in order to resolve these ambiguities. However, this simplification exhibits several downsides, such as the explicit inclusion of information about certain symmetries in each trained object. Moreover, this is not always easy to model, as *e.g.* in the case of partial view ambiguity. Further, all these approaches rely on prior knowledge and annotation of the object symmetries and aim to solve the ambiguity by providing a single outcome in terms of estimated pose and object. Added to this, these methods are also unable to deal with ambiguities generated by other common factors such as occlusion.

On the contrary, Sundermeyer *et al*. [42] and Corona *et al*. [7] recently proposed novel methods to conduct pose estimation in an ambiguity-free manner. In the core, both learn a feature embedding solely based on visual appearance. Nonetheless, although [42] is able to deal with ambiguities implicitly, it does not model their detection and description explicitly. In contrast, [7] also learns to classify the order of rotational symmetry, in particular the number of equivalent views around an axis of rotation. However, they require explicit hand-annotated labels and, in addition, cannot deal with ambiguities aside from these symmetry classes such as (self-) occlusion.

In this paper we propose to model the ambiguity of the object detection and pose estimation tasks directly by allowing our learned model to predict multiple solutions, or *hypotheses*, for a given object's visual appearance (Fig 2). Inspired by Rupprecht *et al*. [40] we propose a novel architecture and loss function for monocular 6D pose estimation by means of multiple predictions. Essentially, each predicted hypothesis itself corresponds to a 3D translation and rotation. When the visual appearance is ambiguous, the model predicts a point estimate of the distribution in 3D pose space. Conversely, when the object's appearance is unique, the hypotheses will collapse into the same solution. Importantly, our model is capable of learning the distribu-

tion of these 6D hypotheses from one single ground truth pose per sample, without further supervision.

Besides providing more insight and a better explanation for the task at hand, the additional knowledge gained from rotation distributions can be exploited to improve the accuracy of the pose estimates. In essence, analyzing the distribution of the hypotheses enables us to classify if the current perceived viewpoint is ambiguous and to compute the axis of ambiguity for that specific object and viewpoint. Subsequently, when ambiguity is detected, we can employ mean shift [6] clustering over the hypotheses in quaternion space to find the main modes for the current pose. A robust averaging in 3D rotation space for each mode then yields a highly accurate pose estimate. When the view is ambiguity-free, we can improve our pose estimates by robustly averaging over all 6D hypotheses, and by taking advantage of the predicted pose distribution as a confidence measure.

Our contributions are threefold:

- We propose a novel method for 6DoF pose estimation, which can deal with the inherent ambiguities in pose by means of multiple hypotheses.

- Explicit detection of rotational ambiguities and characterization of the uncertainty in the problem without further annotation or supervision.

- A mechanism to measure the reliability and to increase the robustness of the unambiguous 6D pose prediction.

2. Related Work

We first review recent work in object detection and pose estimation from 2D and 3D data. Afterwards, we discuss common grounds and main differences with approaches aimed at symmetry detection for 3D shapes.

Object Detection and Pose Estimation. Almost all current research focus on deep learning-based methods. [48, 25, 7] employ CNNs to learn an embedding space for the pose and class from RGB-D data, which can subsequently be utilized for retrieval. Notably, the majority of most recent deep learning based methods focus on RGB as input [24, 37, 8, 46, 51, 42]. Since utilizing pre-trained networks often accelerates convergence and leads to better local minima, these methods are usually grounded on state-of-the-art backbones for 2D object detection, such as Inception [44] or ResNet [16]. In particular, Kehl *et al*. [24] employ SSD [31] with an InceptionV4 [43] backbone and extend it to also classify viewpoint and in-plane rotation. Similarly, Sundermeyer *et al*. [42] also use SSD for localization, but employ an augmented auto-encoder for the unambiguous retrieval of the associated 6D pose. Rad *et al*. [37] utilize VGG [41] and augment it to provide the 2D projections

of the 3D bounding box corners. A similar approach is chosen by [46], based on YOLO [38]. Afterwards, both apply P*n*P to fit the associated 3D bounding box into the regressed 2D projections, in order to estimate the 3D pose of the detection. In [51], Xiang *et al.* compute a shared feature embedding for subsequent object instance segmentation paired with pose estimation. Finally, Do *et al.* [8] extend Mask-RCNN [15] with a third branch, which provides the 3D rotation and the distance to the camera for each prediction.

Object Symmetry Detection Oftentimes, object pose ambiguity arises from symmetric shapes. We review relevant methods that extract symmetry from 3D models to outline commonalities and differences with our approach.

To our knowledge, [7] is the only method which estimates both: the 6D pose, and the symmetry of the perceived object. In particular, the network is trained to also predict the rotational order (*i.e.* the number of identical views), posing it as a classification task.

Generally, most methods for symmetry detection are found in the shape analysis community. Among the different kinds of symmetries, axial symmetries are of particular interest, and multiple approaches have been proposed. Most methods rely on feature matching or spectral analysis: [9] treat the problem as a correspondence matching task between a series of keypoints on an object, determining the reflection symmetry hyperplane as an optimization problem. Elawady *et al.* [10] rely on edge features extracted using a Log-Gabor filter in different scales and orientations coupled with a voting procedure on the computed histogram of local texture and color information. In addition, [5] and [35] are also grounded on wavelet-based approaches. Recently, neural network approaches have also been proposed. Ke *et al.* [23] adapt an edge-detection architecture with multiple residual units and successfully apply it to symmetry detection using real-world images.

Notably, all these approaches aim at detecting symmetries of 3D shapes alone, while our focus is to model the ambiguity arising from objects under specific viewpoints with the goal of improving and explaining pose estimation.

3. Methodology

In this section we describe our method for handling symmetries and other ambiguities for object detection and pose estimation in detail. We will first define what we understand as an ambiguity.

3.1. Ambiguity in Object Detection and Pose Estimation

We describe the rigid body transformations $SE(3)$ via the semi-direct product of $SO(3)$ and $\mathbb{R}^3$. While for the latter, we use Euclidean 3-vectors, the algebra $\mathbb{H}_1$ of unit

quaternions is used to model the spatial rotations in $SO(3)$. A quaternion is given by

$$\mathbf{q} = q_1\mathbf{1} + q_2\mathbf{i} + q_3\mathbf{j} + q_4\mathbf{k} = (q_1, q_2, q_3, q_4), \quad (1)$$

with $(q_1, q_2, q_3, q_4) \in \mathbb{R}^4$ and $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -\mathbf{1}$. We regress quaternions above the $q_1 = 0$ hyperplane and, thus, omit the southern hemisphere, such that any possible 3D rotation can be expressed by only one single quaternion.

Under ambiguities, a direct naive regression of the rotation as a quaternion will lead to poor results, as the network will learn to predict a rotation that is closest to all results in the symmetry group. This prediction can be seen as the (conditional) mean rotation. More formally, in a typical supervised setting we associate images I_i with poses p_i in a dataset (I_i, p_i) where $i \in \{1, \ldots, N\}$. To describe symmetries, we define for a given image I_i, the set $\mathcal{S}(I_i)$ of poses p that all have an identical image

$$\mathcal{S}(I_i) = \{p_j | I_j = I_i\}. \quad (2)$$

Note that in the case of non-discrete symmetries the set $\mathcal{S}$ will contain infinitely many poses, which in turn transforms the sums of S in the following to integrals. For the sake of a simpler notation and a finite training set in practice, we chose to continue with a notion of a finite $|S|$. The naive model $f(I, \theta)$, that directly regresses a pose p' from I, optimizes a loss $\mathcal{L}(p, p')$ by minimizing

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{N} \mathcal{L}(f_\theta(I_i), p_i) \quad (3)$$

over the training set. However, due to symmetry, the mapping from I to p is not well defined and cannot be modeled as a function. By minimizing Equation 3, f is learned to predict a pose $\tilde{p}$ approximating all possible poses for this image equally well.

$$f(I_i, \theta^*) = \tilde{p} = \min_{p} \sum_{j=1}^{|\mathcal{S}(I_i)|} \mathcal{L}(p, p_j) \quad (4)$$

This is an unfavorable result since $\tilde{p}$ is chosen to minimize the sum of all losses towards the different symmetries. In the following section, we will describe how we model these ambiguities inside our method.

3.1.1 Multiple Pose Hypotheses

The key idea behind the proposed method is to model the ambiguity by allowing multiple pose predictions from the network. In order to predict M pose hypotheses from f, we extend the notation to $f_\theta(I) = (f_\theta^{(1)}(I), \ldots, f_\theta^{(M)}(I))$ where f now returns M pose hypotheses for each image I.

For training, the idea is not to punish all hypotheses given the current pose annotation, since they might be correct under ambiguities. Thus, we use a loss that optimizes only one of the M hypotheses for each annotation. The most intuitive choice is to pick the closest one. We adapt the meta loss $\mathcal{M}$ from [40] that operates on f,

$$\theta^* = \underset{\theta}{\arg\min} \sum_{i=1}^{N} \mathcal{M}(f_\theta(I_i), p_i), \qquad (5)$$

while we use the original pose loss $\mathcal{L}$ for each $f^{(j)}$

$$\hat{\mathcal{M}}(f_\theta(I), p) = \min_{j=1,\dots,M} \mathcal{L}(f_\theta^{(j)}(I), p). \qquad (6)$$

However, the hard selection of the minimum in equation 6 does not work in practice as some of the hypothesis functions $f_\theta^{(j)}(I)$ might never be updated if they are initialized far from the target values. We relax $\hat{\mathcal{M}}$ to $\mathcal{M}$ by adding the average error for all hypotheses with an epsilon weight:

$$\mathcal{M}(f_\theta(I), p) = \left(1 - \epsilon \frac{M}{M-1}\right) \hat{\mathcal{M}}(f_\theta(I), p) +$$
$$\frac{\epsilon}{M-1} \sum_{j=1}^{M} \mathcal{L}(f_\theta^{(j)}(I), p). \qquad (7)$$

The normalization constants before the two components are designed to give a weight of $(1 - \epsilon)$ to $\hat{\mathcal{M}}$ and ϵ to the gradient distributed over all other hypotheses. When $\epsilon \to 0$, $\mathcal{M} \to \hat{\mathcal{M}}$. This is necessary since the average in the second term already contains the minimum from the first one.

3.2. Architecture

We employ SSD-300 [31] with an extended InceptionV4 [43] backbone and adjust it to also provide the 6D pose along with each detection. In particular, we append two more 'Reduction-B' blocks to the backbone. Essentially, we branch off after each dimensionality reduction block and place in total 6.099 anchor boxes to cover objects at different scales. Moreover, to include the unambiguous regression of the 6D pose, we modify the prediction kernel such that it provides $C + M \cdot P$ outputs for each anchor box. Thereby, C denotes the number of classes, M denotes the number of hypotheses, and P denotes the number of parameters to describe the 6D pose. In our case, for each of the M predicted hypotheses, we regress $P = 5$ values to characterize the 6D pose, composed of an explicitly normalized 4D quaternion for the 3D rotation and the object's distance towards the camera. We can estimate the remaining two degrees-of-freedom by back-projecting the center of the 2D bounding box using the inferred depth.

Additionally, in line with [32, 24] we conduct hard negative mining to deal with foreground-background imbalances. Thus, given a set of positive boxes *Pos* and hardmined negative boxes *Neg* for a training image, we minimize the following energy function:

$$\mathcal{L}(\text{Pos}, \text{Neg}) := \sum_{b \in \text{Neg}} \mathcal{L}_{class} +$$
$$\sum_{b \in \text{Pos}} (\mathcal{L}_{class} + \alpha \mathcal{L}_{fit} + \beta \mathcal{M}(f_\theta(I), p)). \qquad (8)$$

For the class and the refinement of the anchor boxes, we employ the cross-entropy loss $\mathcal{L}_{class}$ and the smooth L1-norm $\mathcal{L}_{fit}$, respectively. In order to compare the similarity of two quaternions, we compute the angle between the estimated rotation and the ground truth rotation according to

$$\mathcal{L}_{rotation}(q, q') = \arccos\left(2\langle q, q'\rangle^2 - 1\right). \qquad (9)$$

Additionally, we employ the smooth L1-norm as loss for the depth component $\mathcal{L}_{depth}$. Altogether, we define the final loss for each hypothesis j and input image I as follows

$$\mathcal{L}(f_\theta^{(j)}(I)) = \mathcal{L}_{rotation}(q^{(j)}, q') + \lambda \mathcal{L}_{depth}(d^{(j)}, d'). \quad (10)$$

3.3. Processing Multiple Hypotheses

During inference we further analyze the predicted multiple hypotheses in order to determine whether the pose of the object is ambiguous. Notice that prior to this, we first map all hypotheses to reside on the upper hemisphere. If we detect an ambiguity, we additionally exploit the multiple hypotheses to estimate the view-dependent axes of ambiguity.

Detection of Visual Ambiguities in Scenes. We analyze the distribution of predicted hypotheses in quaternion space to determine whether the pose exhibits an ambiguity. To this end, Principal Component Analysis (PCA) is performed on the quaternion hypotheses $\mathbf{q}_i$. The singular value decomposition of the data matrix indicates the ambiguity: if the dominant singular values $\sigma_{1/2} \gg 0$ ($\sigma_i > \sigma_{i+1} \,\forall i$), an ambiguity in the pose prediction is likely, while small singular values imply a collapse to a single unambiguous solution.

We determine the existence of ambiguity by thresholding the value of σ_2. Empirically, we find the criteria $\sigma_2 > 0.8$ to offer good estimations for ambiguity. It is noteworthy that we can learn to detect ambiguities without further supervision, directly from standard datasets.

Estimation of the Axis of Ambiguity. As mentioned, very prominent representatives for visual ambiguities are symmetries in the objects of interest, as illustrated in Fig. 3 (left) and (mid). Nevertheless, for other objects such as

Figure 3: **Examples of pose ambiguity.** Left: Rotational ambiguity. Mid: Two different possible poses for each side. Right: Ambiguity around an arc through (self-) occlusion.

cups, also (self-) occlusion can induce ambiguities in appearance (right).

To calculate a viewpoint dependant ambiguity axis, we take a closer look at the following scenario. A rotation $\mathbf{q}_i = (q_{i1}, q_{i2}, q_{i3}, q_{i4})$ rotates the camera c_0 to c_i around the rotation axis

$$a_i = (q_{i2}, q_{i3}, q_{i4}) \big/ \sqrt{q_{i2}^2 + q_{i3}^2 + q_{i4}^2}. \tag{11}$$

All these rotation axes lie in the same plane which is perpendicular to the ambiguity axis $s \perp a_i \; \forall i$. Thus, if we stack the rotation axes $A = \left(a_1^T, a_2^T, \cdots, a_n^T\right)$, we can formulate the overdetermined linear equation system $A^T s = 0$. The ambiguity axis can be found as the solution to the optimization problem

$$\min_{s \in \mathbb{R}^3} \left\| A^T s \right\|_p, \tag{12}$$

which we solve for $p = 2$ using SVD.

3.4. From Multiple Hypotheses to 6D Pose

After analyzing the distribution of the hypotheses, we can robustly compute the associated 6D pose for each case.

Unambiguous Object Pose. In case of an unambiguous object pose, we utilize the multiple hypotheses as an input for a geometric median (geodesic L_1-mean [14]) to improve robustness of the overall estimation

$$\mathbf{q}_{\text{gm}} = \operatorname*{argmin}_{\mathbf{q} \in \mathbb{H}_1} \sum_i \mathrm{d}_{\text{geo}}\left(\mathbf{q}_i, \mathbf{q}\right). \tag{13}$$

The iterative calculation follows the Weiszfeld algorithm [47, 13] in the tangent spaces to the quaternion hypersphere [4]. From a statistical perspective, our rotation measures are treated as inputs for an L_1-estimator to robustly detect the geometric median where d_{geo} gives the geodesic distance on the quaternion hypersphere. Note that Gramkow [12] showed that locally, using the Euclidean distance in the ambient, quaternion space well approximates the Riemannian one. In addition, we compute the median depth of all hypotheses. Afterwards, we utilize the center of the 2D detection and backproject it into 3D to obtain the translation and therewith the full 6D pose of the detection.

Ambiguous Object Pose. As the number of possible 3D rotations is finite yet unknown, we employ mean shift [6] to cluster the hypotheses in quaternion space. Specifically, we use the the angular distance of the quaternion vectors to measure similarity and the Weiszfeld algorithm to merge clusters inside mean shift. This yields either one cluster (if the poses are connected) or multiple (if they are unconnected) as illustrated in Fig. 3. For each cluster we compute a median rotation and the median depth to retrieve the associated 3D translation. Note that we only consider the depths of the hypotheses, which contributed to the corresponding cluster. We apply simple contour checks [24] to find the best fitting cluster from which we extract the final 6D pose.

Synthetic Data. As noted in [19], domain adaptation between synthetically generated data samples and real-world images trivializes the collection of training data. We render CAD models in random poses and add a series of augmentations, such as illumination changes, shadows and blur, as well as background images taken from the MS COCO [30].

4. Evaluation

In this section, we first introduce our experimental setup. Following that, we clearly demonstrate the benefits of our method compared to typical pose estimation systems on a toy dataset. Next, we show robustness in determining whether a view exhibits an ambiguity. Fourth, we report our 6D pose estimation accuracy for the unambiguous and the ambiguous case on common benchmark datasets. Finally, we demonstrate how we can model reliability in pose estimation by analyzing the variance across hypotheses.

4.1. Experimental Setup

Evaluation metrics. In order to properly assess the 6D pose performance, we distinguish between potentially ambiguous and non-ambiguous objects. When dealing with non-ambiguous objects, we report the absolute error for the 3D rotation in degrees and 3D translation in millimeters. We also show our accuracy using the Average Distance of Distinguishable Model Points (ADD) metric from [18], which measures if the average deviation of the transformed model points is less than 10% of the object's diameter.

For 'ambiguous' objects we rely on the Average Distance of Indistinguishable Model Points (ADI) metric, which extends ADD for ambiguity, measuring error as the average distance to the *closest* model point [21, 17].

We also show our results for the Visual Surface Similarity (VSS) metric. As [24], we define VSS similar to the Visual Surface Discrepancy (VSD) [21], however, set $\tau = \infty$. Hence, we measure the pixel-wise overlap of the rendered ground truth pose and the rendered prediction, which is not subject to ambiguities.
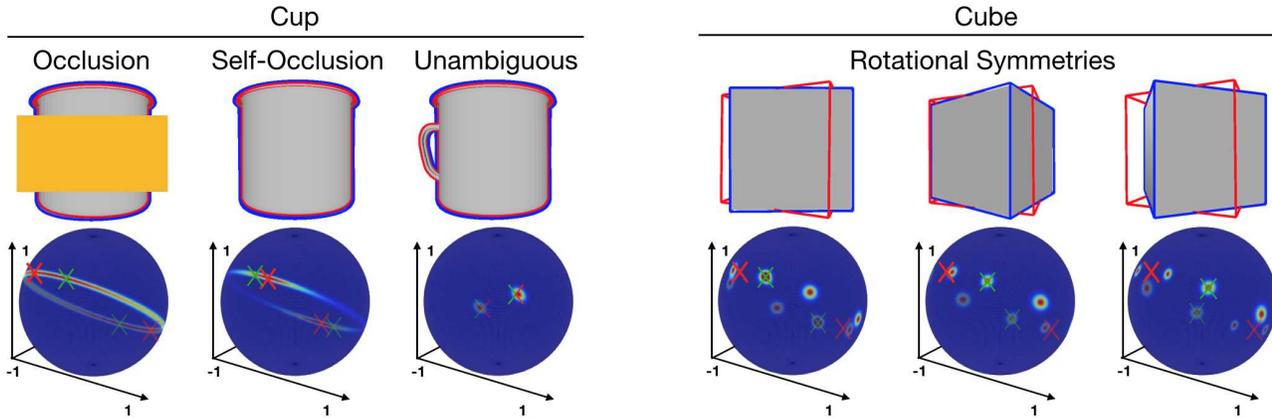
Figure 4: **Synthetic toy dataset.** Top: Contours of the rendered poses for the naive SH (M=1) model in red and our MH (M=30) model in blue. Bottom: Bingham distributions for each pose cluster, together with the ground truth quaternion in green and the SH predicted quaternion in red. Our model is not only correct in both cases but can also predict the full range of valid poses. SH fails on the cube example.

Object	Ambiguity	SH		MH	
		VSS [%]	ADI [%]	VSS [%]	ADI [%]
Cup	(Self-) Occlusion	97.0	**100**	**98.1**	**100**
Cube	Plane Symmetries	87.4	15.6	**98.6**	**100**

Table 1: **Synthetic results.** for the naive SH ($M = 1$) and our MH ($M = 30$) model on the synthetic toy dataset.

Bingham Distributions. In order to visually analyze the multi-hypotheses output of our network, we inspect the underlying rotation distributions. A *Bingham distribution* [1] (BD) is a special equivalent to a Gaussian distribution on a hypersphere. BDs represent a probability distribution on S^d with antipodal symmetry well suited to study poses parametrized by quaternions, where q and $-q \in \mathbb{H}_1$ represent the same element in $SO(3)$. In line with previous works [28, 11, 2], we visualize an equatorial projection of the closest distribution to our pose output using BDs.

4.2. Synthetic Ambiguity Evaluation

We render a simple synthetic dataset of a rotating cup and cube. We compare the baseline with $M = 1$ hypothesis and our method with $M = 30$ hypotheses. The results are shown in Fig. 4, Tab.1, and the supplement. For the cup, both methods yield an ADI score of 100%. The single hypothesis approach SH is indeed able to compute visually correct poses even though it cannot model the pose distribution along an arc. It has learned the conditional mean pose where the handle is exactly opposite of the camera. Nonetheless, this is only one of the infinitely many possible solutions. In contrast, our method is able to predict the whole distribution as seen in the Bingham plots. This is essential for tasks such as next-best-view prediction or robotic manipulation. When there is no ambiguity, both methods predict only the one correct pose.
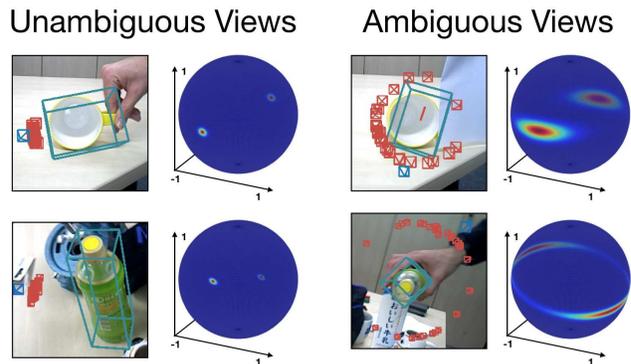
Figure 5: **Real data.** The red frustums visualize ($M = 30$) pose hypotheses. The blue frustum constitutes the median, which determines the predicted 3D bounding box. In the unambiguous case (left) the hypotheses agree. However, partial symmetries and occlusion lead to multiple possible outcomes on the right, which meaningfully reflect to the Bingham distribution of hypotheses.

For the cube object, SH fails (red outline) with an ADI of only 15.6%. Here, the conditional mean is not inside the set of correct poses. Our method is again able to estimate the underlying distribution and can correctly estimate all four modes of correct poses. This yields a perfect ADI of 100%.

When applying our method to real data (Fig. 5), we achieve similar results. If there is a unique solution, the method is able to robustly estimate the correct pose. For ambiguous views, we retrieve the governing distribution as depicted by the viewpoint frustums and spherical plots.

4.3. Real World Datasets

To conduct evaluations on real data, we build two datasets addressing both *unambiguous* and *ambiguous*
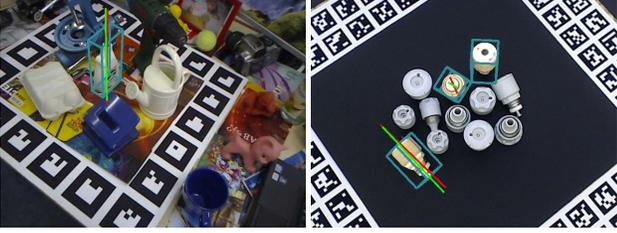
Figure 6: **Ambiguity detection.** Symmetry axis (green line) estimation. Notice that one screw was classified to be unambiguous (*i.e.* no axis), because the ambiguity could be resolved through the texture.

cases. In particular, for the former, we use the popular 'LineMOD' [18] and 'LineMOD Occlusion' dataset [27]. The authors of [27] selected one sequence from the original 'LineMOD' dataset and labeled eight additional objects. Nevertheless, we moved the 'glue' and 'eggbox' object to the *ambiguous dataset*, since both exhibit several views (mostly from the top), which are not unique. Additionally, following [24, 37] we removed the 'cup' and 'bowl' objects, because no watertight CAD models are provided for them. We also discard the 'lamp' since the CAD model does not possess correct normal vectors for proper rendering. To the latter, the *ambiguous* dataset, besides the 'glue' and 'bowl' objects, we added several models from T-LESS [20] to cover different types of ambiguities. In essence, T-LESS mostly consists of symmetric and texture-less industrial objects. For our experiments we choose a subset that covers both cases: complete rotational symmetry along an axis (object 4) and objects with more than one rotational symmetry (object 5, 9, 10).

4.4. Ambiguity Detection Analysis

To evaluate the ability of our model to learn pose distributions, we manually labeled for each validation image of the *ambiguous* dataset, whether the current object view exhibits ambiguity based on the visible object texture and shape. This ground truth is used to quantitatively assess our capability of detecting pose ambiguity. Additionally, we compute the ground truth symmetry axis for each object. It is important to note that we do not conduct object symmetry detection, instead, we describe the perceived pose ambiguity in terms of a symmetry axis. These annotations are only used for evaluation and not during training.

For each detected ambiguity, we compute the average discrepancy of the computed symmetry axis from the ground truth annotation. For the ambiguity-free case, we achieve to report an accuracy of more than 99%, while for the ambiguous case we can also state a high accuracy of 82% correctly classified views. Furthermore, the mean axis only deviates by 24°, which shows that our formulation is able to precisely explain the perceived ambiguity.

	Rot. [°]	Trans. [mm]	VSS [%]	ADD [%]	F1
SSD-6D [24]	28.0	72.4	67.4	9.4	88.8
[42]	–	–	–	22.1	–
SH ($M=1$)	17.9	45.6	76.8	31.2	91.6
MH ($M=5$)	**17.4**	**39.5**	**78.2**	**35.3**	**93.4**

Table 2: **Pose errors of unambiguous objects with synthetic training data**. Comparison with [42], [24]. Results of [24] from their released models and code.

	ape	can	cat	dril	duck	holep	mean
Tekin [46]	2.5	17.5	0.7	7.7	1.1	5.5	5.8
MH ($M=5$)	**5.9**	**22.4**	**4.2**	**32.0**	**12.2**	**17.0**	**15.6**

	BB-8 [37]	Tekin [46]	MH ($M=5$)
ADD [%]	45.9	**47.9**	44.4

Table 3: **Pose errors of unambiguous objects with real training data split from [3]**. Top: Comparison with [46] on LineMOD Occlusion. Bottom: Comparison with [37] and [46] on LineMOD. Results of [46] from their released models and code.

In Fig. 6, we respectively show one sample of estimated ambiguity axis from 'LineMOD' and 'T-LESS'. For each detection, we draw the estimated axis in red, while the green line denotes the hand-annotated groundtruth axis.

4.5. Comparison to State-of-the-Art

Unambiguous Pose Estimation. In Tab 2 and Tab 3, we report our results for the *unambiguous* subset for training with synthetic data and with the train data split from [3]. Since the number of predicted hypotheses M is a hyperparameter, we will show an ablation in the supplement and only report our best results with $M=5$ here.

For the case of synthetic training only, even for the single hypothesis case, our approach outperforms SSD-6D by more than 35% of relative error while also being more robust in terms of 2D detection. Comparing with Sundermeyer *et al.* [42] we can report a relative improvement of approximately 50% referring to ADD. In addition, our averaging over all hypotheses leads to more robustness towards outliers and, thus, another improvement of all metrics.

When also employing real data, we can improve our results by approximately 9% to 44.4% and are on par with the state-of-the-art methods from [37] and [46], even though we employ no crop and paste augmentations. Further, when using the more challenging 'LineMOD Occlusion' dataset, we can exceed Tekin *et al.* [46] for all objects and overall almost triple their ADD score from 5.8% to 15.6%.

Ambiguous Pose Estimation. Referring to Tab 4, for the ambiguous 'LineMOD' objects, we attain a VSS score of 79% and an ADI score of 55%, which is a relative improvement of approximately 13% and 145% compared to

	VSS [%]			ADI [%]			F1		
	MH	SH	[24]	MH	SH	[24]	MH	SH	[24]
eggbox	**83.1**	78.5	76.3	55.7	**56.0**	26.3	**98.0**	83.0	93.7
glue	**74.6**	74.0	65.1	54.6	**58.7**	17.6	**90.1**	74.0	76.8
mean	**78,9**	76.3	70.7	55.2	**57.4**	22.0	**94.1**	78.5	85.5

		VSS [%]			ADI [%]		
	Scene	MH	SH	[42]	MH	SH	[42]
obj_04	5, 9	70.8	68.6	**78.5**	**19.7**	14.1	15.2
obj_05	2, 3, 4	87.6	82.8	**88.8**	**78.0**	48.3	76.3
obj_09	5, 11	84.4	79.1	**86.5**	69.9	54.5	**77.3**
obj_10	5, 11	82.0	78.5	**82.3**	57.9	29.4	31.9
mean		81.2	77.3	**84.0**	**56.4**	36.6	50.6

Table 4: **Ambiguous dataset.** (top: 'LineMOD') (bottom: T-LESS). We compare our multiple hypotheses MH ($M = 30$) results and the same predictor trained to output a single hypothesis SH ($M = 1$) with [42][1] and SSD-6D [24].

SSD-6D. In the 6D setting, the multiple hypothesis detector overall achieves similar performance as the single hypothesis predictor. However, for the 2D detection case, we are able to increase the accuracy from 79% to 94%. As constituted, only a few views are ambiguous for these objects. Investigating the results, we discovered that the single hypothesis predictor is not able to understand exactly these views and tends to simply discard them. In contrast, the multiple hypotheses predictor is indeed able to understand these views and yields reliable pose predictions.

For all ambiguous 'T-LESS' objects (Tab 4), our multiple hypotheses approach surpasses the single hypothesis estimator, which, when trained and evaluated under the same conditions, is not able to capture the ambiguities in pose. Thus, the single hypothesis predictor is not able to produce equally accurate results, being only capable of computing precise poses for unambiguous views.Comparing with [42], we report similar performance in pose. Our ADI improves with 56.4% compared to 50.6% while VSS falls slightly behind by 2.8%. For fairness, we only compare the 6D pose accuracy for correctly detected objects (*i.e. $IoU \leq 0.5$*) since [42] trained their 2D detector for T-LESS on real data.

4.6. Measuring Reliability

To the best of our knowledge, there is no prior work capable of modelling the confidence in the continuous pose estimate. Yet, this information can highly improve the overall robustness and accuracy. In our case, we can utilize the different hypotheses to first determine whether the current view is unambiguous and subsequently employ them as a confidence measurement in the unambiguous 6D pose. To quantify the effect of this, we report our test results on the unambiguous subset of 'LineMOD' in Fig. 7 (top), where we compute a confidence measure via the standard deviation with respect to the Karcher mean [22].

STD σ	Rot. [°]	Trans. [mm]	VSS [%]	ADD [%]	Rejects [%]
< 0.05	11.8	39.4	80.0	37.7	32.6
< 0.075	13.8	41.3	79.1	35.5	18.2
< 0.10	15.5	43.0	78.3	34.3	10.5
< 0.15	17.3	44.0	77.7	33.4	4.0
$< \infty$	19.2	44.8	77.3	32.7	0.0
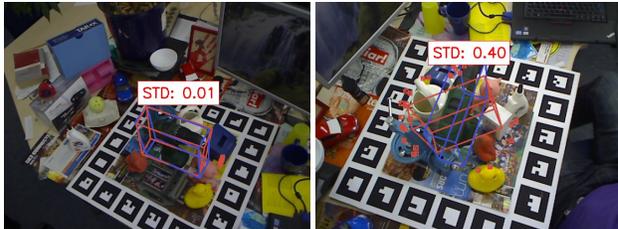
Figure 7: **Reliability.** Top: results for different bins for the standard deviation over all hypotheses for the poses. Bottom: pose with the lowest (left) and the highest (right) standard deviation in the hypotheses. GT pose in blue, predicted pose in red. The red frustums illustrate the hypotheses.

Naturally, a lower standard deviation means more accurate poses. By only allowing poses with $\sigma < 0.1$, all metrics improve, while only losing about 10.5% of all estimates. The rotational error decreases by approximately 20% and the translation error drops from 44.8mm to 43.0mm. Accordingly, using an even lower threshold (*e.g. $\sigma < 0.05$*) gives another significant improvement for pose (especially in rotation), however, at the cost of rejecting more estimates. The qualitative example image in Fig. 7 also confirms these results. The pose with the lowest standard deviation for the 'driller' is very accurate, and the one with the highest is rather imprecise. We experience the same behavior for all *unambiguous* 'LineMOD' objects.

5. Conclusion

We propose a new approach for pose estimation that implicitly models ambiguities without requiring any input preprocessing as well as the feasibility of domain adaptation between synthetic and real data. In addition, we can estimate the axis of rotational ambiguity and perform pose refinement based on clustering without knowing the number of clusters in advance. Our experiments show that our method is suitable for detecting both challenging objects with multiple rotational symmetries and datasets with little ambiguity. Lastly, we argue that our method constitutes a metric of reliability for the 6D pose.

In conclusion, we believe that the new formulation of the pose detection problem from images as an ambiguous task paves the way towards interesting applications in the domain of robotic interactions and automation.

References

[1] Christopher Bingham. An antipodally symmetric distribution on the sphere. *The Annals of Statistics*, pages 1201–1225, 1974.

[2] Tolga Birdal, Umut Simsekli, Mustafa Onur Eken, and Slobodan Ilic. Bayesian pose graph optimization via bingham distributions and tempered geodesic mcmc. In *NeurIPS*, 2018.

[3] Eric Brachmann, Frank Michel, Alexander Krull, Michael Ying Yang, Stefan Gumhold, et al. Uncertainty-driven 6d pose estimation of objects and scenes from a single rgb image. In *CVPR*, 2016.

[4] Benjamin Busam, Tolga Birdal, and Nassir Navab. Camera pose filtering with local regression geodesics on the riemannian manifold of dual quaternions. In *ICCV Workshop*, 2017.

[5] Marcelo Cicconet, Vighnesh Birodkar, Mads Lund, Michael Werman, and Davi Geiger. A convolutional approach to reflection symmetry. *PRL*, 95(1):44–50, 2017.

[6] Dorin Comaniciu, Peter Meer, and Senior Member. Mean shift: A robust approach toward feature space analysis. *TPAMI*, 24:603–619, 2002.

[7] Enric Corona, Kaustav Kundu, and Sanja Fidler. Pose estimation for objects with rotational symmetry. In *IROS*, 2018.

[8] Thanh-Toan Do, Ming Cai, Trung Pham, and Ian D. Reid. Deep-6dpose: Recovering 6d object pose from a single RGB image. *CoRR*, abs/1802.10367, 2018.

[9] Frederik Eaton and Zoubin Ghahramani. Choosing a variable to clamp. In *Artificial Intelligence and Statistics*, 2009.

[10] Mohamed Elawady, Christophe Ducottet, Olivier Alata, Cécile Barat, and Philippe Colantoni. Wavelet-based reflection symmetry detection via textural and color histograms. *ICCV Workshop*, 2017.

[11] Jared Glover and Leslie Pack Kaelbling. Tracking the spin on a ping pong ball with the quaternion bingham filter. In *ICRA*, 2014.

[12] Claus Gramkow. On averaging rotations. *Journal of Mathematical Imaging and Vision*, 15(1-2):7–16, 2001.

[13] Richard Hartley, Khurrum Aftab, and Jochen Trumpf. L1 rotation averaging using the weiszfeld algorithm. In *CVPR*, 2011.

[14] Richard Hartley, Jochen Trumpf, Yuchao Dai, and Hongdong Li. Rotation averaging. *IJCVn*, 103(3):267–305, 2013.

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *ICCV*, 2017.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[17] Stefan Hinterstoisser, Stefan Holzer, Cedric Cagniart, Slobodan Ilic, Kurt Konolige, Nassir Navab, and Vincent Lepetit. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In *ICCV*, 2011.

[18] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *ACCV*, 2013.

[19] Stefan Hinterstoisser, Vincent Lepetit, Paul Wohlhart, and Kurt Konolige. On pre-trained image features and synthetic images for deep learning. In *ECCV*, 2018.

[20] Tomáš Hodan, Pavel Haluza, Štěpán Obdrzalek, Jíí Matas, Manolis Lourakis, and Xenophon Zabulis. T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects. In *WACV*, 2017.

[21] Tomas Hodan, Jiri Matas, and Stepan Obdrzalek. On Evaluation of 6D Object Pose Estimation. In *ECCV Workshop*, 2016.

[22] Hermann Karcher. Riemannian center of mass and mollifier smoothing. *Communications on pure and applied mathematics*, 30(5):509–541, 1977.

[23] Wei Ke, Jie Chen, Jianbin Jiao, Guoying Zhao, and Qixiang Ye. Srn: Side-output residual network for object symmetry detection in the wild. In *CVPR*, 2017.

[24] Wadim Kehl, Fabian Manhardt, Slobodan Ilic, Federico Tombari, and Nassir Navab. SSD-6D: Making RGB-Based 3D Detection and 6D Pose Estimation Great Again. In *ICCV*, 2017.

[25] Wadim Kehl, Fausto Milletari, Federico Tombari, Slobodan Ilic, and Nassir Navab. Deep Learning of Local RGB-D Patches for 3D Object Detection and 6D Pose Estimation. In *ECCV*, 2016.

[26] Iasonas Kokkinos. Ubernet: Training a 'universal' convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *CVPR*, 2017.

[27] Alexander Krull, Eric Brachmann, Frank Michel, Michael Ying Yang, Stefan Gumhold, and Carsten Rother. Learning Analysis-by-Synthesis for 6D Pose Estimation in RGB-D Images. In *ICCV*, 2015.

[28] Gerhard Kurz, Igor Gilitschenski, Simon Julier, and Uwe D Hanebeck. Recursive estimation of orientation based on the bingham distribution. In *FUSION*, 2013.

[29] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *ECCV*, 2018.

[30] Tsung Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014.

[31] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-yang Fu, and Alexander C Berg. SSD : Single Shot MultiBox Detector. In *ECCV*, 2016.

[32] Yuanliu Liu, Zejian Yuan, Badong Chen, Jianru Xue, and Nanning Zheng. Illumination Robust Color Naming via Label Propagation. In *ICCV*, 2015.

[33] Fabian Manhardt, Wadim Kehl, and Adrien Gaidon. Roi-10d: Monocular lifting of 2d detection to 6d pose and metric shape. In *CVPR*, 2019.

[34] Fabian Manhardt, Wadim Kehl, Nassir Navab, and Federico Tombari. Deep model-based 6d pose refinement in rgb. In *ECCV*, 2018.

[35] Maks Ovsjanikov, Jian Sun, and Leonidas Guibas. Global intrinsic symmetries of shapes. *Eurographics Symposium on Geometry Processing*, 27(5):1341–1348, 2008.

[36] Lerrel Pinto and Abhinav Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *ICRA*, 2016.

[37] Mahdi Rad and Vincent Lepetit. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *ICCV*, 2017.

[38] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016.

[39] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

[40] Christian Rupprecht, Iro Laina, Robert DiPietro, Maximilian Baust, Federico Tombari, Nassir Navab, and Gregory D Hager. Learning in an uncertain world: Representing ambiguity through multiple hypotheses. In *ICCV*, 2017.

[41] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR*, abs/1409.1, 2014.

[42] Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, Manuel Brucker, and Rudolph Triebel. Implicit 3d orientation learning for 6d object detection from rgb images. In *ECCV*, 2018.

[43] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *ICLR Workshop*, 2016.

[44] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going Deeper with Convolutions. In *CVPR*, 2015.

[45] Keisuke Tateno, Federico Tombari, Iro Laina, and Nassir Navab. Cnn-slam: Real-time dense monocular slam with learned depth prediction. In *CVPR*, 2017.

[46] Bugra Tekin, Sudipta N. Sinha, and Pascal Fua. Real-time seamless single shot 6d object pose prediction. In *CVPR*, 2018.

[47] Endre Weiszfeld. Sur le point pour lequel la somme des distances de n points donnés est minimum. *Tohoku Mathematical Journal, First Series*, 43:355–386, 1937.

[48] Paul Wohlhart and Vincent Lepetit. Learning Descriptors for Object Recognition and 3D Pose Estimation. In *CVPR*, 2015.

[49] Bin Xu and Zhenzhong Chen. Multi-level fusion based 3d object detection from monocular images. In *CVPR*, 2018.

[50] Jian Yao, Sanja Fidler, and Raquel Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *CVPR*, 2012.

[51] Xiang Yu, Schmidt Tanner, Narayanan Venkatraman, and Fox Dieter. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. In *RSS*, 2018.