SpyroPose: SE(3) Pyramids for Object Pose Distribution Estimation

Rasmus Laurvig Haugaard Frederik Hagelskjær Thorbjørn Mosekjær Iversen
SDU Robotics, University of Southern Denmark
{rlha,frhag,thmi}@mmmi.sdu.dk

Abstract

Object pose estimation is an essential computer vision problem in many robot systems. It is usually approached by estimating a single pose with an associated score, however, a score conveys only little information about uncertainty, making it difficult for downstream manipulation tasks to assess risk. In contrast to pose scores, pose distributions could be used in probabilistic frameworks, allowing downstream tasks to make more informed decisions and ultimately increase system reliability. Pose distributions can have arbitrary complexity which motivates unparameterized distributions, however, until now they have been limited to rotation estimation on SO(3) due to the difficulty in training on and normalizing over SE(3). We propose a novel method, SpyroPose, for pose distribution estimation using an SE(3) pyramid: A hierarchical grid with increasing resolution at deeper levels. The pyramid enables efficient training through importance sampling and real time inference by sparse evaluation. SpyroPose is state-of-the-art on SO(3) distribution estimation, and to the best of our knowledge, we provide the first quantitative results on SE(3) distribution estimation. Pose distributions also open new opportunities for sensor-fusion, and we show a simple multi-view extension of SpyroPose. Project page at spyropose.github.io

1. Introduction

Many tasks in robotics involve manipulation of rigid objects and require that objects' rotations and translations, referred to as the objects' poses, are known. Vision systems are often relied upon to estimate object poses when the environment is relatively uncontrolled, such as when objects lie cluttered on a table or in a bin.

Most of the pose estimation literature has been dedicated to algorithms which provide a single best guess of the pose. Estimating a single pose can be adequate if the estimate is always good enough for the downstream task to succeed, or if the task is allowed to fail sometimes. However, even with an ideal method, there may be too much inherent visual ambiguity in an image to reliably perform the task, and for some tasks, failing is detrimental.

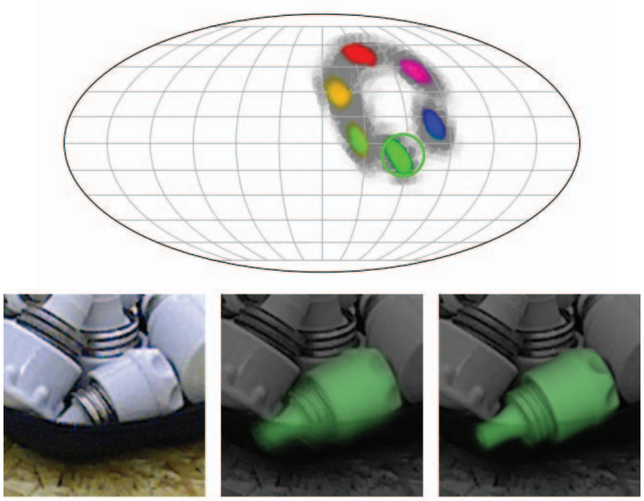

Figure 1. Visualization of SE(3) distributions at different levels of resolution in the pyramid. Bottom: Input image (left) and renders in green of poses weighted by their estimated probabilities for pyramid level three (middle) and five (right). Top: Marginalized SO(3) distribution with two dimensions shown by a Mollweide projection and the last dimension by hue. To show both resolution levels in the same plot, level three is shown in grayscale. The true rotation is indicated by a circle.

Representing uncertainties from vision can facilitate not only sensor-fusion, but also more informed, principled interactions between computer vision and robotics. This has e.g. been shown in [7], where the vision uncertainty, assumed to be normal, and success thresholds on a grasping task are combined to estimate the success of a grasp in a constrained environment. However, visual uncertainties are not always independent and normal, and making these assumptions inhibits probabilistic frameworks at the intersection of robot control and computer vision.

There are many ways to model probability distributions, including mixture models of parameterized functions, histograms, weighted ensembles of particles, and implicit functions. Implicit functions are especially interesting due to their ability to express arbitrary distributions and they have been used successfully to estimate distributions on SO(3) [21] using a contrastive loss with uniformly sampled negatives during training and a uniform grid during inference. Due to the curse of dimensionality, there are two problems with extending this method to SE(3). Firstly, uniformly sampled negatives provide less information as the space grows from SO(3) to SE(3), and secondly, evaluation of a uniform grid on SE(3) becomes prohibitively expensive at any practical resolution.

We present SpyroPose, a novel method for pose distribution estimation, using an SE(3) pyramid, a hierarchical grid, with higher resolution at deeper levels of the pyramid. An example of an estimated pose distribution under a six-fold symmetry ambiguity is shown in Fig. 1.

Our method is based on three key ideas:

- Importance sampling during training, enabled by the pyramid, providing harder negatives and lower variance estimates of the partition function.
- Sparse evaluation of the pyramid at inference, reducing the number of required evaluations by several orders of magnitude, enabling real time pose distribution estimation, even on a CPU.
- Keypoint feature extraction, inducing a camera model bias into the model to enable translational equivariance and to avoid relying on a single latent embedding to represent complex, high-resolution distributions.

We present state-of-the-art rotation distribution estimation results on SYMSOL [21] and TLESS [12], and to the best of our knowledge, we present the first quantitative results on SE(3) distribution estimation.

2. Related Work

Handling visual ambiguities is a challenging part of object pose estimation. Most work [30, 26, 24, 19] on pose estimation uses manual annotations to explicitly handle one cause of visual ambiguity, symmetries, and infers a “best guess” pose estimate without expressing uncertainty. The defacto pose estimation benchmark, BOP [13], is also centered around point estimates and known object symmetries.

Some methods [11, 8] that provide point estimates do however handle visual ambiguities in a principled way. EPOS [11] regresses dense, per pixel, histograms of object surface regions to obtain 2D-3D correspondences and use them in a PnP-RANSAC framework to provide pose estimates without knowing about symmetries a priori. SurfEmb [8] extends this idea and estimates dense full 2D-3D correspondence distributions. However, the dense correspondence histograms and -distributions have not yet shown to be useful to model pose distributions.

There are also pose estimation methods which address pose uncertainties. In [28], model ensembles are used to estimate epistemic uncertainty which indicates a degree of generalization uncertainty caused by insufficient training data or a domain gap. This however does not help represent the inherent aleatoric ambiguities in pose estimation.

Work on modeling the aleatoric uncertainty has traditionally been approached with parametric distributions. The most common parametric model of uncertainty is the multivariate normal distribution. Early work has focused on propagating correspondence uncertainties to pose space for classical pose estimation methods such as for ICP in [1] and for PnP in [4]. They assume that the correspondence ambiguities are independent and Gaussian, however, even simple symmetries and occlusions can break this assumption.

Rotation and position are defined on different manifolds, SO(3) and $\mathbb{R}^3$, respectively, and it has been custom to assume them to be decoupled and treat the two parts separately, usually assuming the position to be normal. For rotational uncertainty, Bingham is the most popular parametric distribution model. Estimating the parameters of a Bingham mixture model has been done in various ways, including deep learning based regression [22, 2, 5] and fitting a Bingham mixture model to an ensemble of pose hypotheses [20]. Additionally, [25] shows that the parameters of a von Mises mixture model can be estimated.

Finally, there are works on unparametric distributions on SO(3). [22, 17] directly regress a rotation histogram which is able to represent arbitrary distributions at a coarse resolution, but the generalization across bins is limited, requiring all bins to be well represented in the training data, making it difficult to scale the histogram to higher resolution, not to mention SE(3). [29, 3] trains a denoising autoencoder on image crops and uses cosine similarities on the learnt latent embeddings to estimate visual ambiguities. However, since the loss is a reconstruction loss rather than a probabilistic loss, and the distribution is designed rather than learnt, the resulting distributions are heuristic.

Recently, a number of methods has been proposed which model the unparametric distributions implicitly. In [15], the marginal distributions of keypoint projections are estimated across the image which can provide a conservative estimate of unnormalized pose likelihoods, however, the capacity of the model is limited because the joint distribution of keypoints is not modeled. Their method could also be used for pose distribution estimation, however, they only present results on rotation distribution estimation due to the lack of a method to normalize over SE(3). ImplicitPDF [21], which has inspired this work, trains a Multi Layer Perceptron (MLP) to map an image embedding and a rotation to an unnormalized likelihood. They show a single qualitative result on SE(3) distribution estimation on simple synthetic data in their supplementary material, however, no quantitative results nor results on real data are shown. HyperPosePDF [14] is similar, but instead of estimating a latent embedding from an image and feeding it to an MLP, they learn a mapping from an image to the weights of an MLP which maps rotations to unnormalized likelihoods. I2S [18] is also similar, but maps features from the image domain to SO(3) followed by SO(3) equivariant layers, before mapping to the unnormalized rotation likelihood. They present good generalization capabilities but lower resolution than the related methods.

Like [21, 14, 18], we also use an implicit formulation to estimate unnormalized log likelihoods, however, we use an SE(3) pyramid, a hierarchical grid, which enables importance sampling during training, allowing efficient learning of unparameterized SE(3) distributions. The same pyramid is used at inference for sparse evaluation to enable real time pose distribution estimation. We also make use of the spatial dimensions of the image in our latent pose distribution embedding, to relieve a single image embedding to represent complex, high-resolution distributions.

3. Methods

At its core, our method is based on learning pose distributions at different levels of resolution. Given a pose hypothesis, we project object keypoints into the image, extract image features at the projected points and feed the sampled features to resolution-specific MLPs. At inference, this allows a sparse top-down evaluation of a pyramid of poses, only expanding the most likely poses to the next, higher-resolution level. During training, having models at different resolutions allows sampling negatives with known probabilities from a pose distribution which is closer to the model distribution than uniform, enabling importance sampling.

3.1. Problem Definition

Given an image crop, $I \in \mathbb{R}^{H\times W\times C}$, of an object of interest, we aim to estimate an unparameterized distribution, $p(x|I)$, of the object's six-dimensional pose, $x \in$ SE(3).

3.2. SE(3) Pyramid Definition

We use an equivolumetric hierarchical grid in SE(3), which we'll refer to as an SE(3) pyramid. Each layer of the pyramid is the cartesian product between a positional grid in $\mathbb{R}^3$ and rotational grid in SO(3).

For the rotational part of the pyramid, we use the HealPix grid [6] extended to SO(3) by [31]. Like previous work [21, 14, 15], we use the grid for its equivolumetric property, but we also use its hierarchical structure. Let $R^{(r)} \subset$ SO(3) denote the grid of rotations at recursion r, and let $R_i^{(r)} \in R^{(r)}$ denote a cell in the grid, represented by its center. The coarsest level, level 0, consists of 72 cells, $|R^{(0)}| = 72$, and for each recursion, each cell is split into eight cells, $|R^{(r)}| = 72 \cdot 8^r$. The volume of the grid is $V(R^{(r)}) = \pi^2$ and because the grid is equivolumetric, a rotation cell has volume $V(R_i^{(r)}) = \pi^2/(72 \cdot 8^r)$.

For the positional part, the bounds of the grid need to be defined, since $\mathbb{R}^3$ is unbounded. We define the bounds in two steps. In the first step, the positional error is modeled using a conservative estimate of visual ambiguity to ensure that the true pose is within the estimated bounds. In the second step, a hierarchical grid is defined such that it fully encompasses the conservatively estimated bounds. Let $\hat{t} \in \mathbb{R}^{3\times 1}$ be a coarse estimate of the object's position, t. Depending on the application, this estimate could come from a detector, be known a priori, or obtained otherwise. In this work, we assume $\hat{t}$ to come from a detector. We then define a convservative bound on t based on $\hat{t}$. Specifically, we presume that the maximum perceived positional ambiguity is equal to the object's radius. Let d be the diameter of the object of interest. Then parallel to the image plane, we let the error be up to $d/2$. Along the view direction, we let the object's distance to the camera be down to half, which assuming a pinhole camera model would cause the appeared size to be approximately doubled.

Formally, we define a bound which meets the above criteria by introducing a truncated multivariate normal variable, $\tilde{e} = \mathcal{N}(0, \sigma I, 1/2)$, where "1/2" indicates, that its truncated at $||\tilde{e}||_2 = 1/2$, and then define our random error variable e as

$$e = A\tilde{e}, \quad A = \begin{bmatrix} d & 0 & \hat{t}_x \\ 0 & d & \hat{t}_y \\ 0 & 0 & \hat{t}_z \end{bmatrix}. \tag{1}$$

The resulting bound of $\tilde{t} = \hat{t} + e$ is a sphere centered around $\hat{t}$ and elongated along the view direction, encompassing more depth- than in-plane ambiguity. We represent a positional cell by its center, and define the positional grid, $p^{(r)} \in \mathbb{R}^{3\times N}$, with $N = 8^r$ at recursion r as

$$p^{(r)} = \hat{t} + Ag^{(r)}, \tag{2}$$

where $g^{(r)} \in \mathbb{R}^{3\times N}$ consists of the centers of the cubes in the 2^r by 2^r by 2^r regular grid inside an origo-centered unit cube. Note that $p^{(r)}$ encompasses $\tilde{t}$, since $g^{(r)}$ encompasses $\tilde{e}$. Also note, that $p^{(r)}$ is hieararchical and equivolumetric, since $g^{(r)}$ is hieararchical and equivolumetric and $g^{(r)} \mapsto \hat{t} + Ag^{(r)}$ is an affine transformation. The volume of the grid is $V(p^{(r)}) = \det(A)$, since the volume of $g^{(r)}$ is 1. The volume of a cell in $p^{(r)}$ is thus $V(p_i^{(r)}) = \det(A)/8^r$.

The SE(3) pyramid is simply the cartesian product of the positional and rotational grid, however, it must be chosen at which recursion to align them. At $R^{(0)}$, the angular distance to the nearest neighbour is approximately $\phi = 1$ rad, causing a visual distance of up to approximately $d/2$. Since $p^{(1)}$ has the same visual resolution, we define recursion 0 of the SE(3) pyramid, $x^{(0)}$, to be the cartesian product of $R^{(0)}$ and $p^{(1)}$: $x^{(0)} = R^{(0)} \times p^{(1)}$. Since $x^{(r)}$ is also an equivolumetric grid, it follows that

$$V(x_i^{(r)}) = \frac{V(x^{(r)})}{|x^{(r)}|} = \frac{\det(A)\pi^2}{(72+8)64^r}. \tag{3}$$

To prevent our models from learning the structure of a fixed SE(3) grid, we randomly offset and rotate $p^{(1)}$ and rotate $R^{(0)}$ during training.

3.3. Contrastive Loss

The InfoNCE loss was presented in [23], inspired by Noise Contrastive Estimation,

$$L_{\text{InfoNCE}} = -\mathop{\mathbb{E}}_{x,I,X}\left[\log \frac{f_\theta(x,I)}{f_\theta(x,I) + \sum_{x_j \in X} f_\theta(x_j, I)}\right], \tag{4}$$

where (x, I) is sampled from the data distribution $p(x, I)$, X is a set of N samples from a noise distribution, $x_j \sim p_n(x)$, and θ are the parameters of the model. They show that for any N, the loss leads to approximating $f_\theta(x, I) \propto p(x|I)p_n(x)^{-1}$, and it follows that letting the noise distribution be uniform, f_θ approximates an unnormalized distribution, $f_\theta(x|I) = \tilde{p}(x|I;\theta) \propto p(x|I)$.

Note that the last term in the denominator of Eq. (4), $\hat{Z} = \sum_{x_j \in X} f_\theta(x_j, I)$, is proportional to an unbiased estimate of the partition function, $\mathbb{E}_X\ \hat{Z} \propto \int f_\theta(x, I)dx$. An inherent problem with scaling the noise contrastive loss with uniform sampling of negatives to higher dimensions is that the variance of the partition function estimate becomes higher, due to the curse of dimensionality.

Importance sampling could be used to lower the variance of the estimate of the partition function, but it requires a heavy-tailed distribution close to $p(x|I;\theta)$ which can be sampled from with known sample likelihoods, and such a distribution is generally not available.

3.4. Pyramid Models & Importance Sampling

We use the loss in Eq.(4) and let the positive sample be the cell in $x^{(r)}$ which encompasses the true pose, x. Instead of learning $\tilde{p}(x|I;\theta)$ at one, high resolution, we learn the distribution at different resolutions, one for each level in the pyramid, $\tilde{\mathrm{p}}(x_i^{(r)}|I;\theta^{(r)})$. For a model at recursion r, the coarser models can then be used to provide an estimate of $\mathrm{p}(x_i^{(r)}|I;\theta^{(r)})$ which can be used for importance sampling.

Let $P(x_i^{(r)}) = x_{i\backslash 64}^{(r-1)}$ be the parent of $x_i^{(r)}$ in the previous recursion, where $\backslash$ denotes integer division, and let $C(x_i^{(r)}) = \left\{x_{64i+0}^{(r+1)}, \dots, x_{64i+63}^{(r+1)}\right\}$ be the set of children of $x_i^{(r)}$ in the next recursion. The siblings of $x_i^{(r)}$, including itself, is thus $S(x_i^{(r)}) = C(P(x_i^{(r)}))$, and let $S(x_i^{(0)}) = x^{(0)}$.

In the following notation, the parameters, $\theta^{(r)}$, and conditioning on I is assumed and left out for clarity. We denote the relative probabilities among siblings as

$$\mathrm{q}(x_i^{(r)}) = \frac{\tilde{\mathrm{p}}(x_i^{(r)})}{\sum_{x_j^{(r)} \in S(x_i^{(r)})} \tilde{\mathrm{p}}(x_j^{(r)})}, \tag{5}$$

and applying them recursively across resolutions results in a generative coarse-to-fine Markov chain model, similar to an auto-regressive model, but where the models are dedicated to resolutions rather than dimensions,

$$\mathrm{p}(x_i^{(r)}) \approx \bar{\mathrm{p}}(x_i^{(r)}) = \mathrm{q}(x_{i\backslash 64^r}^{(0)})\mathrm{q}(x_{i\backslash 64^{r-1}}^{(1)}) \cdots \mathrm{q}(x_i^{(r)}). \tag{6}$$

In the InfoNCE loss in Eq. (4), $\bar{\mathrm{p}}$ can thus be used as an importance sampling distribution,

$$\hat{Z}_{\text{IS}} = \sum_{x_i^{(r)} \in X} \frac{\tilde{\mathrm{p}}(x_i^{(r)})}{\bar{\mathrm{p}}(x_i^{(r)})}, \quad x_i^{(r)} \sim \bar{\mathrm{p}}(x_i^{(r)}), \tag{7}$$

which is proportional to an unbiased estimate of the partition function, like $\hat{Z}$, but with lower variance.

Note that we could maximize the log likelihood directly, normalized by the importance sampling partition function estimate, but initial experiments showed that the additional term in the denominator of Eq. (4) led to more stable training. One intuitive reason is that if the negatives are easy, and $f_\theta(x, I)$ is the dominating term in the denominator of Eq. (4), the gradient is close to zero, $\nabla_\theta L_{\text{InfoNCE}} \approx 0$. Without $f_\theta(x, I)$ in the denominator, this would not be the case.

3.5. Inference with Pyramid Models

At inference, only a sparse tree of the pyramid is evaluated, obtaining highest resolution where the probability is highest. Initially, a distribution over all of the coarsest grid cells is obtained by the coarsest model, $\mathrm{p}(x^{(0)}|I;\theta^{(0)})$. Let $x_k^{(0)}$ denote the top k cells with respect to estimated probabilities, and let p_k^0 denote the sum of probabilities for $x_k^{(0)}$. The top k cells are then expanded to their children, $C(x_k^{(0)})$, which are evaluated by the next model to redistribute the probability p_k^0 with higher resolution:

$$\hat{\mathrm{p}}(x_i^{(r)}) = p_k^{(r-1)} \frac{\tilde{\mathrm{p}}(x_i^{(r)})}{\sum_{x_j^{(r)} \in C(x_k^{(r-1)})} \tilde{\mathrm{p}}(x_j^{(r)})}. \tag{8}$$

This process is repeated until the last recursion, and the cells which have not been expanded, including all cells of the last recursion, are leaf nodes in the sparse tree. The leaf node probabilities sum up to one and make up the estimated distribution. The estimated likelihood of a continuous pose, x, is determined by the leaf node, $x_i^{(r)}$, encompassing x,

$$\hat{p}(x) = \hat{\mathrm{p}}(x_i^{(r)})V(x_i^{(r)})^{-1}. \tag{9}$$

Table 1. Rotation distribution estimation results on SYMSOL. The table entries are estimated log likelihoods ↑ of the true rotation averaged over 5 k test images per object. Results below the gray line is on our implementation of SYMSOL I. For verification, we show Ours w/o KP for both the original and our implementation of the dataset. IS: Importance Sampling. KP: Keypoints.

	SYMSOL I						SYMSOL II			
Method	avg.	cone	cyl.	tet.	cube	ico.	avg.	sphX	cylO	tetX
Prokudin et al. [25] (2018)	-1.87	-3.34	-1.28	-1.86	-0.50	-2.39	0.48	-4.19	4.16	1.48
Gilitschenski et al. [5] (2019)	-0.43	3.84	0.88	-2.29	-2.29	-2.29	3.70	3.32	4.88	2.90
Deng et al. [2] (2020)	-1.48	0.16	-0.95	0.27	-4.44	-2.45	2.57	1.12	2.99	3.61
ImplicitPDF [21] (2021)	4.10	4.45	4.26	5.70	4.81	1.28	7.57	7.30	6.91	8.49
I2S [18] (2023)	3.41	3.75	3.10	4.78	3.27	2.15	4.84	3.74	5.18	5.61
HyperPosePDF [14] (2023)	5.78	5.74	4.73	7.04	6.77	5.10	7.72	7.73	7.12	8.53
Ours w/o KP & IS	5.65	6.77	6.07	6.04	6.23	3.12	7.16	6.96	7.59	6.92
Ours w/o KP	**7.33**	**7.62**	**6.46**	**8.69**	**8.63**	**5.23**	**9.27**	**9.07**	**9.32**	**9.41**
Ours w/o KP	7.12	7.37	6.54	8.39	8.62	4.70				
Ours w/o IS	8.19	7.40	6.69	10.04	8.82	7.99				
Ours w/ cube KP	8.86	7.55	**7.06**	9.60	**10.58**	**9.50**				
Ours	**8.97**	**7.67**	6.96	**11.04**	10.10	9.06				

Note that in contrast to the importance sampling distribution from Eq. (6), which is used during training, during inference, relative probabilities among cells at a given recursion are entirely decided by the model at that recursion and are not affected by the probabilities at earlier recursions. This allows cell-border ambiguities at low resolution to be resolved at higher resolutions.

3.6. Network architecture

We use a UNet [27] with a ResNet18 [10] backbone to obtain a 64 dimensional feature map with the same spatial resolution as the image. From the 3D mesh of the object, 16 approximately evenly spread keypoints are sampled with farthest point sampling. Given a pose at the center of a pose cell, the keypoints are projected into the image and features are extracted from the feature map at the projected points with bilinear interpolation. Keypoints that are projected outside the image receive a learnt out-of-image embedding. The sampled keypoint features are concatenated and fed to a three-layer MLP with 256 hidden neurons. The output of the network is a scalar, representing the estimated unnormalized log likelihood of the pose cell.

4. Experiments

We show results on SO(3) distribution estimation, comparing with previous work, and then to the best of our knowledge, we show the first quantitative results on SE(3) distribution estimation. We show SO(3) results on SYMSOL I, SYMSOL II and TLESS; and SE(3) results on TLESS and HB. Lastly, we show a straight forward multi-view extension of SpyroPose which provides drastic improvements over single-view results, indicating the potential of sensor-fusion using unparameterized distributions.

4.1. SO(3) Results

SYMSOL. ImplicitPDF [21] introduced the synthetic dataset, Symmetric Solids (SYMSOL), for evaluation of distribution estimation on SO(3). The dataset includes a variety of geometric primitives with different kinds of symmetries. The dataset has two parts: SYMSOL I with textureless objects, and SYMSOL II with markers which are only visible in certain rotations, causing dynamic ambiguities.

The dataset does not include camera intrinsics or 3D models of the objects, and while the translation of the objects is fixed, the translation is unknown. Since our method is based on projection of keypoints, we cannot apply it directly to the original SYMSOL dataset. Instead, we apply a modified version of our method without keypoints on the original dataset, and implement SYMSOL I with known camera intrinsics, translation and 3D models, to evaluate our full method.

For our method without keypoints, we use a similar architecture as ImplicitPDF [21], with a ResNet50 [10] to obtain a latent image embedding, a positional encoding of the rotation, and pass both embeddings to a small MLP. See [21] for details. While the architecture of our method without keypoints is similar to ImplicitPDF, our method still has an MLP for each level in the SO(3) pyramid, and results are shown with and without importance sampling. For our implementation of SYMSOL I, we approximately match perspective, scale and shader of the original dataset. Since SYMSOL II has textures, it is not as easy to implement for a fair comparison.

In our full method, we sample keypoints with farthest point sampling from the 3D model, however in some applications, 3D models may not be available, so we also show results where keypoints are chosen at the corners of two

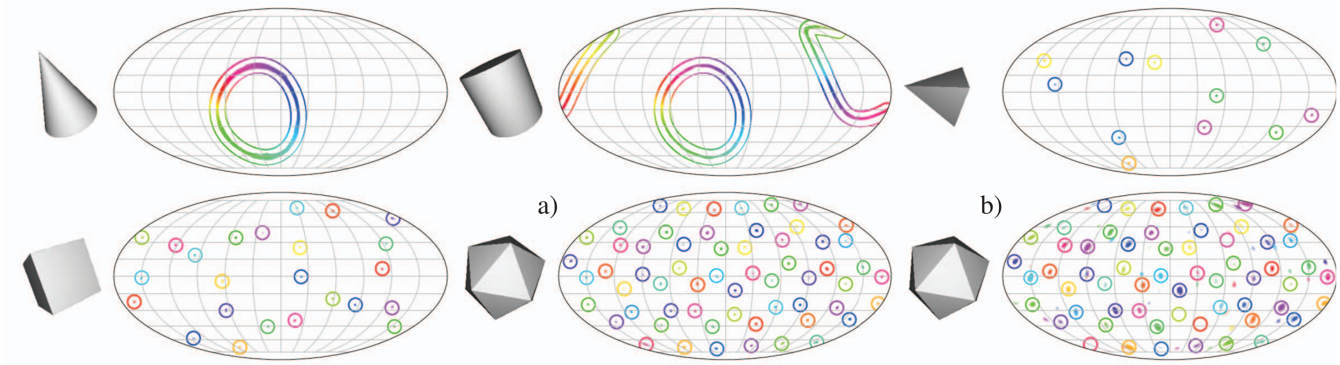

Figure 2. Qualitative SYMSOL I results. We visualize the rotations at the last pyramid level (level 6) and their likelihoods as alpha, normalized for viewing. Circles, or for continuous symmetries donut-like shapes, indicate the correct rotation up to symmetry. a) and b) are from the same image, but b) shows our method w/o KP. Our method accurately captures all 60 modes of the icosahedron.

cubes with side lengths of 1 and 0.5 of the object diameter.

We train object-specific models with seven MLPs ranging from $R^{(0)}$ to $R^{(6)}$, and 1024 negatives per recursion per image. For the models with importance sampling this is obtained with 128 sample trajectories, which with siblings at each recursion amounts to 1024 per MLP, because the branching factor is eight in the rotation pyramid. See Section 3.2 and 3.4. Note that while the training set contains multiple rotation annotations due to symmetry, only the first of the provided annotations per image is used during training, assuming no knowledge about the symmetries. For our full method, we use a batch size of 4. Because the models without keypoints are computationally cheaper, we use a batch size of 16 for those to obtain similar training times. We train all our SO(3) models for 50 k iterations, approximately 2 hours per object on a single RTX 2080.

The results are provided in Table 1, and qualitative examples are shown in Fig. 2. We provide state-of-the-art results on SYMSOL I and SYMSOL II across all objects. Our method predicts 24 and 130 times higher likelihoods on average for the true rotation than HyperPosePDF [14] and ImplicitPDF [21], respectively. Sampling keypoints from the surface of objects assumes that 3D models are available, but our results with cube keypoints perform almost as well, getting rid of this assumption.

Using keypoints provides a big improvement. For our models without keypoints, with architectures similar to ImplicitPDF, a non-spatial latent embedding from the vision model has to express complex and high-resolution distributions. Extracting keypoint features allows the model to use the image-space as an intermediate representation of the distribution and obtain translational equivariance.

Because the evaluation of the SO(3) pyramid is sparse, a distribution down to recursion 6 with 18.8 M rotations only requires 21 k evaluations with $k = 512$, almost three orders of magnitude fewer evaluations than evaluating the full grid. This allows our method to be run in real time, even on CPU, where it runs faster than ImplicitPDF on GPU. See Table 2.

Table 2. Inference time comparison for a single image. For our method we use an Intel i9-9820X CPU and an Nvidia RTX 2080 GPU. Batching improves fps further, obtaining 241 fps on SO(3) for Ours w/o KP on GPU. #eval: number of function evaluations.

Space	Method	grid size	#eval	dev.	fps
SO(3)	ImplicitPDF	2.3 M	2.3 M	gpu	2.4
	Ours w/o KP	18.9 M	21 k	cpu	16.9
				gpu	53.5
	Ours			cpu	3.3
				gpu	48.3
SE(3)	Ours	618 B	164 k	cpu	0.5
				gpu	16.1

While the pyramid allows us to efficiently evaluate beyond recursion 6, we are also SOTA if we evaluate at recursion 5 as ImplicitPDF, with avg. log likelihoods at 8.58, 8.42 and 8.80 for sphX, cylO and tetX, respectively.

Both keypoints and importance sampling improves learning at deeper recursions. See Fig. 3. The log likelihoods without keypoints and importance sampling flatten out around recursion 5, however, our full method could presumably benefit from even more recursions.

ImplicitPDF and HyperPosePDF train one model across all SYMSOL I objects, while we train a model per object, however, we use fewer function evaluations per object during training, and we provide similar improvements on SYMSOL II, where they also train a model per object.

Generalization. I2S [18] applies their method and ImplicitPDF's method in a low-data regime, training on SYMSOL with only 10k images instead of 45k. I2S's ImplicitPDF models perform poorly and seem to have severely overfit, so instead of re-reporting I2S's ImplicitPDF results, we note that Ours w/o KP & IS is very similar to ImplicitPDF and performs similarly on the full dataset. We train

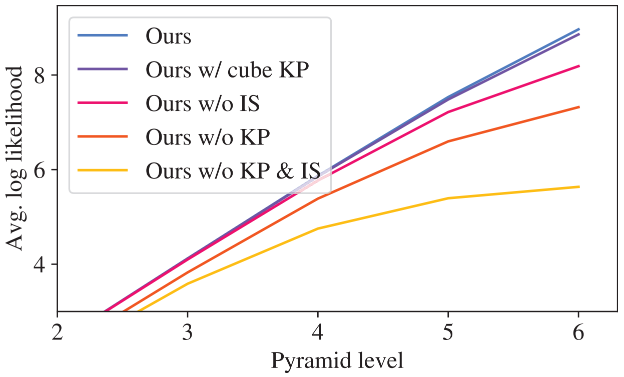

Figure 3. Log likelihoods on SYMSOL I, averaged over objects, at different recursion levels of the pyramid. Both keypoints and importance sampling improves learning at deeper levels.

Table 3. Rotation distribution estimation on SYMSOL II in a low-data-regime with 10k training images instead of 45k. LL ↑.

Method	avg.	sphX	cylO	tetX
I2S [18] (2023)	3.61	3.12	3.87	3.84
Ours w/o KP & IS	3.95	3.56	4.68	3.60
Ours w/o KP	**5.47**	**5.38**	**6.82**	**4.19**

Table 4. Rotation distribution estimation on TLESS.

Method	LL ↑
Prokudin et al. [25] (2018)	8.8
Gilitschenski et al. [5] (2019)	6.9
ImplicitPDF [21] (2021)	9.8
Ours w/o KP & IS	10.3
Ours w/o KP	**11.9**

our models with and without importance sampling on 9.5k of the training images, using the remaining 500 for early stopping. The results are shown in Table 3. Our method outperforms I2S.

TLESS. We follow ImplicitPDF and train a single model across all 30 objects. Results are presented in Table 4. Our method provides almost an order of magnitude higher likelihoods than ImplicitPDF. Note that recursion 5 of our method w/o KP & IS provides an average log likelihood of 9.9, similar to ImplicitPDF. This benchmark uses tight crops of the Kinect training images of singled-out objects, uniform backgrounds and no occlusions, for both training and testing using a random split. The benchmark thus contains no domain gap, no need to generalize wrt. translation or handle occlusions or background clutter. This is in contrast to the experiments in the following section.

4.2. SE(3) results on TLESS and HB

We train SE(3) pyramid models on TLESS [12] and HB [16] down to recursion level 5, selecting four objects from TLESS, representing different levels of symmetries, and four objects from HB, which are common in the pose estimation community. We train on the Physically Based Renders (PBR) from [13] and show results on both held out PBR scenes as well as real images. For TLESS, we show results on the real test images, and for HB, we show results on the real validation images, since test annotations are not publicly available. The results are presented in Table 5, and qualitative examples on TLESS are shown in Fig. 4.

The chosen objects for TLESS have 16 k, 19 k, 23 k and 24 k image crops for training, and for HB: 25 k, 25 k, 24 k and 23 k. Note that we have fewer images per object than in SYMSOL, there's a sim2real gap, and we're attempting to learn distributions on SE(3). For the above reasons, we regularize the models using dropout in the MLPs and heavy data augmentation during training. We use 2048 negatives per recursion corresponding to 32 trajectories through the pyramid using Eq. (6).

Since we are the first to present quantitative results on SE(3) distributions, there are no direct baselines to compare with, but we show results with and without importance sampling as well as results for a uniform distribution on the SE(3) grid. We considered comparing with point estimators, adding gaussians around point estimates, but this could easily be worse than uniform, since they would be punished severely for estimating a wrong mode.

We attempted to compare our joint distribution with a decoupled version, but we were not able to successfully train a rotation model without keypoints, similar to ImplicitPDF, on these more challenging images. In fact, marginalizing SO(3) in our SE(3) distributions from our full method, we get 5.8 avg. log likelihood for the true rotation across the four T-LESS objects, while w/o KP and w/o KP & IS (similar to ImplicitPDF), we get 0.5 and 0.1, respectively. For comparison, -2.3 corresponds to a uniform distribution. Learning joint distributions thus allows an architecture which significantly improves *even* the marginal distributions on this more challenging benchmark.

On the held out PBR images, our model predicts 220 times higher likelihood of the true pose with importance sampling than without, compared to only 2 times higher on SYMSOL I. This is consistent with Fig. 3, which indicates that importance sampling becomes more beneficial when the probability is more concentrated, since evaluation of uniform samples then provides less information.

4.3. Multi-view

Lastly, we show a straight forward application: multi-view pose estimation. Since uncertainty is represented in pose space rather than image space, it is well-suited for principled sensor-fusion, combining information from multiple sources. With multi-view crops and known extrinsics, we let $A = dI$, such that $p^{(r)}$ is a cubic grid, and use the same

Table 5. SE(3) distribution estimation on four representative objects from each of TLESS and HB. Models are trained on synthetic data (PBR), and we present results on both held out PBR images and real images. Entries are avg. log likelihoods of the ground truth poses.

		TLESS					HB				
Data	Method	avg.	1	14	25	27	avg.	2	7	9	21
PBR	Uniform	2.7	3.5	3.2	2.4	1.7	0.8	0.4	0.5	1.4	0.9
	Ours w/o IS	16.7	16.5	16.6	17.8	16.0	15.9	16.3	15.1	16.4	15.8
	Ours	21.6	22.4	20.5	22.6	20.9	21.7	21.7	21.2	21.8	21.9
Real	Uniform	2.7	3.5	3.3	2.4	1.7	0.9	0.6	0.6	1.5	1.0
	Ours w/o IS	16.6	14.1	15.9	19.2	17.2	16.0	18.6	16.5	13.2	15.8
	Ours	18.8	16.9	17.5	20.8	20.0	18.9	21.3	20.6	16.4	17.3

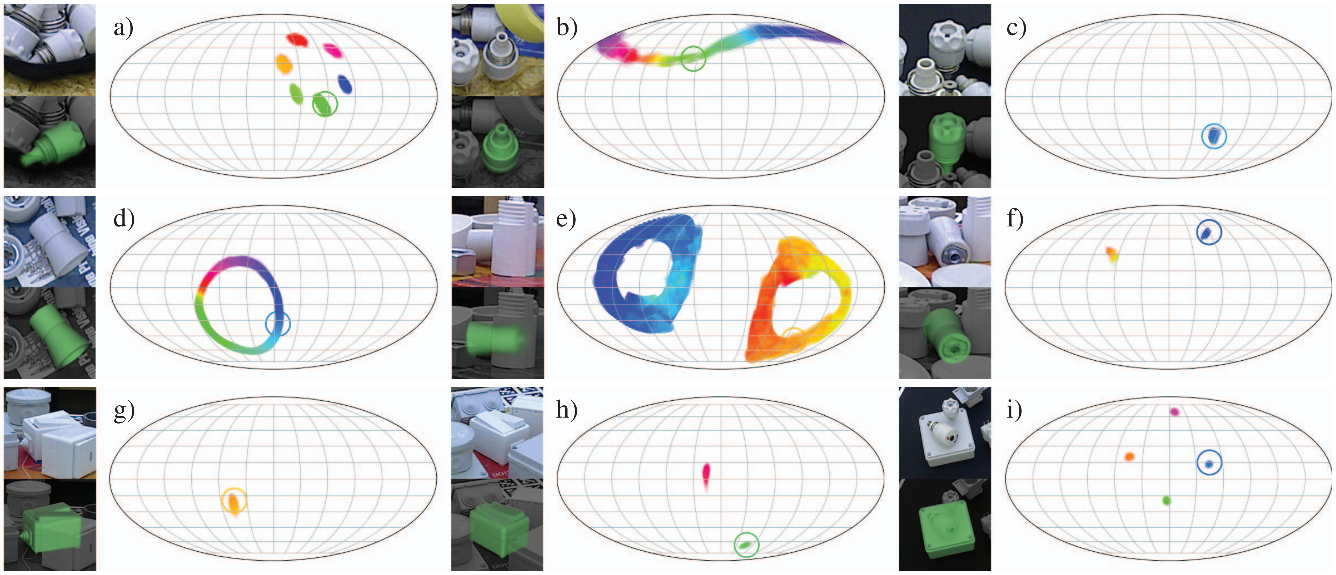

Figure 4. SE(3) distributions on TLESS. First row shows distributions for object 1. a) Six-fold rotational symmetry. b) Continuous rotational symmetry. c) No symmetry. Second row shows distributions for object 14. d) Continuous rotational symmetry. e) The object of interest is behind the foreground object. Two-fold and continuous rotational symmetry. Note that the two discrete modes have different depths, which can only be represented by a joint distribution. f) The continuous rotational symmetry is disambiguated by now visible features at the end of the object, and only a two-fold rotational symmetry along the same axis remains. g) and h) shows no symmetry and a two-fold rotational symmetry, respectively, for object 25. i) shows a four-fold rotational symmetry for object 27.

Table 6. SE(3) distribution estimation results on TLESS. LL↑.

Method	avg.	1	14	25	27
Ours	18.8	16.9	17.5	20.8	20.0
Ours w/ $A = dI$	18.4	16.4	16.6	20.9	19.9
Ours w/ Multi-view	25.2	23.7	23.6	28.2	25.2

grid in a common frame across views. For each recursion in the pyramid at inference, the unnormalized log likelihoods are simply averaged across views. We show multi-view pose distribution estimation results on TLESS in Table 6 with the same sets of up to four views as in [19, 9]. We note that $A = dI$ alone do not improve performance. In fact, it can be harmful to have too much resolution along depth at inference, as most nodes are then spent on representing depth ambiguity. This simple extension increases the likelihood of the true pose by almost three orders of magnitude.

5. Conclusion

This work has proposed SpyroPose, a novel method for pose distribution estimation on SE(3). Our method is based on learning pose distributions at different levels of resolution using a hierarchical SE(3) grid, a pyramid, which enables importance sampling for efficient learning at training time and sparse evaluation at inference, allowing real time pose distribution estimation. Our method outperforms state-of-the-art methods for rotation distribution estimation on SO(3) on the SYMSOL and TLESS datasets, and to the best of our knowledge, we provide the first quantitative results on pose distribution estimation on SE(3). Lastly, with a straight forward multi-view extension of SpyroPose, we have shown how easily pose distributions allow information to be fused from multiple sources, showing great potential for our method as a core component of future work.

References

[1] Andrea Censi. An accurate closed-form estimate of ICP's covariance. In *2007 IEEE International Conference on Robotics and Automation*, pages 3167–3172, Apr. 2007.

[2] Haowen Deng, Mai Bui, Nassir Navab, Leonidas Guibas, Slobodan Ilic, and Tolga Birdal. Deep bingham networks: Dealing with uncertainty and ambiguity in pose estimation. *International Journal of Computer Vision*, pages 1–28, 2022.

[3] Xinke Deng, Arsalan Mousavian, Yu Xiang, Fei Xia, Timothy Bretl, and Dieter Fox. Poserbpf: A rao–blackwellized particle filter for 6-d object pose tracking. *IEEE Transactions on Robotics*, 37(5):1328–1342, 2021.

[4] Marek Franaszek and Geraldine S Cheok. Propagation of orientation uncertainty of 3d rigid object to its points. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2183–2191, 2017.

[5] Igor Gilitschenski, Roshni Sahoo, Wilko Schwarting, Alexander Amini, Sertac Karaman, and Daniela Rus. Deep orientation uncertainty learning based on a bingham loss. In *International Conference on Learning Representations*, 2019.

[6] Krzysztof M Gorski, Eric Hivon, Anthony J Banday, Benjamin D Wandelt, Frode K Hansen, Mstvos Reinecke, and Matthia Bartelmann. Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759, 2005.

[7] Frederik Hagelskjær, Aljaž Kramberger, Adam Wolniakowski, Thiusius Rajeeth Savarimuthu, and Norbert Krüger. Combined optimization of gripper finger design and pose estimation processes for advanced industrial assembly. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2022–2029. IEEE, 2019.

[8] Rasmus Laurvig Haugaard and Anders Glent Buch. Surfemb: Dense and continuous correspondence distributions for object pose estimation with learnt surface embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6749–6758, 2022.

[9] Rasmus Laurvig Haugaard and Thorbjørn Mosekjær Iversen. Multi-view object pose estimation from correspondence distributions and epipolar geometry. *arXiv preprint arXiv:2210.00924*, 2022.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[11] Tomas Hodan, Daniel Barath, and Jiri Matas. Epos: Estimating 6d pose of objects with symmetries. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11703–11712, 2020.

[12] Tomáš Hodan, Pavel Haluza, Štepán Obdržálek, Jiri Matas, Manolis Lourakis, and Xenophon Zabulis. T-less: An rgb-d dataset for 6d pose estimation of texture-less objects. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 880–888. IEEE, 2017.

[13] Tomáš Hodaň, Martin Sundermeyer, Bertram Drost, Yann Labbé, Eric Brachmann, Frank Michel, Carsten Rother, and Jiří Matas. Bop challenge 2020 on 6d object localization. In *European Conference on Computer Vision*, pages 577–594. Springer, 2020.

[14] Timon Höfer, Benjamin Kiefer, Martin Messmer, and Andreas Zell. Hyperposepdf-hypernetworks predicting the probability distribution on so (3). In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2369–2379, 2023.

[15] Thorbjørn Mosekjær Iversen, Rasmus Laurvig Haugaard, and Anders Glent Buch. Ki-pode: Keypoint-based implicit pose distribution estimation of rigid objects. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, page 222. BMVA Press, 2022.

[16] Roman Kaskman, Sergey Zakharov, Ivan Shugurov, and Slobodan Ilic. Homebreweddb: Rgb-d dataset for 6d pose estimation of 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[17] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *Proceedings of the IEEE international conference on computer vision*, pages 1521–1529, 2017.

[18] David M Klee, Ondrej Biza, Robert Platt, and Robin Walters. Image to sphere: Learning equivariant features for efficient pose prediction. *arXiv preprint arXiv:2302.13926*, 2023.

[19] Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. Cosypose: Consistent multi-view multi-object 6d pose estimation. In *European Conference on Computer Vision*, pages 574–591. Springer, 2020.

[20] Fabian Manhardt, Diego Martin Arroyo, Christian Rupprecht, Benjamin Busam, Tolga Birdal, Nassir Navab, and Federico Tombari. Explaining the ambiguity of object detection and 6d pose from visual data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6841–6850, 2019.

[21] Kieran Murphy, Carlos Esteves, Varun Jampani, Srikumar Ramalingam, and Ameesh Makadia. Implicit-pdf: Non-parametric representation of probability distributions on the rotation manifold. *arXiv preprint arXiv:2106.05965*, 2021.

[22] Brian Okorn, Mengyun Xu, Martial Hebert, and David Held. Learning orientation distributions for object pose estimation. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10580–10587. IEEE, 2020.

[23] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[24] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4561–4570, 2019.

[25] Sergey Prokudin, Peter Gehler, and Sebastian Nowozin. Deep directional statistics: Pose estimation with uncertainty quantification. In *Proceedings of the European conference on computer vision (ECCV)*, pages 534–551, 2018.

[26] 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 *Proceedings of the IEEE international conference on computer vision*, pages 3828–3836, 2017.

[27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[28] Guanya Shi, Yifeng Zhu, Jonathan Tremblay, Stan Birchfield, Fabio Ramos, Animashree Anandkumar, and Yuke Zhu. Fast uncertainty quantification for deep object pose estimation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5200–5207. IEEE, 2021.

[29] Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, Manuel Brucker, and Rudolph Triebel. Implicit 3d orientation learning for 6d object detection from rgb images. In *Proceedings of the european conference on computer vision (ECCV)*, pages 699–715, 2018.

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

[31] Anna Yershova, Swati Jain, Steven M Lavalle, and Julie C Mitchell. Generating uniform incremental grids on so (3) using the hopf fibration. *The International journal of robotics research*, 29(7):801–812, 2010.