One-Shot Imitation Learning: A Pose Estimation Perspective

Anonymous Author(s)
Affiliation
Address
email

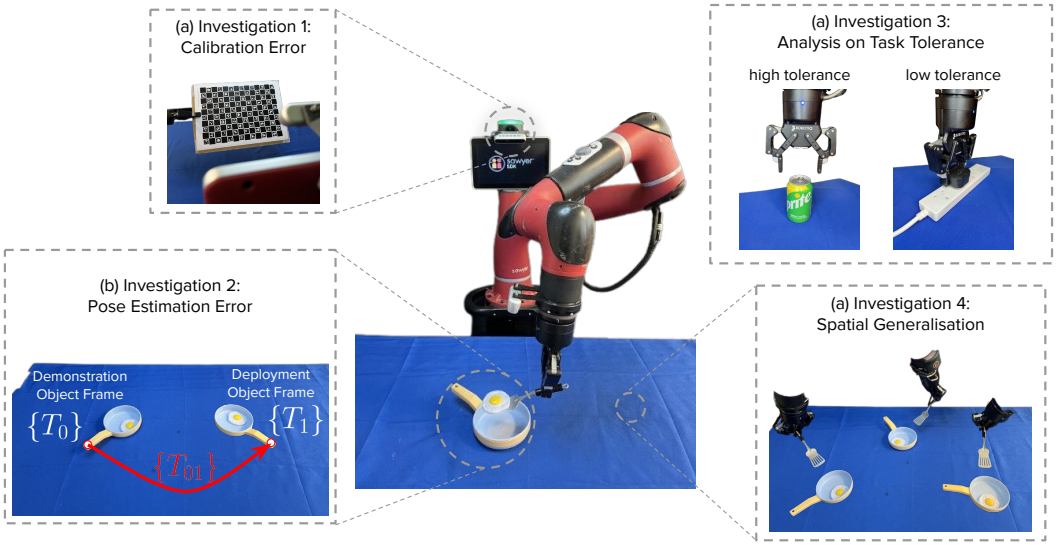

Figure 1: We are going to inspect one-shot imitation learning from the perspective of object pose estimation, through a series of four investigations shown above, under the following assumptions: (1) the robot has no prior task or object knowledge, and (2) the robot does not collect any further data following the single demonstration.

Abstract: Why is one-shot imitation learning so hard? In a quest to answer this, we study imitation learning under the challenging setting when there is (1) only a single demonstration, (2) no further data collection, and (3) no prior task knowledge. We show that in this setting, one-shot imitation learning can be formulated as a trajectory alignment problem, using unseen object pose estimation. And through a series of experiments and a deep dive into the most important design choices under this formulation, our investigation ultimately leads us to a new framework which is able to very efficiently learn diverse, real-world, everyday robot manipulation tasks, each from just a single demonstration. For videos and code we refer the reader to our website https://sites.google.com/view/one-shot-il-posest.

Keywords: One-Shot Imitation Learning, One-Shot Unseen Object Pose Estimation, Robot Manipulation

1 Introduction

Imitation Learning (IL) can be a convenient and intuitive approach to teach a robot how to perform a task. However, many of today's methods for learning vision-based policies require tens to hundreds of demonstrations per task [1, 2, 3, 4, 5, 6]. Whilst fine-tuning with reinforcement learning [7] or pre-training on similar tasks [8, 9, 10] can help, in this paper we take a look at *one-shot imitation learning*, where we assume (1) only a single demonstration, (2) no further data collection following

Submitted to the 7th Conference on Robot Learning (CoRL 2023). Do not distribute.

the demonstration, and (3) no prior task knowledge. But with little past successes in this setting, our paper is then motivated by the following question: **Why is one-shot imitation learning so hard?**

With only a single demonstration and no prior knowledge about the object(s) the robot is interacting with, the optimal imitation is one where the robot and object(s) are aligned in the same way as during the demonstration. For example, imitating a "scoop the egg" task (see Figure 1) could be achieved by aligning the spatula and the egg with the same sequence of relative poses as was provided during the demonstration.

But without any prior knowledge about the object(s), such as 3D object models, the reasoning required by the robot now distills down to an *unseen object pose estimation* problem: the robot must infer the relative pose between its current observation of the object(s) and its observation during the demonstration, in order to perform this trajectory alignment (see Figure 2). Unseen object pose estimation is already a challenging field within the computer vision community [11, 12, 13, 14], and these challenges are further compounded in a robotics setting.

By formulating one-shot imitation learning from the perspective of pose estimation and trajectory alignment, in this paper we are now able to break down the problem into individual challenges. Specifically, we investigate the following four main factors of this formulation when working with real-world scenes: (1) errors in extrinsic camera calibration, (2) errors in unseen object pose estimation, (3) the tolerance of the task, and (4) changes in viewpoint between demonstration and deployment.

Through these four investigations, we reveal new insights into what factors make one-shot imitation learning so hard. But whilst the focus of this paper is a deep dive into these challenges, our investigation ultimately leads us to a new imitation learning framework with very encouraging performance. With real-world experiments on 10 every day tasks, such as inserting a plug into a socket, scooping a toy egg, and putting a plate into a dishwasher, we show how modelling one-shot imitation learning from the perspective of unseen object pose estimation achieves 84% success rate on these everyday tasks. We hope that the lessons learned from these investigations help to guide the community as imitation learning continues to show promise as a viable option for scalable robot learning.

2 Related Work

Whilst there are many methods which study imitation learning with multiple demonstrations per task [3, 15, 16], in this section, we put our paper in the context of existing one-shot IL methods.

Methods that require further data collection Since one demonstration often does not provide sufficient information to satisfactorily learn a task, some methods rely on further data collection. For instance, Coarse-to-Fine approaches [17, 18] train a visual servoing policy by collecting data around the object in a self-supervised manner. On the other hand, FISH [7] fine-tunes a base policy learned with imitation learning using reinforcement learning and interactions with the environment. While these approaches have their strengths, the additional environment interactions require time and sometimes human supervision. In contrast, modelling one-shot imitation learning as unseen object pose estimation avoids the need for real-world data collection, hence enabling scalable learning.

Methods that required prior knowledge Another way of compensating from the lack of demonstration data is to leverage prior task knowledge. For instance, many imitation learning methods require access to object poses [19, 20, 21] which is is often impractical in real-world everyday scenarios. Another dominant approach that assumes prior knowledge for learning tasks from a single demonstration is Meta-learning [8, 9, 10, 22, 23, 24, 25]. In this paradigm, a policy is pretrained on a set of related tasks in order to infer actions for similar tasks from a single demonstration. However, the applicability of the learned policy is limited to tasks closely related to the training set. Contrary to the meta-learning formulation of one-shot imitation learning, approaching it as unseen object pose estimation assumes no prior knowledge, increasing its generality.

Methods which do not require further training or prior knowledge DOME [26] and FlowControl [27] are one-shot imitation learning algorithms that assumes no prior knowledge and can imitate tasks immediately after demonstrations. The main difference between these methods and our framework is that they focus on top-down settings and rely on a wrist-mounted camera. On the contrary, we explore a general one-shot imitation learning framework that relies on an external camera, allowing the robot to constantly have a complete view of the workspace that cannot get occluded by held objects.

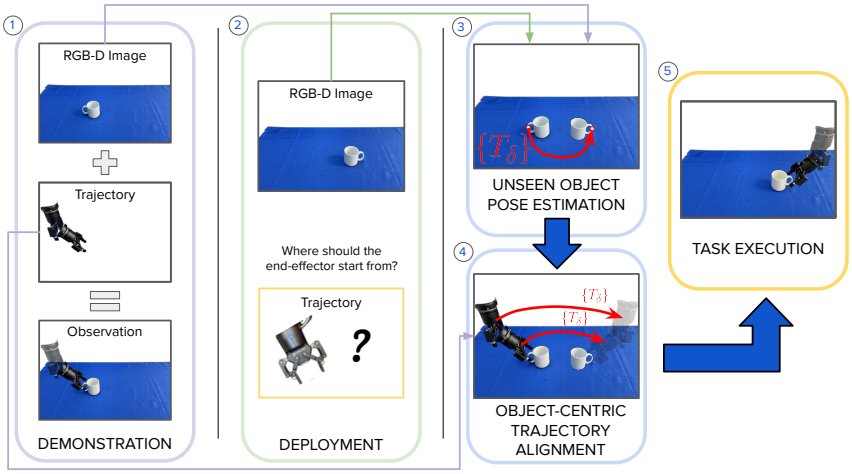

Figure 2: Our framework. (1) The robot receives a demonstration as an RGB-D Image and an end-effector trajectory. (2) At deployment the robot sees the object in a new pose and has to understand how to align the end-effector so that the demonstrated trajectory can be executed. (3) To do so the robot relies on estimating the relative pose of the object between the demonstration and deployment. (4) It can then apply this transformation to the end-effector positions. (5) Ultimately aligning the end-effector to the same object-centric pose experienced at demonstration time.

3 One-shot Imitation Learning for Robotic Manipulation

In this work, we study one-shot imitation learning under the challenging setting when there is: (1) only a single demonstration, (2) no further data collection, and (3) no prior task knowledge. This is an appealing setting to aim for, since it encourages design of a general, efficient method. In this section, we now explore this from the perspective of object pose estimation. First, we provide a formulation of imitation learning using the notation introduced by Osa et al. [28]. Then, we model one-shot imitation learning for manipulation as a trajectory alignment problem. Finally, we introduce the unseen object pose estimation problem which underpins the trajectory alignment problem.

3.1 Imitation Learning Formulation

We observe demonstrations as trajectories $\boldsymbol{\tau} = (\{\boldsymbol{x}_t\}_{t=1}^{T}, \boldsymbol{s})$, where $\boldsymbol{x}$ represents the state of the system, T a finite horizon, and $\boldsymbol{s}$ the context vector. The state $\boldsymbol{x}$ encompasses various measurements relevant to the task, and should include sufficient information for inferring optimal actions. In the context of robotic manipulation, the state could be the object(s) pose(s) and the end-effector (EEF) pose. Similarly, the context vector $\boldsymbol{s}$ captures various information regarding the conditions the demonstration was recorded under, and can assume different forms, ranging from images of the robot's task pace captured before the demonstration, to a description of the task in a multi-task setting.

Given a dataset of demonstrated trajectories $\mathcal{D} = \{\boldsymbol{\tau}_i\}_{i=1}^{N}$, IL aims to learn a policy π^* that satisfies the objective:

$$\pi^* = \arg\min D(q(\boldsymbol{x}), p(\boldsymbol{x})), \tag{1}$$

where $q(\boldsymbol{x})$ and $p(\boldsymbol{x})$ are the distributions of states induced by the demonstrator and policy respectively, and $D(q, p)$ is a similarity measure between q and p.

3.2 Modelling One-Shot Imitation Learning as Trajectory Alignment

In this section, we model one-shot imitation learning as trajectory alignment, that we define as the process, at test time, of moving the EEF to the same set of relative poses, with respect to the object, that it had during the demonstration. Let R define the frame of the robot and E_t of the end-effector at time step t. A homogeneous transformation matrix $\boldsymbol{T}_{AB}$ represents frame B expressed in frame A. During demonstrations, the robot receives instructions through teleoperation or kinesthetic teaching. Demonstrations are defined as

$$\boldsymbol{\tau}^{Demo} = \left(\boldsymbol{X}^{Demo}, I_t^{Demo}\right), \tag{2}$$

and comprise of an RGB-D image I^{Demo} of the scene captured at the beginning of the demonstration using an external camera, and a sequence of EEF poses from the demonstration, $\boldsymbol{X}^{Demo} = \{\boldsymbol{T}_{RE_t}^{Demo}\}_{t=1}^{T}$, expressed in the robot frame.

With only a single demonstration and no prior knowledge about the object the robot is interacting with, the optimal imitation can be considered to be where the robot and object are aligned in the same way as during the demonstration, throughout the task. For example, imitating grasping a mug (see Figure 2), could be achieved by aligning the EEF in such a way that the relative pose between the EEF and mug are the same as during the demonstration. Note that this also holds for more complex trajectories beyond grasping, such as insertion or twisting manoeuvres. Moreover, if the task involves a grasped object, assuming that the latter is fixed and rigidly attached to the gripper, aligning the EEF will also align the grasped object.

Now consider Equation 1 in the context of manipulation, where the optimal policy π^* should result in replicating the demonstrated task state, $\boldsymbol{x} = \boldsymbol{T}_{OE}$, at every timestep during deployment. This demonstrated sequence of EEF poses (expressed in the object frame) is defined as $\boldsymbol{\tau}_O^{Demo} = \{\boldsymbol{T}_{OE_t}^{Demo}\}_{t=1}^{T}$, where

$$\boldsymbol{T}_{OE_t}^{Demo} = \boldsymbol{T}_{OR}^{Demo}\boldsymbol{T}_{RE_t}^{Demo} = \left(\boldsymbol{T}_{RO}^{Demo}\right)^{-1}\boldsymbol{T}_{RE_t}^{Demo}, \tag{3}$$

and $\boldsymbol{T}_{RO}^{Demo}$ is the unknown object pose during the demonstration. Now during deployment, for optimal imitation, π^* should replicate $\boldsymbol{\tau}_O^{Demo}$ given a novel unknown object pose $\boldsymbol{T}_{RO}^{Test}$, i.e. we would like $\boldsymbol{T}_{OE}^{Test} = \boldsymbol{T}_{OE}^{Demo}$ during every time step of the interaction. Hence, the sequence of EEF poses (expressed in the robot frame) that aligns with the demonstration can be defined as $\boldsymbol{\tau}_R^{Test} = \{\boldsymbol{T}_{RE_t}^{Test}\}_{t=1}^{T}$, where

$$\boldsymbol{T}_{RE_t}^{Test} = \boldsymbol{T}_{RO}^{Test}\boldsymbol{T}_{OE_t}^{Demo}. \tag{4}$$

Substituting Equation 3 into Equation 4 yields

$$\boldsymbol{T}_{RE_t}^{Test} = \boldsymbol{T}_{RO}^{Test}\boldsymbol{T}_{OR}^{Demo}\boldsymbol{T}_{RE_t}^{Demo} = {}^{R}\boldsymbol{T}_{\delta}\boldsymbol{T}_{RE_i}^{Demo}, \tag{5}$$

where we defined

$${}^{R}\boldsymbol{T}_{\delta} = \boldsymbol{T}_{RO}^{Test}\boldsymbol{T}_{OR}^{Demo} \tag{6}$$

which represents the transformation of the object between the demonstration and deployment, where we use the superscript to indicate that ${}^{R}\boldsymbol{T}_{\delta}$ is expressed in the robot frame R (note that Equation 6 can be rearranged to get $\boldsymbol{T}_{RO}^{Test} = {}^{R}\boldsymbol{T}_{\delta}\boldsymbol{T}_{RO}^{Demo}$).

This then leads to the crux of our investigation: the trajectory alignment problem, i.e. computing $\boldsymbol{\tau}_R^{Test}$ from $\boldsymbol{\tau}_R^{Demo}$, distills down to the problem of estimating the relative object pose between demonstration and deployment, given a single image of each. Once this pose is estimated, controlling the EEF to follow this trajectory can simple make use of inverse kinematics. And given that we assume no prior object knowledge, the challenge becomes one of one-shot, unseen object pose estimation, an active field in the computer vision community [11, 12, 13, 14].

3.3 One-shot Unseen Object Pose Estimation for Trajectory Alignment

One-shot unseen object pose estimation [29] is concerned with estimating the relative pose of a novel object visible in two images. Formally, let C denote the frame of reference of a camera, and consider one image I^{Demo}, taken when a novel object was at a pose $\boldsymbol{T}_{CO}^{Demo}$, and a second image I^{Test}, taken when that same object was at a pose $\boldsymbol{T}_{CO}^{Test}$. One-shot unseen object pose estimation aims to estimate the relative transformation between the two object poses, ${}^{C}\boldsymbol{T}_{\delta}$, that satisfies $\boldsymbol{T}_{CO}^{Test} = {}^{C}\boldsymbol{T}_{\delta}\boldsymbol{T}_{CO}^{Demo}$, where we use the superscript to indicate that ${}^{C}\boldsymbol{T}_{\delta}$ is applied in the camera frame C. Rearranging this equation yields

$${}^{C}\boldsymbol{T}_{\delta} = \boldsymbol{T}_{CO}^{Test}\left(\boldsymbol{T}_{CO}^{Demo}\right)^{-1} = \boldsymbol{T}_{CO}^{Test}\boldsymbol{T}_{OC}^{Demo}. \tag{7}$$

Comparing Equations 6 and 7 reveals that ${}^{C}\boldsymbol{T}_{\delta}$ and ${}^{R}\boldsymbol{T}_{\delta}$ both represent the relative object pose, but are expressed in different frames of reference. In fact, after estimating ${}^{C}\boldsymbol{T}_{\delta}$ using one-shot unseen object pose estimation, we can calculate ${}^{R}\boldsymbol{T}_{\delta}$ using the following relationship:

$${}^{R}\boldsymbol{T}_{\delta} = \boldsymbol{T}_{RC}{}^{C}\boldsymbol{T}_{\delta}\left(\boldsymbol{T}_{RC}\right)^{-1} = \boldsymbol{T}_{RC}{}^{C}\boldsymbol{T}_{\delta}\boldsymbol{T}_{CR}, \tag{8}$$

where $\boldsymbol{T}_{RC}$ is the pose of the camera in the robot frame (see the Supplementary Material). Hence, the trajectory alignment problem can now be solved by using one-shot unseen object pose estimation to calculate the value of ${}^{C}\boldsymbol{T}_{\delta}$.

Examining Equation 8 reveals that there are two potential sources of error that could degrade the accuracy of ${}^{R}\boldsymbol{T}_{\delta}$ and compromise the performance during deployment. The first source of error is the error in extrinsic camera calibration, while the second source of error is the error in unseen object pose estimation itself. In the next section, we dive deep into the influence of these errors on the trajectory alignment problem from Section 3.2.

4 Sensitivity Analysis of Trajectory Alignment

4.1 Controlled Noise Experiments

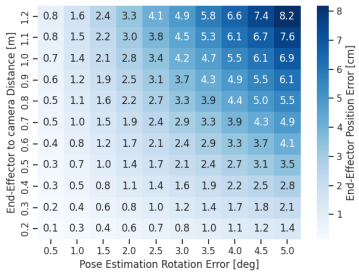

Figure 3: The relationship between the end-effector position error in aligning with a demonstration, the rotation error in ${}^{C}\boldsymbol{T}_{\delta}$, and the initial distance end-effector and the camera.

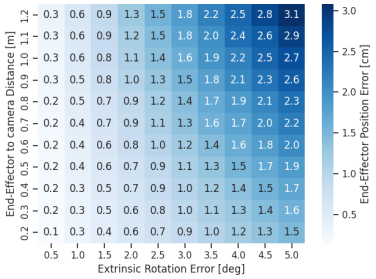

Figure 4: The relationship between the end-effector position error in aligning with a demonstration, the rotation error in the camera extrinsic matrix, and the initial distance end-effector and the camera.

Sensitivity to Error in Extrinsic Camera Calibration We begin by investigating the influence of camera calibration errors on trajectory alignment. To this end, we devise a controlled noise experiment in which we calculate rotation and translation errors in trajectory alignment, given controlled amounts of noise in camera calibration and assuming perfect pose estimation. We summarise this experiment below, and provide further details in the Supplementary Material.

To begin, we define a camera pose $\boldsymbol{T}_{RC}$. We then conduct 1000 iterations, where in each iteration we sample a random end-effector pose $\boldsymbol{T}_{RE}^{Demo}$ and a random object transformation between the demonstration and deployment ${}^{C}\boldsymbol{T}_{\delta}$ expressed in the camera frame. For each iteration, we also add noise to either the translation or the rotation component of camera calibration. Finally, we use the ground truth extrinsics $\boldsymbol{T}_{RC}$ to compute $\boldsymbol{T}_{RE}^{Test}$ from $\boldsymbol{T}_{RE}^{Demo}$ and ${}^{C}\boldsymbol{T}_{\delta}$, while similarly using the noisy extrinsics $\bar{\boldsymbol{T}}_{RC}$ to compute the predicted $\bar{\boldsymbol{T}}_{RE}^{Test}$. We can then compare $\boldsymbol{T}_{RE}^{Test}$ and $\bar{\boldsymbol{T}}_{RE}^{Test}$ to find the translation and rotation error in trajectory alignment, which are defined as

$$t_{\text{error}} = \left|\left|\boldsymbol{t}_{RE}^{Test} - \bar{\boldsymbol{t}}_{RE}^{Test}\right|\right|_2 \qquad R_{\text{error}} = \left|\left|\log\left(\boldsymbol{R}_{RE}^{Test}\left(\bar{\boldsymbol{R}}_{RE}^{Test}\right)^{T}\right)\right|\right|_2 \tag{9}$$

where $\boldsymbol{T}_{RE}^{Test} = [\boldsymbol{R}_{RE}^{Test}|\boldsymbol{t}_{RE}^{Test}]$ and $\bar{\boldsymbol{T}}_{RE}^{Test} = [\bar{\boldsymbol{R}}_{RE}^{Test}|\bar{\boldsymbol{t}}_{RE}^{Test}]$ are the ground truth and predicted EEF pose after trajectory alignment, respectively, and $\log(\cdot)$ is the logarithm map of the $SE(3)$ group.

In Figure 4 we present the results for the influence of rotation errors in camera calibration on the translation error of trajectory alignment. The figure clearly shows that the position error of the EEF depends on both the rotation error in the calibration and the initial EEF-to-camera distance. Hence, for a given calibration rotation error, by placing the camera in close proximity to the robot's workspace we can attenuate the propagation of errors from camera calibration to trajectory alignment. We also note that the other results from this experiment did not provide any major insights and can be therefore found in the Supplementary Material.

Sensitivity to Error in Relative Pose Estimation We now turn our attention to how object pose estimation errors propagate to trajectory alignment. To this end, we conduct an analogous experiment to that in the previous section. However, during each iteration, instead of perturbing $\boldsymbol{T}_{RC}$, we add noise to either the translation or rotation component of the sampled ${}^{C}\boldsymbol{T}_{\delta}$. This allows us to then compute and independently estimate $\boldsymbol{T}_{RE}^{Test}$ from the noise free pose estimate ${}^{C}\boldsymbol{T}_{\delta}$ and $\bar{\boldsymbol{T}}_{RE}^{Test}$ from the noisy pose estimate ${}^{C}\bar{\boldsymbol{T}}_{\delta}$. Once again, we calculate errors in trajectory alignment using Equation 9. Figure 3 presents the results concerning the influence of rotation errors in pose estimates

on the translation error in trajectory alignment. Once again, we observe that the position error is a function of both the pose estimation rotation error and the initial EEF-to-gripper distance. However, by comparing Figures 4 and 3 it becomes apparent that the position error in trajectory alignment is much more sensitive to rotation errors in pose estimation than in camera calibration. The other results, which can be found in the Supplementary Material, also show the same conclusion Finally, we once again observe that the errors propagating from the rotation error in pose estimation can be attenuated by placing the camera close to the task space of the robot.

4.2 Pose Estimation Errors of One-Shot Unseen Object Pose Estimation Methods

	Pose Est. Trans. Error [cm]	Pose Est. Rot. Error [deg]
Class.	$\mathbf{5.9 \pm 11.2}$	$\mathbf{4.3 \pm 10.1}$
ASpan. (FT)	6.0 ± 13.0	6.1 ± 13.4
Reg.	9.8 ± 15.3	9.4 ± 15.3
DINO	11.3 ± 18.2	11.6 ± 18.9
ASpan.	11.5 ± 17.4	11.7 ± 18.8
NOPE	18.8 ± 17.4	18.4 ± 17.2
ICP	14.3 ± 30.8	14.4 ± 32.4
GMFlow	28.4 ± 24.6	27.6 ± 24.5

Table 1: Results for eight different baselines on pose estimation simulated experiments

As previously explained, the proposed formulation attempts to estimate the relative transformation of an object's pose between the demonstration and deployment, and it then uses this estimate to align the end-effector according to the demonstration trajectory. As a result, the precision of the pose estimate will directly influence the quality of the alignment and consequently the success of the task. To study this, we now consider several one-shot unseen object pose estimation methods, and quantify their pose estimation errors. To do so, we generate a simulated dataset consisting of 1100 image pairs of 55 different objects from Google Scan Object Downs et al. [30], using Blender [31]. Our considered baselines include: 1) **ICP**: We use the Open3D [32] implementation of point-to-point ICP [33]. ICP is given a total of 5s to make each estimate, giving it enough time to try $50 - 150$ different initialisations. 2) **GMFlow**: We use GMFlow [34] to estimate correspondences between two RGB images and solve for the relative object pose ${}^{C}\boldsymbol{T}_{\delta}$ using Singular Value Decomposition (SVD) [35]. 3) **DINO**: We use DINO [36] to extract descriptors for pixels in the two RGB images and use the SciPy [37] implementation of the Hungarian algorithm to establish correspondences. Again, the relative pose ${}^{C}\boldsymbol{T}_{\delta}$ is obtained with SVD. 4) **ASpan.**: We use the pretrained ASpanFormer [38] to establish correspondences between two RGB images and estimate ${}^{C}\boldsymbol{T}_{\delta}$ using SVD. 5) **ASpan. (FT)**: The ASpan. baseline, with model weights fine-tuned on a custom object-centric dataset generated in Blender using ShapeNet and Google Scan Objects [39]. 6) **NOPE**: We use the pretrained NOPE [29] model to estimate the relative object rotation from two images, and a heuristic that centres two partial point clouds to predict the relative translation. 7) **Reg.**: We train an object-agnostic PointNet++ to regress relative object orientations around the world's z-axis from two coloured point clouds, using data generated in simulation and domain randomisation. We solve for the relative translation using a heuristic that centres two partial point clouds. 8) **Class.**: This is equivalent to the Reg. baseline with the exception that PointNet++ is trained to classify the relative object orientation. For a more detailed description of these experiments, we refer the reader to the Supplementary Material.

Regarding the latter three models, we note that we also experimented with predicting the relative object translation from pairs of RGB-D images. However, we found that a simpler solution which centres partial point clouds for translation prediction, had a similar performance, and thus we use this for these regression and classification baselines, and only train a network to predict the rotation.

The results for this experiment are shown in Table 1. To gain an intuition for the implications of the errors of the baselines on one-shot imitation learning, consider the Class. one-shot pose estimator. Given a mean rotation error of $4.3°$, from Section 4.1 we know that the lower bound on the error in aligning with demonstrations is 3.5cm for an end-effector 0.5m from the camera during the demonstration, and 6.9cm for an end-effector 1m from the camera during the demonstration.

5 Learning Real-World Tasks with One-Shot Imitation Learning

5.1 Task Tolerance

We now investigate if our formulation of one-shot imitation learning can learn real-world, everyday tasks, of varying tolerances, and we study this across a range of unseen object pose estimation methods.

	Plug	Lid	Toaster	Dishwasher	Mug	Egg	Tea	Cap	Bowls	Can	Mean
GMFlow	10	40	0	20	40	20	20	20	60	60	31.1
ASpan. (FT)	0	10	10	50	60	50	30	50	30	50	37.8
ASpan.	0	10	0	20	50	50	40	50	60	70	38.9
DINO	0	20	10	30	50	60	40	40	80	70	44.4
NOPE	0	10	0	50	0	70	**100**	90	70	**100**	54.4
ICP	10	**70**	80	40	60	80	**100**	**100**	**100**	**100**	81.1
Class.	**20**	10	**90**	**70**	**100**	**90**	**100**	**100**	80	**100**	82.2
Reg.	**20**	30	**90**	**70**	**100**	**90**	**100**	**100**	80	**100**	84.4
Mean	7.5	25	35	43.75	57.5	63.75	66.25	68.75	70	81.25	

Table 2: Real-world success rates (%), from ten trials for each combination of method and task.

Implementation Details For trajectory alignment, we use a given unseen object pose estimator to estimate ${}^{C}\boldsymbol{T}_{\delta}$ and use Equations 8 and 5 alongside inverse kinematics to align the robot with the first state of the demonstration. From this state, we align the full robot trajectory with the demonstrated trajectory by replaying the demonstrated EEF velocities expressed in the EEF frame. To segment RGB images, we use a combination of OWL-ViT [40] and SAM [41]. All segmented RGB-D images are downsampled to ensure compatibility with a given baseline.

Experimental Procedure To this end, we conducted the experiments using a 7-DoF Sawyer robot operating at 30Hz. The robot is equipped with a head-mounted camera capturing 490-by-640 RGB-D images. The task space is defined as a 30×75cm region on a table in front of the robot, which is further divided into ten quadrants measuring 15×15cm each. During the demonstration phase, all objects are positioned in approximately the same location in the task space (see Figure 6), and a single demonstration is provided for each task via kinesthetic teaching. In the testing phase, the object is randomly placed within each quadrant with a random orientation between $\pm 45°$ of the demonstrated orientation. In total, we test each method on a single object pose within each of the quadrants, resulting in ten evaluations per method.

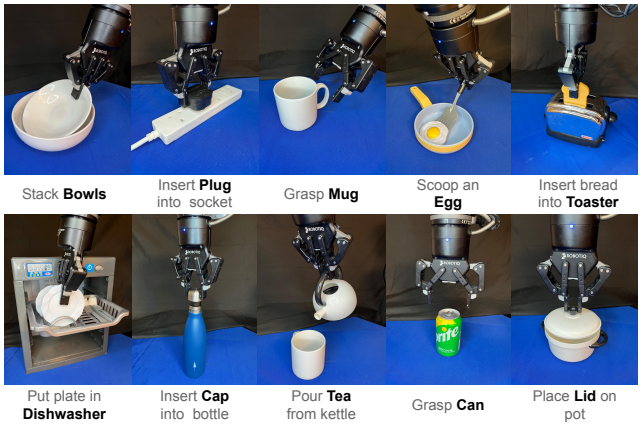

Figure 5: The 10 real-world tasks we evaluate on.

Tasks We choose ten everyday robot manipulation tasks that span a broad range of task tolerances, as depicted in Figure 5. Below we provide a brief description of the tasks, ordered according to their task tolerance, starting with tasks that have a very high tolerance to error and ending on tasks that require very precise trajectory alignment. These tasks are: grasping a can (**Can**), placing one bowl into another (**Bowls**), inserting a cap into a bottle (**Cap**), pouring a marble from a kettle into a mug (**Tea**), scooping an egg from a pan (**Egg**), grasping a mug by the handle (**Mug**), inserting a plate into a specific slot in a dish rack (**Dishwasher**), inserting a toast into a toaster (**Toaster**), placing a lid on top of a pot (**Lid**), and inserting a plug into a socket (**Plug**). Although it may seem that the **Cap** task has a low tolerance to error, this task is compliant when the bottle is not fixed to the table. Also, notably, the plug, lid and toast tasks have task tolerances in the order of several millimetres and require precise alignment with the demonstrations.

Results. The results of the experiment are summarised in Table 2, with tasks ordered by mean success rate across methods and methods ordered by mean success rate across tasks. Based on these results, we conclude that the trajectory alignment formulation of one-shot imitation learning is capable of learning tasks with high and moderate tolerance to errors, and the success rate decreases for the tasks requiring more precision. The best-performing pose estimation method across all tasks was found to be regression, followed by classification and ICP. Note that the fine-tuned ASpanFormer has a considerably worse performance compared to the previous experiment in simulation. This is due to the sim-to-real gap, which we would like to address in future work.

We have also made an observation that the trajectory alignment formulation of one-shot imitation learning is not very suitable for tasks with low tolerances to error, such as the **plug** task. After a qualitative evaluation of predicted pointcloud alignments we attribute this to errors in extrinsic camera calibration (see Section 4.1) and acknowledge this as a limitation of the trajectory alignment formulation of one-shot imitation learning. We also argue that for such task a calibration-free method based on a moving wrist-mounted camera may be more appropriate.

5.2 Spatial Generalisation

Another insight that emerged from the real-world experiments, is the impact that the object's position has on the performance of the pose estimators, and hence, the ability to solve the task. When we aggregate the success rates across all the methods and tasks for each of the quadrants in the task space, we notice a decline in the success rate of one-shot imitation learning as the object position deviates from the demonstrated position. In Figure 6, we display a mug at the approximate location of all demonstrations (labelled as *DEMO*), as well as a mug at the centre of each quadrant. Note that, in this figure, the orientation of the mug is fixed. The opacity of the mugs located in the different quadrants is faded proportionally to the average success rate for those quadrants.

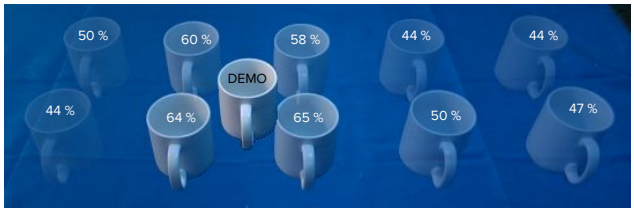

Figure 6: The correlation between success rate and displacement from the place where the demonstration was given.

The cause of this behaviour lies in the *camera perspective*. More specifically, even when kept at a fixed orientation, simply changing the position of an object will result in changes in its visual appearance. Moreover, contrary to the effect of errors in camera calibration (Section 4.1, the changes in visual appearance lessen as the camera is placed further away from the task space. These insights might seem intuitive, but for this same reason they could be easily overlooked by researchers in the field. As a result, for optimal spatial generalisation, **we recommend to provide demonstrations at the centre of the task space**, as this minimises the variations in object appearance when the object is translated from the demonstration pose.

6 Discussion and Limitations

In this section we give a summary of our proposed framework. Shedding light on the valuable insights that we were able to obtain while also exposing some of the limitations of our approach.

By phrasing one-shot imitation learning as an **unseen object pose estimation** problem we are able to use a single demonstration to learn a new task without any further data collection or prior knowledge. We demonstrate this from a theoretical perspective while also showing its potential when applied to real world robot experiments. This being said, **we do not address generalisation to intra-class instances**. However this has been shown to work [42] and could possibly become future work.

Our formulation relies on the concept of **object-centric trajectory alignment**, which should allow the robot to execute a wide range of motions, including twisting or pouring. Nonetheless, **trajectory alignment may struggle for tasks requiring more complex dynamics**. This is a limitation of our approach, although similar related work has shown that trajectory replay was able to scoop up a bag and hammer in a toy nail [17], which are tasks dependent on the dynamics of the movement.

Another limitation of our method is its **reliance on camera calibration**. With this in mind, while investigating this framework we were able to do an **analysis on the effects of pose estimation and camera calibration errors** on both task success rates and spatial generalisation. However, these studies are **tailored to object manipulation**. We are aware that the concept of aligning with an object in the same way as during a demonstration could also be studied more widely, such as in multi-fingered dexterous manipulation or robot navigation. However we believed that a focused study would have resulted in better analyses rather than a more general overview of various ares in robotics.

Lastly, we note when dealing with the three-dimensional space, two-dimensional camera observations do not suffice to unambiguously determine the distance and size of an object. As a result all of our baselines had to **rely on depth data**, which is still not always dependable.

References

[1] S. Young, D. Gandhi, S. Tulsiani, A. Gupta, P. Abbeel, and L. Pinto. Visual imitation made easy, 2020.

[2] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn. BC-z: Zero-shot task generalization with robotic imitation learning. In *5th Annual Conference on Robot Learning*, 2021. URL https://openreview.net/forum?id=8kbp23tSGYv.

[3] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, K.-H. Lee, S. Levine, Y. Lu, U. Malla, D. Manjunath, I. Mordatch, O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch, J. Quiambao, K. Rao, M. Ryoo, G. Salazar, P. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. Tran, V. Vanhoucke, S. Vega, Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich. Rt-1: Robotics transformer for real-world control at scale. In *arXiv preprint arXiv:2212.06817*, 2022.

[4] T. Yu, T. Xiao, A. Stone, J. Tompson, A. Brohan, S. Wang, J. Singh, C. Tan, D. M, J. Peralta, B. Ichter, K. Hausman, and F. Xia. Scaling robot learning with semantically imagined experience. In *arXiv preprint arXiv:2302.11550*, 2023.

[5] A. Stone, T. Xiao, Y. Lu, K. Gopalakrishnan, K.-H. Lee, Q. Vuong, P. Wohlhart, B. Zitkovich, F. Xia, C. Finn, and K. Hausman. Open-world object manipulation using pre-trained vision-language model. In *arXiv preprint*, 2023.

[6] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. *arXiv e-prints*, art. arXiv:2304.13705, Apr. 2023. doi:10.48550/arXiv.2304.13705.

[7] S. Haldar, J. Pari, A. Rai, and L. Pinto. Teach a robot to fish: Versatile imitation from one minute of demonstrations. *arXiv preprint arXiv:2303.01497*, 2023.

[8] Y. Duan, M. Andrychowicz, B. Stadie, O. Jonathan Ho, J. Schneider, I. Sutskever, P. Abbeel, and W. Zaremba. One-shot imitation learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/ba3866600c3540f67c1e9575e213be0a-Paper.pdf.

[9] C. Finn, T. Yu, T. Zhang, P. Abbeel, and S. Levine. One-shot visual imitation learning via meta-learning. In S. Levine, V. Vanhoucke, and K. Goldberg, editors, *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pages 357–368. PMLR, 13–15 Nov 2017. URL https://proceedings.mlr.press/v78/finn17a.html.

[10] T. Yu, C. Finn, A. Xie, S. Dasari, T. Zhang, P. Abbeel, and S. Levine. One-shot imitation from observing humans via domain-adaptive meta-learning. *ArXiv*, abs/1802.01557, 2018.

[11] G. Pitteri, A. Bugeau, S. Ilic, and V. Lepetit. 3d object detection and pose estimation of unseen objects in color images with local surface embeddings. In *Asian Conference on Computer Vision*, 2020.

[12] M. Gou, H. Pan, H. Fang, Z. Liu, C. Lu, and P. Tan. Unseen object 6d pose estimation: A benchmark and baselines. *ArXiv*, abs/2206.11808, 2022.

[13] J. Wu, Y. Wang, and R. Xiong. Unseen object pose estimation via registration. In *2021 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, pages 974–979, 2021. doi:10.1109/RCAR52367.2021.9517491.

[14] K. Park, A. Mousavian, Y. Xiang, and D. Fox. Latentfusion: End-to-end differentiable reconstruction and rendering for unseen object pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[15] S. Reed, K. Zolna, E. Parisotto, S. G. Colmenarejo, A. Novikov, G. Barth-Maron, M. Gimenez, Y. Sulsky, J. Kay, J. T. Springenberg, T. Eccles, J. Bruce, A. Razavi, A. Edwards, N. Heess, Y. Chen, R. Hadsell, O. Vinyals, M. Bordbar, and N. de Freitas. A generalist agent, 2022.

[16] M. Shridhar, L. Manuelli, and D. Fox. Cliport: What and where pathways for robotic manipulation, 2021.

[17] E. Johns. Coarse-to-fine imitation learning: Robot manipulation from a single demonstration. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[18] C. Yu, Z. Cai, H. Pham, and Q.-C. Pham. Siamese convolutional neural network for sub-millimeter-accurate camera pose estimation and visual servoing, 2019.

[19] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal. Dynamical movement primitives: learning attractor models for motor behaviors. *Neural computation*, 25(2):328–373, 2013.

[20] Y. Hu, M. Cui, J. Duan, W. Liu, D. Huang, A. Knoll, and G. Chen. Model predictive optimization for imitation learning from demonstrations. *Robotics and Autonomous Systems*, 163: 104381, 2023.

[21] D. Totsila, K. Chatzilygeroudis, D. Hadjivelichkov, V. Modugno, I. Hatzilygeroudis, and D. Kanoulas. End-to-end stable imitation learning via autonomous neural dynamic policies. *arXiv preprint arXiv:2305.12886*, 2023.

[22] A. Bonardi, S. James, and A. J. Davison. Learning one-shot imitation from humans without humans. *IEEE Robotics and Automation Letters*, 5(2):3533–3539, 2020. doi:10.1109/LRA.2020.2977835.

[23] X. Yang, Y. Peng, W. Li, J. Z. Wen, and D. Zhou. Vision-based one-shot imitation learning supplemented with target recognition via meta learning. In *2021 IEEE International Conference on Mechatronics and Automation (ICMA)*, pages 1008–1013, 2021. doi:10.1109/ICMA52036.2021.9512607.

[24] Z. Mandi, F. Liu, K. Lee, and P. Abbeel. Towards more generalizable one-shot visual imitation learning. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2434–2444, 2022. doi:10.1109/ICRA46639.2022.9812450.

[25] M. Sieb, Z. Xian, A. Huang, O. Kroemer, and K. Fragkiadaki. Graph-structured visual imitation, 2020.

[26] E. Valassakis, G. Papagiannis, N. Di Palo, and E. Johns. Demonstrate once, imitate immediately (dome): Learning visual servoing for one-shot imitation learning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022.

[27] M. Argus, L. Hermann, J. Long, and T. Brox. Flowcontrol: Optical flow based visual servoing, 2020.

[28] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, and J. Peters. *An Algorithmic Perspective on Imitation Learning*. Now Foundations and Trends, 2018.

[29] V. N. Nguyen, T. Groueix, Y. Hu, M. Salzmann, and V. Lepetit. Nope: Novel object pose estimation from a single image. *arXiv preprint arXiv:2303.13612*, 2023.

[30] L. Downs, A. Francis, N. Koenig, B. Kinman, R. Hickman, K. Reymann, T. B. McHugh, and V. Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2553–2560, 2022. doi:10.1109/ICRA46639.2022.9811809.

[31] B. O. Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. URL `http://www.blender.org`.

[32] Q.-Y. Zhou, J. Park, and V. Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018.

[33] P. Besl et al. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. doi:10.1109/34.121791.

[34] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, and D. Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8121–8130, 2022.

[35] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(5):698–700, 1987. doi:10.1109/TPAMI.1987.4767965.

[36] S. Amir, Y. Gandelsman, S. Bagon, and T. Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2021.

[37] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi:10.1038/s41592-019-0686-2.

[38] H. Chen, Z. Luo, L. Zhou, Y. Tian, M. Zhen, T. Fang, D. McKinnon, Y. Tsin, and L. Quan. Aspanformer: Detector-free image matching with adaptive span transformer. *European Conference on Computer Vision (ECCV)*, 2022.

[39] L. Downs, A. Francis, N. Koenig, B. Kinman, R. Hickman, K. Reymann, T. B. McHugh, and V. Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items, 2022.

[40] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy, A. Mahendran, A. Arnab, M. Dehghani, Z. Shen, X. Wang, X. Zhai, T. Kipf, and N. Houlsby. Simple open-vocabulary object detection with vision transformers. *ECCV*, 2022.

[41] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick. Segment anything. *arXiv:2304.02643*, 2023.

[42] D. Hadjivelichkov, S. Zwane, M. P. Deisenroth, L. Agapito, and D. Kanoulas. One-shot transfer of affordance regions? affcorrs!, 2022.