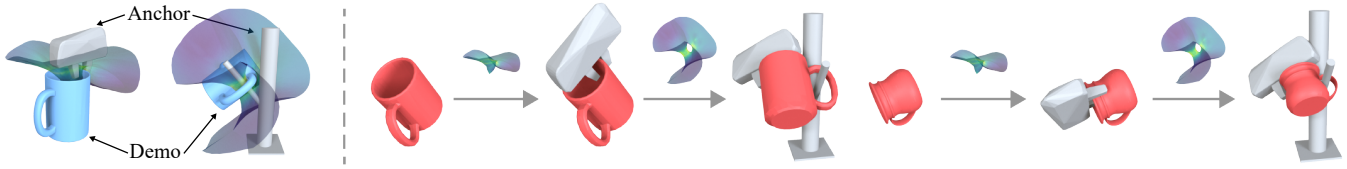
2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

NIFT: Neural Interaction Field and Template for Object Manipulation

Zeyu Huang[1], Juzhan Xu[1], Sisi Dai[2], Kai Xu[2], Hao Zhang[3], Hui Huang[1], Ruizhen Hu[1,*]
[1]Shenzhen University [2]National University of Defense Technology [3]Simon Fraser University
[*]Corresponding Author



(a) Demo interactions and their NIFT (b) Interaction imitations generated using NIFT on two new objects (in red)

Fig. 1: We introduce NIFT, *Neural Interaction Field and Template*, to represent object interactions for imitation learning. Given a few demo interactions (a) for an object manipulation task, NIFT guides the generation of interaction imitations to manipulate new object instances (in red) provided in arbitrary poses (b).

***Abstract*— We introduce NIFT, *Neural Interaction Field and Template*, a descriptive and robust interaction representation of object manipulations to facilitate imitation learning. Given a few object manipulation demos, NIFT guides the generation of the interaction imitation for a new object instance by matching the *Neural Interaction Template* (NIT) extracted from the demos in the target *Neural Interaction Field* (NIF) defined for the new object. Specifically, the NIF is a neural field that encodes the relationship between each spatial point and a given object, where the relative position is defined by a *spherical distance function* rather than occupancies or signed distances, which are commonly adopted by conventional neural fields but less informative. For a given demo interaction, the corresponding NIT is defined by a set of spatial points sampled in the demo NIF with associated neural features. To better capture the interaction, the points are sampled on the Interaction Bisector Surface (IBS), which consists of points that are equidistant to the two interacting objects and has been used extensively for interaction representation. With both point selection and pointwise features defined for better interaction encoding, NIT effectively guides the feature matching in the NIFs of the new object instances such that the relative poses are optimized to realize the manipulation while imitating the demo interactions. Experiments show that our NIFT solution outperforms state-of-the-art imitation learning methods for object manipulation and generalizes better to objects from new categories.**

I. INTRODUCTION

Teaching a robot with few-shot demonstrations has been a long-standing goal in robotics [2, 3, 8, 16, 19, 20, 32]. Ideally, robots should be able to learn from a few demonstrations of a manipulation task and then generalize to new instances of target objects. Especially for object manipulation tasks such as using a robot gripper to pick up a mug or hanging it on a rack (see Figure 1), the gripper and the rack are always fixed, which can be considered as the anchor objects, and the goal is usually to perform similar manipulation or interaction to new object instances to imitate the interactions shown in the demonstrations. If we refer to the source object in the demonstrations as a demo object and the new object instance as the target, our goal then is to generate the interaction between the anchor object and the target so that it is analogous to the interaction between the demo object and the anchor object as shown in the demonstrations.

To optimize the pose of the anchor object with respect to a new target object to mimic the demo interaction, we need to address fundamental questions on how to represent and optimize 3D interactions, and how to measure similarity between the target and the source, or demo, interactions. The most important task is to find the right representation, which should encode rich information of object-object interactions, rather than the individual 3D objects as well as robust against variation in shapes of the interacted objects.

Recent work on neural descriptor fields (NDF) [30] aims to solve this problem with a neural interaction representation and optimization via feature matching in the neural fields. NDF characterizes object interactions by sampling a fixed set of query points, called a Basis Point Set (BPS) [25], around the anchor object. However, BPS has been shown to provide an efficient and compact means to encode features of the anchor object alone, and not the interaction between the anchor and other objects. Moreover, their point-wise features are learned with a network that predicts the occupancy of each point relative to the given object; no spatial relationship between the point and the object is encoded.

Our key contribution is the introduction of *neural interaction field and template (NIFT)* to provide an informative interaction representation and similarity measure that fulfills the criteria mentioned above. Our representation is designed to effectively guide the feature matching for interaction imitation adaptive to the new geometry of a target object.

To encode the open space around the object, we first build the *neural interaction field (NIF)*. For each spatial point, rays are cast from this point in all directions, and the distances to the object in all directions form a normalized spherical

(a) Demo interaction (b) NIT extracted in the demo NIF (c) Target object (d) NIT matched in the target NIF (e) Interaction imitation

Fig. 2: Overview of our interaction imitation method. (a) Given a demo interaction with a robot gripper picking a blue mug, (b) we first represent the interaction with the NIT, which consists of points sampled on the IBS associated with features defined in the NIF of the blue mug. (c) To generate an analogical interaction for a target red mug in a random pose, (d) we perform the interaction imitation by optimizing the global transformation of the NIT to find its best match in the NIF of the red mug. (d) Finally, the gripper is transformed based on the optimized pose of NIT to form the imitated interaction.

function. Then the spherical harmonics expansion of this spherical function is computed to get the rotation-invariant space coverage feature (SCF) [38] of this point relative to the object. However, the computation of the SCF is time-consuming and non-differentiable, so instead of using SCF directly, we train a neural network to predict the pointwise SCF and then use the vector of concatenated activations as the neural feature per point, which constitutes the NIF of a given object. For the given demo interaction, we sample a set of Interaction Bisector Surface (IBS) [36] points that are equidistant to two interacting objects. IBS has been shown to be an informative spatial descriptor of object-object interactions, while robust against shape variations, and thus we denote this set of IBS points associated with the NIF feature as our *neural interaction template (NIT)*. Then given a new target object, our goal is to find the optimal pose of the anchor object together with the NIT in the target NIF with matched features.

We conduct experiments on three pick-and-place tasks to show that NIFT-guided interaction imitation outperforms the state-of-the-art methods by at least a 10% overall success rate boost. With the more descriptive and robust interaction representation, our method can also generalize well to out-of-distribution objects. Ablation studies are also presented to show the importance of both our point selection and neural feature design. Apart from experiments in a simulated environment, we also validate our method on a real robot.

II. RELATED WORK

Our work is related to geometric representations of interactions and imitation learning for manipulation. In this section, we cover prior works most relevant to our method.

Interaction representation. Interaction representations have been extensively studied to encode spatial relations between two or more objects. Relative vector used in most of works for scene generation [1] is relatively simple and thus usually need to be incorporated with other properties together to be able to characterize the spatial relationship between two objects accurately. On the contrary, the IBS [36] provides more detailed and informative interaction representation with the geometric and topological features extracted from the spatial boundary between two objects, which have been used for scene completion and synthesis [37, 38]. The corresponding regions on the interacting objects, denoted as Interaction Region (IR), are further explored for functionality analysis of 3D shapes [14, 15]. Pirk et al. [24] further introduce interaction landscapes to build a spatial and temporal representation of interactions that considers dynamic changes of the interaction between two objects. However, all of these previous works only focus on the interaction analysis between two objects without considering the possibility of the transition from one representation to the other to make it applicable for interaction imitation.

Imitation learning for manipulation. Imitation learning for manipulation has been extensively studied in robotics. Based on different assumptions on the difference between the demo and target objects, the design of the method focuses on different aspects and thus have different levels of generality. For example, pose estimation [28, 35, 39] is the key for known objects, while primitive-based template-matching [13, 27, 34] can be more robust to shape and pose changes. To learn the manipulation policy directly, usually a large number of demonstrations are required [4, 12, 29, 31]. There are also many recent works that use category-level keypoints as an object representation for transferable robotic manipulation [10, 11, 33]. The most related work in this direction is NDF [30], following the idea of using neural shape representation [6, 22, 23, 26]. NDF divides the whole object manipulation task into two stages: interaction synthesis and motion planning, and in the few-shot imitation setting, the interaction synthesis task here is essentially to perform interaction imitation. In this work, we focus on solving the interaction imitation problem with the same pipeline as in NDF [30] but using the proposed NIFT to represent the demo interactions to guide and improve the performance of few-shot imitation learning of 3D interaction.

III. METHOD

Given a demo interaction (O_a, O_s), where O_a is the anchor object and O_s is the source object that is to be replaced with a target object O_t given in a random pose, our goal is to optimize the global pose T of the anchor object O_a w.r.t O_t such that their interaction imitates the demo interaction.

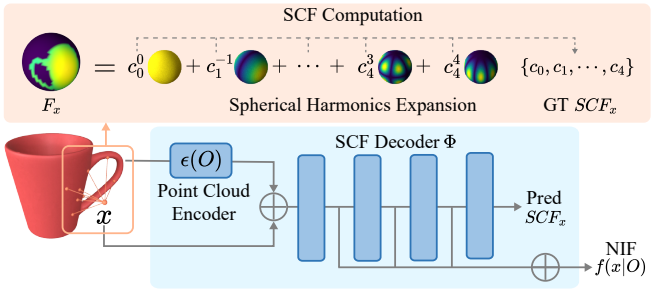


Fig. 3: NIF computation. For each spatial point around the object, we predict its SCF using a neural network and concatenate the activation of the network decoder as its feature in the neural interaction field.

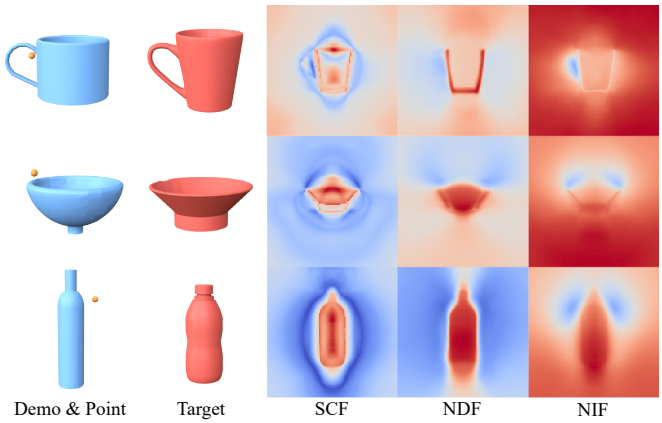


Fig. 4: Comparison of different pointwise features. In each row, for a spatial point located around the source object, we compute the feature differences of this point to all the points around the target object and show the difference map.

An overview of our method is illustrated in Figure 2. Given the demo interaction shown on the left, we first represent the interaction by sampling the IBS points $\mathcal{X}_{sa}$ on the transparent surface in the NIF of the demo object to form the NIT. The global pose of $\mathcal{X}_{sa}$ is then optimized relative to the target object O_t to form an interaction imitation, based on the feature matching in the target NIF. So the key idea of our method is to use the NIT to represent the interaction between O_a and O_s, and further guide the global transformation of the O_a towards O_t based on the feature matching in the target NIF to achieve the interaction imitation.

A. Neural Interaction Field (NIF)

The definition of NIF around a given object is inspired by NDF [30]. As shown in Figure 3, we train a network to predict the spatial feature of a point x relative to the object O. However, instead of predicting occupancy of each query point as in NDF [30], we use more informative SCF [38] to effectively quantify the relationship between the point and the object. The concatenated activations of the network decoder Φ is then used as the point descriptor in the NIF:

$$f(x|O) = \bigoplus_{i=1}^{L} \Phi^i(x, \epsilon(O)), \tag{1}$$

where ϵ is a point cloud encoder, Φ^i is the activation of the i-th layer of the decoder, L is the total number of layers of the decoder, and $\bigoplus$ denotes concatenation. Note that SCF is SO(3)-invariant, and we use the SO(3)-equivariant network architecture proposed in [9] as in NDF [30] to achieve SO(3)-equivariance such that the descriptor of a point x remains constant when its relative position to the object O is fixed, regardless of the change of their global configuration.

In more details, SCF encodes the geometry of the open space around objects in the frequency domain using spherical harmonics. For each spatial point x, rays are cast from x in all directions to compute a normalized spherical function F_x:

$$F_x(\theta, \phi) = \frac{d_{min} + d_{avg}}{d_x(\theta, \phi) + d_{avg}} \tag{2}$$

where $d_x(\theta, \phi)$ is the hit distance of the ray along the direction (θ, ϕ), and d_{min} and d_{avg} are the minimum and mean value of all non-infinity distance of d_x. Then F_x is decomposed into weighted sum of a group of spherical function basis via spherical harmonics expansion [17]:

$$F_x(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{|m| \le l} c_l^m Y_l^m(\theta, \phi) \tag{3}$$

where c_l^m are the spherical harmonics coefficient and Y_l^m are the orthonormalized spherical harmonics at frequency l. The SCF of point x is finally defined as:

$$SCF_x = \{c_0, c_1, ..., c_n\} \tag{4}$$

where $c_l = \|\sum_{|m| \le l} c_l^m Y_l^m\|_2 = \sqrt{\sum_{|m| \le l} (c_l^m)^2}$ is the power of the function at frequency l.

Figure 4 shows some comparisons of different pointwise features, including SCF [38] predicted by the network, NDF [30] based on occupancy, and our NIF based on SCF. For a spatial point around the demo object, we compute the L1 feature differences of this point to all the points around the target object and show a slice of the difference map, where blue indicates smaller difference and red indicates larger difference. We can see that compared to NDF and the predicted SCF, NIF is more distinctive in corresponding to the similar spatial regions of the different objects. Note that as SCF encodes relative distance information in a normalized scale according to Equation 2, our NIF is also robust to the scale differences between the demo and target objects.

B. Neural Interaction Template (NIT)

As the goal is to transfer the anchor object O_a to the target object O_t to imitate the interaction between O_a and O_s, we would like the interaction representation to capture the features of the most important regions related to the interaction. Thus, we opt to sample a set of IBS points $\mathcal{X}_{sa}$ in the demo NIF of O_s to form the NIT representing the demo interaction between O_a and O_s.

Given a demo interaction with a pair of objects (O_a, O_s), IBS [36] is defined as a set of points that are equidistant to

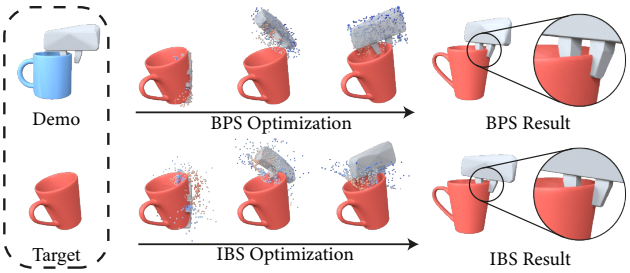


Fig. 5: Comparison of the optimization process using either IBS points or BPS points, both associated with NIF features.

both the objects. To compute IBS, we first sample a set of points on the surfaces or point clouds of the two interacting objects uniformly and compute the Voronoi diagram for all those samples. The IBS is a subset of the Voronoi diagram which lies in between the two objects. Since IBS extends infinitely, we truncate it where it intersects with the bounding sphere of the two objects, and then use an importance-based sampling scheme to sample a set of points on the IBS to be the query point cloud $\mathcal{X}_{sa}$ as in [36]. Intuitively, regions which are closer to the interacting objects are more important for characterizing the interaction and thus more points are sampled from there to give a better abstraction of the IBS.

C. Pose optimization for interaction imitation

The goal of interaction imitation is to find an optimal pose of O_a so that the spatial relationship between O_a and O_t resembles the one between O_a and O_s. With the interaction between O_s and O_a being represented by IBS points $\mathcal{X}_{sa}$ and the associated demo NIF features $\{f(x|O_s)\}$, the goal now becomes to find an optimal pose of $\mathcal{X}_{sa}$ such that the point-wise features conditioned on target NIF $\{f(x|O_t)\}$ are as close to $\{f(x|O_s)\}$ as possible. This is because the relative pose between $\mathcal{X}_{sa}$ and O_a is fixed, and the global pose of O_a is uniquely determined by that of $\mathcal{X}_{sa}$. Thus, the optimization is formulated as:

$$\bar{T} = \arg\min_{T} \sum_{x \in \mathcal{X}_{sa}} \|f(x|O_s) - f(Tx|O_t)\|_1. \quad (5)$$

To optimize the global pose of $\mathcal{X}_{sa}$ relative to O_t, we first move the centroid of $\mathcal{X}_{sa}$ to the centroid of O_t, and then randomly sample a translation t near the origin and a rotation R from the Haar distribution to compose the initial pose $T = (R, t)$. We then optimize the rotation R and translation t to minimize the objective function in Eq. (5) using an iterative optimization solver, in particular, ADAM [18].

Figure 5 shows the comparison of the optimization process using either IBS points or BPS points but with the same pointwise feature NIF. We color-code the points to indicate the point-wise feature distances. We can see that in both cases the gripper (anchor object O_a) gradually moves closer to the mug (target object O_t), guided by the movement of the query points $\mathcal{X}_{sa}$, and finally forms an interaction analogy. Note how the distances of the points are minimized during the optimization, while IBS leads to more accurate interaction without penetration.

D. Few-shot imitation learning

When given more than one demonstration, we incorporate the information extracted from all demo interactions to form the NIT. We first align the IBS points from all demo interactions into a common frame determined by the anchor object, and then perform density-based resampling to keep the points most relevant to the interaction on this anchor. In more detail, we compute a normalized weight for each point based on the average distance to its k nearest neighbors to resample the points. We set k equal to the number of demonstrations in our experiments. We take the resampled points as the query points of NIT to compute the NIF features for each demo object, and the average NIF feature of each point is used as the final pointwise feature of NIT. Once we have an NIT, we can perform imitation learning by performing the optimization explained in Sec. III-C.

IV. EXPERIMENT

A. Experiment setup

Task setup. To evaluate our method, we conduct the same pick-and-place experiments introduced in NDF [30]: 1) Grasp the rim of a *mug* and hang its handle to a rack; 2) Grasp the side of a *bowl* and place it upright on a shelf; and 3) Grasp the neck of a *bottle* and place it upright on a shelf. Given a few demonstrations of the same task with objects initialized in the upright pose, the robot is asked to manipulate unseen objects in similar ways.

Environment setup. Following NDF [30], we create the simulated environment in PyBullet [7] with a two-finger gripper on a table and a RGB-D camera at each corner to obtain the fused point clouds for objects. It is assumed that the object is segmented from the background and the environment remains fixed between demo and test time.

Experiment setup. For each task in the simulated environment, we conduct two groups of experiments with different initial pose settings. For the easy setting, we initialize the test objects with a random upright pose like demonstration. For the hard setting, all objects are initialized in arbitrary SE(3) poses upon the table. We also apply random uniform scaling to all test objects. In each experiment, we provide 10 upright demonstrations and 100 unseen object instances.

Evaluation metrics. Quantitative evaluation is conducted in the simulated environment, and the results are evaluated by the success rates for grasping, placing, and overall processing. A successful grasping means the test object is stably grasped by the arm after disabling the physical collision between the test object and the other objects in the environment. Similarly, a success of placing needs the object to be stably dropped on the rack or shelf after setting the test object to an optimized pose. Note that we do not require the arm to make a successful motion planning from the grasped pose to the pre-place pose for the success of the overall process. As in NDF [30], we use off-the-shelf inverse kinematics and motion planning algorithms to execute the final pick-and-place task based on the predicted relative poses, and the collisions between the gripper and the object are ignored before reaching the target pose.

TABLE I: Comparison of different methods on object manipulation task. CPD uses the demo objects as templates, and other methods use different combinations of query points (BPS and IBS) and pointwise features (NDF, SCF and NIF). Note that the combination of BPS and NDF refers to the method of [30], and our NIT with IBS and NIF is shown in the last row.

Method		Mug			Bowl			Bottle			MEAN		
Point	Feature	Grasp	Place	Overall	Grasp	Place	Overall	Grasp	Place	Overall	Grasp	Place	Overall
Upright Pose													
CPD		0.91	0.50	0.47	0.81	0.99	0.80	0.54	1.00	0.54	0.75	0.83	0.60
	NDF	0.97	0.92	0.89	0.96	0.83	0.81	0.85	0.98	0.85	0.93	0.91	0.85
BPS	SCF	0.97	0.68	0.65	0.92	0.77	0.69	0.45	0.87	0.39	0.78	0.77	0.58
	NIF	0.99	1.00	0.99	**0.98**	**1.00**	**0.98**	0.68	1.00	0.68	0.88	1.00	0.88
	NDF	0.98	0.85	0.84	0.97	1.00	0.97	**0.93**	0.99	**0.93**	**0.96**	0.95	0.91
IBS	SCF	0.96	0.54	0.53	0.97	0.35	0.34	0.46	0.93	0.45	0.8	0.61	0.44
	NIF	**0.99**	**1.00**	**0.99**	0.96	1.00	0.96	0.90	**1.00**	0.90	0.95	**1.00**	**0.95**
Arbitrary Pose													
CPD		0.32	0.34	0.11	0.43	1.00	0.43	0.43	0.97	0.41	0.39	0.77	0.32
	NDF	0.68	0.63	0.47	**0.76**	0.79	0.58	0.55	**1.00**	0.55	0.66	0.81	0.53
BPS	SCF	0.61	0.75	0.43	0.71	0.73	0.51	0.49	0.88	0.41	0.60	0.79	0.45
	NIF	0.65	1.00	0.65	0.67	1.00	0.67	0.46	0.97	0.45	0.59	0.99	0.59
	NDF	0.71	0.54	0.39	0.71	1.00	0.71	0.54	0.98	0.53	0.65	0.84	0.54
IBS	SCF	0.65	0.68	0.44	0.79	0.44	0.31	0.44	0.89	0.36	0.60	0.67	0.37
	NIF	**0.74**	**1.00**	**0.74**	0.75	**1.00**	**0.75**	**0.59**	0.99	**0.59**	**0.69**	**1.00**	**0.69**

Training details. To train networks in our work and baselines, we use the simulation environment to create a dataset with 100,000 point clouds of randomly scaled and posed objects for each object category and train the network across all categories. We sample spatial points uniformly in a box 1.5 times larger than the object's bounding box and compute the SCF and occupancy as ground truth. We use L1 loss for the SCF network and binary cross entropy loss for the occupancy network. The networks have the same architecture. We train all the networks with gradient descent using Adam optimizer [18] with a learning rate 1e-4 for 50 epochs. We use 10% of the dataset as the validation set.The occupancy network achieves 92% in accuracy and the SCF network achieves 86% in mean R^2 score.

Optimization details. To minimize the objective function of few-shot imitation learning, we use the Adam optimizer with learning rate 1e-2 and optimize it for maximum 500 iterations. Following previous work [30], 10 parallel optimizations with different initialization are conducted at the same time, and the optimized pose with the lowest error is taken as the final result. All results are converged within the max iteration in our experiments.

Baselines. We compare our method to two types of baselines. The first set of baselines are methods with the same optimization pipeline as ours but using different combinations of query points and point-wise features. For query points selection, we compare IBS with BPS used in [30]. For the point-wise features, we compare NIF with NDF used in [30] and SCF directly output by the prediction network. The second type of baseline is a more traditional object-matching method following [27]. More specifically, we take all demo objects as templates and register them with the target object using coherent point drift (CPD) [21]. The best registration result is then used to transfer the gripper from the demo object to the target to form the final interaction.

TABLE II: Out-of-domain tests where the source object (Bowl/Bottle) in the demonstration and the testing target object (Mug/Vase) are from different categories.

Method	Bowl:Mug			Bottle:Vase		
	Grasp	Place	Overall	Grasp	Place	Overall
Upright Pose						
CPD	0.89	0.89	0.78	0.52	**1.00**	0.50
NDF	1.00	0.62	0.62	0.72	0.86	0.66
Ours	**1.00**	**0.99**	**0.99**	**0.92**	0.99	**0.92**
Arbitrary Pose						
CPD	0.56	0.94	0.53	0.32	0.89	0.30
NDF	0.66	0.76	0.51	0.37	0.95	0.35
Ours	**0.73**	**1.00**	**0.73**	**0.56**	**0.99**	**0.56**

B. Experiment results

Table I shows the comparison of success rates of our method to different baselines. We can see that methods utilizing neural fields consistently outperform the traditional object-matching method, denoted as CPD. CPD requires an initial coarse alignment to avoid local minima, which leads to apparent performance degradation in the arbitrary pose setting. When checking different combinations of query points (BPS and IBS) and pointwise features (NDF, SCF and NIF), we can see that the combination of IBS and NIF yields the best results, especially for the overall success, confirming the superiority of NIT on interaction representation.

When comparing the baselines with the same pointwise feature but different sets of query points, we see that in most cases IBS produces better results than BPS. When comparing the baselines with the same set of query points but with different pointwise features, we see that the advantage of NIF over NDF is less dominant. We believe the more informative features provided by NIF should be used with points located in important regions rather than points all over the space to provide clearer guidance.

(a) Setup (b) Demonstrations (c) Imiation Results

Fig. 6: Example executions of NIFT for manipulation tasks on real objects. (a) The setup of our experiment with only single RGB-D camera facing the robot. (b) Example demonstrations for the pick-and-place tasks on objects from different categories. (c) Example imitation results on new object instances.

Out-of-domain test. Table II shows the out-of-domain test of our method comparing to NDF [30]. In one experiment, the source and target objects are from different categories but can be applied by the same type of grasp or place action. For example, mugs can be grasped by the rim and placed on the shelf as the bowls, so we can use the bowl demonstration but test on a mug. In the other experiment, the target objects are sampled from an unseen category. We use vases in ShapeNet [5] to imitate the grasping and placing interaction of bottle in the demonstration. We can see that our method generalizes better than the object-matching baseline CPD and the work of [30] in both cases, and we think it is because our NIT can better represent the characteristics of interaction in the demonstrations instead of the specific object. Occupancy is sensitive to the definition of inside and outside of the global shape, while SCF is a local geometric feature of a spatial point which makes it more suitable to represent interactions between objects.

Hyperparameter for SCF computation. Since SCF is the power of spherical harmonic coefficients, its first few orders are related to the spatial structure around the point, while the latter orders represent more details. Table III shows the results when using different SCF orders, n. We can see that the performance is relatively stable but too few orders ($n = 3$) limits the representation power of the SCF. On the other hand, SCF with more orders may introduce confusion by its redundant information. We set $n = 5$ in all our experiments.

Real world execution. Apart from experiments in the simulated environment, we also validate our method on a real robot as shown in Figure 6. We use a UR5 arm with a two-finger gripper to execute object manipulation tasks. To test the robustness of our method, we use only one KinectV2 RGB-D camera in front of the robot to capture a single-view point cloud of the demonstration and test objects in the local coordinate frame of the robot. We also retrain the network with only one viewpoint of the generated dataset in the simulation environment. We record all demonstrations with objects in upright poses and test with unseen objects in the same category. For each test, three demonstrations are used. With only single-view information and a few demonstrations, NIFT is able to guide the interaction imitation on shapes with various geometries. Please check the supplementary video for real-world task execution.

TABLE III: Test with different SCF orders, n.

SCF Order	1	3	5	7	10
Upright Pose	0.92	0.95	**0.95**	0.94	0.93
Arbitrary Pose	0.54	0.65	**0.69**	0.66	0.62

V. CONCLUSIONS

We introduce neural interaction field and template (NIFT) to provide a descriptive and robust representation of demo interactions as well as guide the imitation learning for object manipulation. Experiments show that both point selection and pointwise feature design are essential for the final performance and suggest using the combination of IBS and NIF, leading to our NIT, highly boosts the performance, compared to state-of-the-art methods.

For further investigation, as the NIF we used to compute pointwise features for interaction representation is data-driven, although it's already more informative than NDF as shown in our experiments, it may still not generalize well to objects from unseen categories, so it would be interesting to explore other SE(3)-invariant and differentiable feature designs that are more robust to geometric variations. Moreover, currently we assume that the anchor objects are fixed, e.g., the two-finger gripper and the hanger, we would also like to further explore ways to perform few-shot imitation learning for object manipulation tasks using anchor objects with higher DoF.

VI. ACKNOWLEDGEMENT

We thank the anonymous reviewers for their valuable comments and Sai Raj Perla for proofreading the paper. This work was supported in parts by NSFC(U2001206, U21B2023, 62132021), NSERC (611370), GD Natural Science Foundation (2021B1515020085), GD Talent Program (2019JC05X328), Shenzhen Science and Technology Program (RCYX20210609103121030), DEGP Innovation Team (2022KCXTD025), and Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ).

REFERENCES

[1] Rami Ali Al-Asqhar, Taku Komura, and Myung Geol Choi. "Relationship descriptors for interactive motion adaptation". In: *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 2013, pp. 45–53.

[2] Aris Alissandrakis et al. "An approach for programming robots by demonstration: Generalization across different initial configurations of manipulated objects". In: *2005 International Symposium on Computational Intelligence in Robotics and Automation*. IEEE. 2005, pp. 61–66.

[3] Brenna D Argall et al. "A survey of robot learning from demonstration". In: *Robotics and autonomous systems* 57.5 (2009), pp. 469–483.

[4] Lars Berscheid, Pascal Meißner, and Torsten Kröger. "Self-supervised learning for precise pick-and-place without object model". In: *IEEE Robotics and Automation Letters* 5.3 (2020), pp. 4828–4835.

[5] Angel X Chang et al. "Shapenet: An information-rich 3d model repository". In: *arXiv preprint arXiv:1512.03012* (2015).

[6] Zhiqin Chen and Hao Zhang. "Learning implicit fields for generative shape modeling". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 5939–5948.

[7] Erwin Coumans and Yunfei Bai. *PyBullet, a Python module for physics simulation for games, robotics and machine learning*. `http://pybullet.org`. 2016–2021.

[8] Hao Dang and Peter K Allen. "Robot learning of everyday object manipulations via human demonstration". In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2010, pp. 1284–1289.

[9] Congyue Deng et al. "Vector Neurons: A General Framework for SO (3)-Equivariant Networks". In: *arXiv preprint arXiv:2104.12229* (2021).

[10] Peter R Florence, Lucas Manuelli, and Russ Tedrake. "Dense object nets: Learning dense visual object descriptors by and for robotic manipulation". In: *arXiv preprint arXiv:1806.08756* (2018).

[11] Wei Gao and Russ Tedrake. "kpam-sc: Generalizable manipulation planning using keypoint affordance and shape completion". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2021, pp. 6527–6533.

[12] Marcus Gualtieri, Andreas ten Pas, and Robert Platt. "Pick and place without geometric object models". In: *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE. 2018, pp. 7433–7440.

[13] Kensuke Harada et al. "Probabilistic approach for object bin picking approximated by cylinders". In: *2013 IEEE International Conference on Robotics and Automation*. IEEE. 2013, pp. 3742–3747.

[14] Ruizhen Hu et al. "Interaction context (ICON) towards a geometric functionality descriptor". In: *ACM Transactions on Graphics (TOG)* 34.4 (2015), pp. 1–12.

[15] Ruizhen Hu et al. "Learning how objects function via co-analysis of interactions". In: *ACM Transactions on Graphics (TOG)* 35.4 (2016), pp. 1–13.

[16] Bidan Huang et al. "A modular approach to learning manipulation strategies from human demonstration". In: *Autonomous Robots* 40.5 (2016), pp. 903–927.

[17] Michael Kazhdan, Thomas Funkhouser, and Szymon Rusinkiewicz. "Rotation invariant spherical harmonic representation of 3 d shape descriptors". In: *Symposium on geometry processing*. Vol. 6. 2003, pp. 156–164.

[18] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[19] Nathan Koenig and Maja J Matarić. "Robot life-long task learning from human demonstrations: a Bayesian approach". In: *Autonomous Robots* 41.5 (2017), pp. 1173–1188.

[20] Wenbin Li and Mario Fritz. "Teaching robots the use of human tools from demonstration with non-dexterous end-effectors". In: *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. IEEE. 2015, pp. 547–553.

[21] Andriy Myronenko and Xubo Song. "Point set registration: Coherent point drift". In: *IEEE transactions on pattern analysis and machine intelligence* 32.12 (2010), pp. 2262–2275.

[22] Michael Niemeyer et al. "Occupancy flow: 4d reconstruction by learning particle dynamics". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 5379–5389.

[23] Jeong Joon Park et al. "Deepsdf: Learning continuous signed distance functions for shape representation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 165–174.

[24] Sören Pirk et al. "Understanding and exploiting object interaction landscapes". In: *ACM Transactions on Graphics (TOG)* 36.3 (2017), pp. 1–14.

[25] Sergey Prokudin, Christoph Lassner, and Javier Romero. "Efficient learning on point clouds with basis point sets". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 4332–4341.

[26] Daniel Rebain et al. "Deep medial fields". In: *arXiv preprint arXiv:2106.03804* (2021).

[27] Diego Rodriguez and Sven Behnke. "Transferring category-based functional grasping skills by latent space non-rigid registration". In: *IEEE Robotics and Automation Letters* 3.3 (2018), pp. 2662–2669.

[28] John Schulman et al. "Learning from demonstrations through the use of non-rigid registration". In: *Robotics Research*. Springer, 2016, pp. 339–354.

[29] Qijin She et al. "Learning high-DOF reaching-and-grasping via dynamic representation of gripper-object interaction". In: *ACM Transactions on Graphics (TOG)* 41.4 (2022), pp. 1–14.

[30] Anthony Simeonov et al. "Neural Descriptor Fields: SE (3)-Equivariant Object Representations for Manipulation". In: *arXiv preprint arXiv:2112.05124* (2021).

[31] Shuran Song et al. "Grasping in the wild: Learning 6dof closed-loop grasping from low-cost demonstrations". In: *IEEE Robotics and Automation Letters* 5.3 (2020), pp. 4978–4985.

[32] Halit Bener Suay, Russell Toris, and Sonia Chernova. "A practical comparison of three robot learning from demonstration algorithm". In: *International Journal of Social Robotics* 4.4 (2012), pp. 319–330.

[33] Priya Sundaresan et al. "Learning rope manipulation policies using dense object descriptors trained on synthetic depth data". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2020, pp. 9411–9418.

[34] Skye Thompson, Leslie Pack Kaelbling, and Tomas Lozano-Perez. "Shape-Based Transfer of Generic Skills". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2021, pp. 5996–6002.

[35] Youngrock Yoon, Guilherme N DeSouza, and Avinash C Kak. "Real-time tracking and pose estimation for industrial objects using geometric features". In: *2003 IEEE International conference on robotics and automation (cat. no. 03CH37422)*. Vol. 3. IEEE. 2003, pp. 3473–3478.

[36] Xi Zhao, He Wang, and Taku Komura. "Indexing 3d scenes using the interaction bisector surface". In: *ACM Trans. on Graphics* 33.3 (2014), pp. 1–14.

[37] Xi Zhao et al. "Localization and completion for 3D object interactions". In: *IEEE transactions on visualization and computer graphics* 26.8 (2019), pp. 2634–2644.

[38] Xi Zhao et al. "Relationship templates for creating scene variations". In: *ACM Transactions on Graphics (TOG)* 35.6 (2016), pp. 1–13.

[39] Menglong Zhu et al. "Single image 3D object detection and pose estimation for grasping". In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2014, pp. 3936–3943.