Object-centric Manipulation in Dynamic Environments using Diffused Orientation Fields

Cem Bilaloglu, Tobias Löw, and Sylvain Calinon

***Abstract*—Human environments are filled with objects that exhibit substantial shape variation even within the same category, such as fruits or kitchen utensils. In dynamic settings, a robot will often encounter the same task on the same class of objects with different shapes. For object-centric manipulation skills that require continuous physical interaction along the surface, such as slicing, peeling, or cleaning, this mismatch between the actual and expected shape can lead to failure. To address this, we introduce an object-centric approach that expresses actions in local reference frames adapted to object geometry, enabling shape-invariant task descriptions. Policies represented in these local frames become both robust and adaptable across object instances. We construct the local frames as a smooth orientation field, computed online from raw point clouds and a few keypoints using diffusion processes governed by partial differential equations. We demonstrate that this approach improves transfer of contact-rich tasks such as slicing and peeling across varied objects, enhances robustness under occlusions and keypoint noise, and integrates seamlessly with diverse control paradigms, making it suitable for dynamic, real-world environments.**

I. INTRODUCTION

A variety of approaches have been proposed for object-centric skill representation. Early work defined motions in an object's local reference frame [1] or encoded demonstrations from multiple frames to extract task-relevant variability [2], later extended to dictionaries of coordinate systems such as cylindrical or spherical frames [3]. In contrast, coordinate-free descriptors [4], [5] capture intrinsic shape aspects without reference frames, but they typically overlook surface geometry and inter-category variation. Keypoint-based methods [6], [7] address this by imposing geometric constraints on landmarks and have proven effective for discrete tasks such as pick-and-place. However, they are not ideal for continuous, contact-rich tasks such as cleaning [8], polishing [9], or inspection [10], which demand surface-aware representations. Discrete differential geometry methods [11], [12] incorporate surface geometry but are restricted to mesh inputs and do not leverage keypoint information. Functional correspondence methods [13], [14], [15] use keypoints as landmarks for trajectory transfer, but they remain limited to open-loop surface trajectories and do not extend into free space. More recently, learning-based policies have begun to incorporate diffusion processes for contact-rich manipulation, including Neural Descriptor Fields [16], adaptive compliance [17], and reactive slow-fast visual-tactile policies [18]. While these methods highlight the potential of learning-based approaches, they rely on demonstration data and offline training, and they do not explicitly encode geometric information. In contrast, we propose a model-based spectral geometry approach conditioned on keypoints [6], extending task parameterization across multiple reference frames [2] to the continuous case. Our approach removes the dependence on demonstration data and offline training while explicitly capturing the underlying smooth geometric structure.

Our key insight is that tasks like cleaning or slicing on a planar surface can often be reduced to simple, repetitive motions—such as up/down or back/forth—thanks to the presence of a consistent global frame. In contrast, curved or irregular objects lack such a global reference, making skill specification and transfer significantly more challenging. While oriented keypoints [19] provide sparse, localized geometric structure, they are insufficient for continuous interactions that require smooth, task-consistent motion across the surface and into free space. To enable such interactions, we aim to construct a smoothly varying representation of local frames that extends keypoints throughout the object's surface and surrounding workspace. These frames act as a geometric scaffold for expressing manipulation actions. For instance, a slicing or a probing skill might be described as *"slide along the object, go down and go up "*: an instruction that remains well-defined across object instances, categories, and poses. We call such pose- and shape-invariant task descriptions local action primitives. We summarize our object-centric manipulation approach using local reference frames and local action primitives in Figure 1.

Our contributions are:

- expressing object-centric manipulation skills as shape-invariant action sequences in local reference frames
- representing local reference frames as a smooth orientation field conditioned on the point cloud and keypoints collected online
- formulating and computing the smooth orientation field using surface and workspace diffusion processes

These contributions enable us to represent manipulation skills as shape-invariant, interpretable building blocks that remain valid even as object geometry changes.

II. METHOD

We express object-centric manipulation skills as simple action sequences in object-centric local reference frames, as illustrated in Figure 1. For that purpose we represent the local reference frames as a *smooth, geometry-aware orientation field* across the workspace $\Omega \subset \mathbb{R}^3$ conditioned on objects surfaces collected as point clouds $P = \{\mathbf{x}_i\}_{i=1}^{N}$. We formulate the

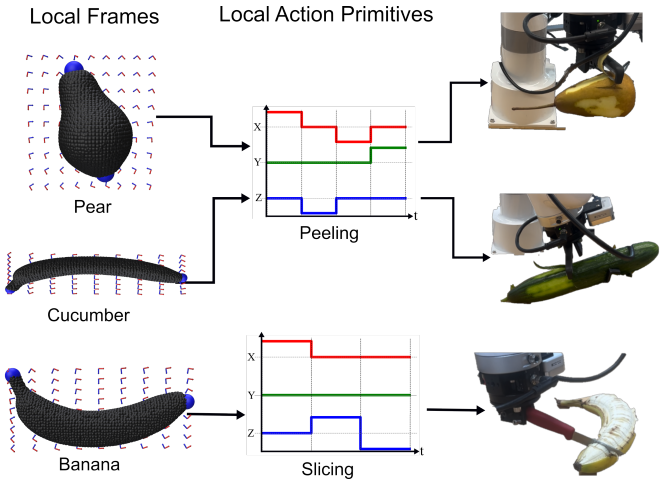

Fig. 1: Overview of our method. *Left*, local reference frames (red and blue arrows) computed from the object point cloud (black) collected online and conditioned on the keypoints (blue). *Center*, local action sequences providing shape-invariant task descriptions for peeling and slicing. *Right*, real-world experiments for peeling and slicing.

smoothness of a field in terms of minimizing the *Dirichlet energy*

$$E[u] = \int_{\Omega} \|\nabla u\|^2 \, d\mathbf{x}, \tag{1}$$

whose Euler–Lagrange equation is Laplace's equation

$$\Delta u = 0 \quad \text{in } \Omega, \tag{2}$$

For controlled smoothness one can use the diffusion equation corresponding to the gradient flow of the Dirichlet energy

$$\dot{u} = \Delta u \quad \text{in } \Omega. \tag{3}$$

We constrain the diffusion process to the object surface by computing the weak Laplacian matrix $\boldsymbol{C}$ and mass matrix $\boldsymbol{M}$ of the point cloud by considering zero-Neumann boundary conditions. The implicit diffusion step is then given by

$$\boldsymbol{u}_\tau = (\boldsymbol{M} - \tau \boldsymbol{C})^{-1} \boldsymbol{M} \boldsymbol{u}_0, \tag{4}$$

where $\boldsymbol{u}_0$ denotes the initial condition encoding the keypoints, $\boldsymbol{u}_\tau$ the corresponding diffused values, both are of size equal to the number of points N, and τ is the diffusion time parameter controlling smoothness. The gradient of the diffused field yields a tangent vector field $\mathbf{u} = \nabla \boldsymbol{u}_\tau$. We then combine the tangent $\mathbf{u}$ and the surface normal $\mathbf{n}$ fields to obtain smooth surface orientation field.

Many tasks require motions that begin in free space and transition into contact and potentially penetrates the object. To extend orientations beyond the surface, we need to solve the diffusion on the workspace conditioned on the surface orientation field. For that purpose, we adopt a grid-free Monte Carlo method called the Walk-on-Spheres (WoS) algorithm [21]. Starting from a query point $\mathbf{x}_q \in \Omega$, WoS simulates a random process where a particle repeatedly jumps to a random point on the sphere of maximum radius that remains inside the domain. The process continues until the particle reaches the boundary (the object surface), where the boundary value is retrieved. To obtain the desired value at a query point, we compute the mean of the boundary values. In the case of the diffused orientation field, this requires averaging the orientations of the boundary points. However, orientations do not lie in a vector space. To address this, we represent the orientations at the boundary points $\mathbf{x}_i$ as unit quaternions for averaging $\mathcal{Q} = \{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n\}$ and employ the method of Markley et al. [22], which solves the following optimization problem:

$$\overline{\mathbf{q}} = \arg\max_{\mathbf{q} \in \mathbb{S}^3} \mathbf{q}^\top \mathbf{M} \mathbf{q}, \tag{5}$$

where the matrix $\mathbf{M}$ is constructed using the outer product of the unit quaternions:

$$\mathbf{M} \triangleq \sum_{\mathbf{q} \in \mathcal{Q}} \mathbf{q}\mathbf{q}^\top. \tag{6}$$

Note that (5) is equivalent to maximizing the Rayleigh quotient and its solution is given by the eigenvector of $\boldsymbol{M}$ with the largest eigenvalue

$$\mathbf{M}\overline{\mathbf{q}} = \lambda_{\max} \overline{\mathbf{q}}. \tag{7}$$

III. EXPERIMENTS

A. Transfer across Object Instances

We show real-world slicing, and peeling experiments in Figure 1. To evaluate task transfer quantitatively, we transferred the peeling task to 50 randomly deformed versions of the Pear object from the YCB dataset [23]. We applied anisotropic scaling with scale factors uniformly sampled from the interval $[0.6, 1.4]$ along each axes. To further break axial symmetry, we applied a twist deformation by rotating each point $\mathbf{p} = (x, y, z)$ around the vertical axis (z-axis) by an angle $\theta = \alpha z$, where α is a twist strength parameter sampled uniformly from the interval $[-10, 10]$. We computed the diffused orientation field (DOF) for each deformed object instance and reused the local action primitives from the peeling task to execute two consecutive peeling motions. We show the aligned action trajectories in Figure 3.

As shown in Figure 3 (i) representing actions in a single body-fixed reference frame, even if pose-invariant, still results in high standard deviation since actions include object-specific geometric information. Moreover, although the task is highly structured and periodic (consisting of two peeling motions), this pattern is not reflected when the actions are resolved in body-fixed frame. Learning from such data is difficult and might lead to overfitting object-specific patterns that should not be transferred. In contrast, Figure 3 (ii) shows that by encoding the surface's geometry through DOF representing local object-centric reference frames, the standard deviation becomes lower as the task description becomes shape-invariant. This enables the transfer of only the task-relevant structure, independent of the specific object shape. Note that in the local z-axis, the standard deviation is higher when compared to other axes because we control the distance to the surface.

Fig. 2: Overview of local reference frame computation. We use Spot [20] as our canonical point cloud throughout the paper as it provides a sufficiently complex object with a well-defined symmetry axis for cross-section views. (a) Our method takes a point cloud and keypoints collected at runtime as its input. (b) We compute a surface orientation field conditioned on the keypoints by solving the diffusion PDE on the point cloud. We visualize the orientation field by using local reference frames. We show x-axis in red, y-axis in green and omit the z-axis for clarity. Next, we solve the diffusion conditioned on the surface orientation field to obtain the diffused orienation field on the robot's workspace. (c) Diffused orientation field represents smoothly varying local reference frames across the workspace by considering the object's surface geometry.

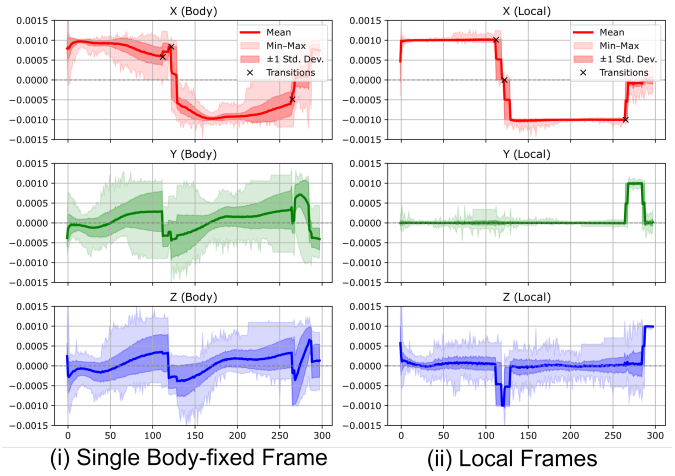

Fig. 3: Mean and the standard deviation of the actions (velocity) calculated from the transferred peeling trajectories across object instances. Actions are expressed in (i) single body-fixed frame, (ii) local reference frames provided by DOF.

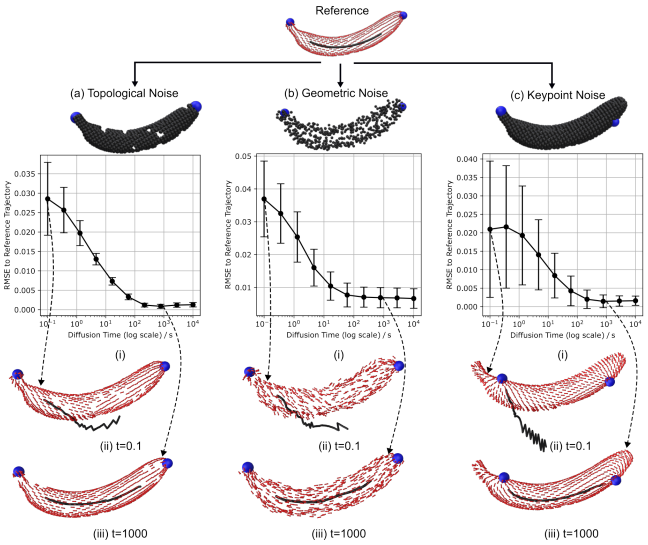

Fig. 4: Robustness to (a) topological, (b) geometric and (c) keypoint noise given in columns with the reference Banana [23] point cloud and trajectory on top. (i) Error bar plots showing that robustness to noise increases with the increased diffusion time. (ii and iii) Two instances with noisy inputs from the experiments visualizing the effect of the short and long diffusion times. Red arrows are the x-axes of the local reference frames, the black curves are the transferred trajectories and the blue points are the keypoints.

B. Robustness to Noise

Operation in dynamic environments requires working with online-acquired point clouds rather than relying on idealized meshes. Such sensed models inherently introduce uncertainty in perception. To evaluate robustness under these conditions, we applied three types of perturbations to the point clouds: (i) topological noise through occlusions and missing regions, (ii) geometric noise via Gaussian perturbations to point positions, and (iii) keypoint noise by jittering annotated keypoints. Figure 4 shows that longer diffusion times consistently smooth the orientation fields, improving trajectory consistency under all perturbations. This confirms that diffusion acts as a tunable mechanism for trading off local detail against global robustness, making DOF well-suited for noisy and incomplete sensory input.

C. Integration with Control Paradigms

DOF serves as an intermediate representation that is agnostic to the generation of high-level control commands. Once computed for a given object, it can be seamlessly integrated with different controllers, enabling adaptation to the object's surface and reducing control complexity, as illustrated in Figure 5 (a).

In trajectory optimization, we used DOF to define cost

functions and their gradients based on the distance to the object's surface and geodesic distances to surface regions encoded using keypoints. These components enabled gradient-based optimization to compute robot trajectories that both maintain a desired distance from the surface and reach target regions while avoiding obstacles, as illustrated in Figure 5 (b) (i, ii). For simplicity, we adopted a batch formulation of the iterative linear quadratic regulator (iLQR), modeling the robot as a velocity-controlled point mass. To further improve convergence, we warm-started the optimization using geodesic shooting (i.e., initializing the trajectory by following the x-axes of local reference frames represented by DOF), as shown in Figure 5 (b) (iii, iv). We evaluated this setup across ten different initial conditions and plotted the number of iterations required for convergence in Figure 5 (c).

Results show that with DOF-based warm-starting, convergence is typically achieved in a single iteration, whereas without warm-starting, at least five to six iterations are required. Qualitative observations of the initial and final trajectories further support this finding. In essence, DOF provides a near-optimal solution for distance tracking, reaching, and surface-based obstacle avoidance purely from geometric information.

In our teleoperation experiments, we used DOF to assist the operator by automatically aligning the tool based on the object's surface and task-specific keypoints. We employed the 3DConnexion Space Mouse, a 6-degree-of-freedom input device, and mapped its control axes to the object-centric local reference frames represented by DOF. Motion along the input device's x-axis translated to movements that approached or retreated from keypoints while maintaining a constant distance from the surface. Motion along the z-axis controlled approach or retreat relative to the object, and motion along the y-axis preserved the distance to both the keypoints and the surface. Throughout the task, the desired tool orientation, relative to the surface and task direction, was maintained. This setup enabled intuitive, surface-aware teleoperation, as illustrated in Figure 5 (d).

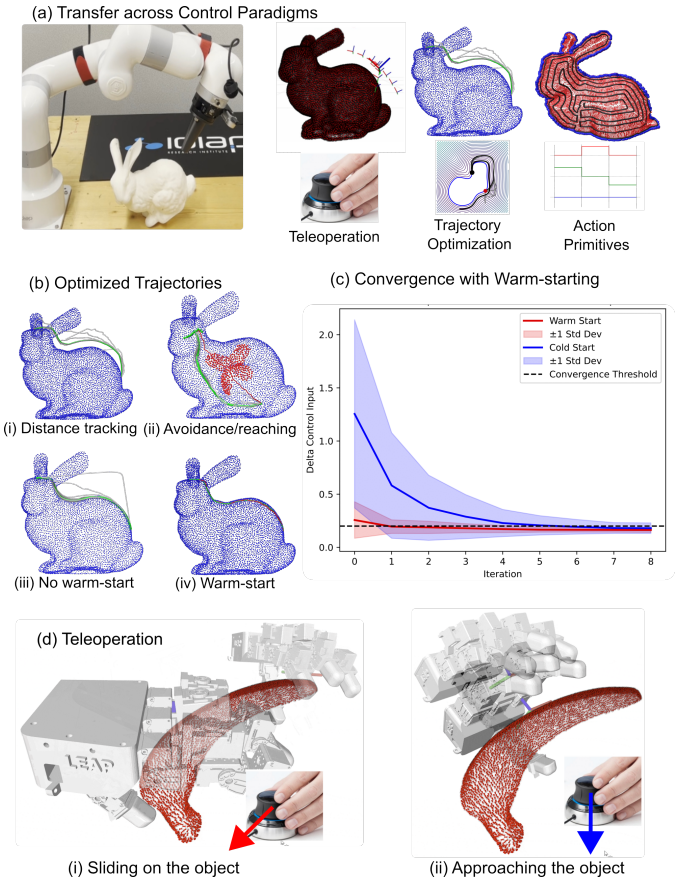

Fig. 5: Transfer across controllers using DOF. (a) Once built, DOF provides a controller agnostic intermediate representation which can be queried by teleoperation, trajectory optimization and action primitive controllers. (b) Trajectory optimization experiments for (i) distance tracking, (ii) target reaching and obstacle avoidance, (iii) reaching without warm-starting and (iv) reaching with warm-starting using the DOF. (c) Norm of the change in the control commands, which is used as the convergence criteria for the trajectory optimization showing the effect of warm-starting. (d) Teleoperation using the LEAP hand [24] and a space mouse, where the input axes are mapped to local frames. Moving along x-axis (red arrow) slides the tool along the surface, while z-axis (blue arrow) approaches the surface.

IV. CONCLUSION & FUTURE WORK

While our results demonstrate that Diffused Orientation Fields (DOF) provide a practical representation for object-centric manipulation in dynamic environments, several important extensions remain.

Our entire pipeline is differentiable with respect to both the task keypoints and the diffusion time parameter [25], [26]. This opens the door for integration into learning-based frameworks where these parameters can be optimized jointly with task performance. A natural extension is to incorporate DOF into a differentiable learning loop, where keypoint selection and diffusion time are learned end-to-end using supervision from demonstrations or task-specific objectives.

In the context of reinforcement learning, the local reference frames defined by DOF can be used to express observations, actions, and rewards. By expressing everything relative to the object's shape, the geometry becomes the reference, not the world. This pose and shape-invariance can lead to more data-efficient learning and improved generalization across different instances of a task. For example, a slicing or peeling policy learned in one object's local frame can be directly reused or fine-tuned on another object, simply by reusing the same local action representation. Similarly, demonstrations collected from different objects or scenes can be normalized into a common local representation, enabling more effective learning from fewer demonstrations. This also opens the door to combining demonstrations from heterogeneous objects into a unified policy, as their local descriptions become comparable under the DOF structure.

References

[1] A. L. P. Ureche, K. Umezawa, Y. Nakamura, and A. Billard, "Task Parameterization Using Continuous Constraints Extracted From Human Demonstrations," *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1458–1471, Dec. 2015. DOI: 10.1109/TRO.2015.2495003.

[2] S. Calinon, "A tutorial on task-parameterized movement learning and retrieval," *Intelligent Service Robotics*, vol. 9, no. 1, pp. 1–29, Jan. 2016. DOI: 10.1007/s11370-015-0187-9.

[3] B. Ti, A. Razmjoo, Y. Gao, J. Zhao, and S. Calinon, "A geometric optimal control approach for imitation and generalization of manipulation skills," *Robotics and Autonomous Systems*, vol. 164, p. 104 413, Jun. 2023. DOI: 10.1016/j.robot.2023.104413.

[4] M. Vochten, T. De Laet, and J. De Schutter, "Generalizing demonstrated motion trajectories using coordinate-free shape descriptors," *Robotics and Autonomous Systems*, vol. 122, p. 103 291, Dec. 2019. DOI: 10.1016/j.robot.2019.103291.

[5] P. So et al., "CITR: A Coordinate-Invariant Task Representation for Robotic Manipulation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, Yokohama, Japan: IEEE, May 2024, pp. 17 501–17 507. DOI: 10.1109/ICRA57147.2024.10611312.

[6] L. Manuelli, W. Gao, P. Florence, and R. Tedrake, *kPAM: KeyPoint Affordances for Category-Level Robotic Manipulation*, Oct. 2019. DOI: 10.48550/arXiv.1903.06684.

[7] J. Gao, Z. Tao, N. Jaquier, and T. Asfour, "K-VIL: Keypoints-Based Visual Imitation Learning," *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3888–3908, Oct. 2023. DOI: 10.1109/TRO.2023.3286074.

[8] C. Bilaloglu, T. Löw, and S. Calinon, "Tactile Ergodic Coverage on Curved Surfaces," *IEEE Transactions on Robotics*, vol. 41, pp. 1421–1435, 2025. DOI: 10.1109/TRO.2025.3532513.

[9] S. Schneyer, A. Sachtler, T. Eiband, and K. Nottensteiner, "Segmentation and Coverage Planning of Freeform Geometries for Robotic Surface Finishing," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 5267–5274, Aug. 2023. DOI: 10.1109/LRA.2023.3293309.

[10] Z. Jiang et al., "Precise Repositioning of Robotic Ultrasound: Improving Registration-based Motion Compensation using Ultrasound Confidence Optimization," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–11, 2022. DOI: 10.1109/TIM.2022.3200360.

[11] M. Dyck, A. Sachtler, J. Klodmann, and A. Albu-Schaffer, "Impedance Control on Arbitrary Surfaces for Ultrasound Scanning Using Discrete Differential Geometry," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7738–7746, Jul. 2022. DOI: 10.1109/LRA.2022.3184800.

[12] M. D. Vedove, F. J. Abu-Dakka, L. Palopoli, D. Fontanelli, and M. Saveriano, *MeshDMP: Motion Planning on Discrete Manifolds using Dynamic Movement Primitives*, Oct. 2024. DOI: 10.48550/arXiv.2410.15123.

[13] M. Ovsjanikov, M. Ben-Chen, J. Solomon, A. Butscher, and L. Guibas, "Functional maps: A flexible representation of maps between shapes," *ACM Transactions on Graphics*, vol. 31, no. 4, pp. 1–11, Aug. 2012. DOI: 10.1145/2185520.2185526.

[14] C. de Farias, B. Tamadazte, R. Stolkin, and N. Marturi, *Grasp Transfer for Deformable Objects by Functional Map Correspondence*, Mar. 2022.

[15] C. de Farias et al., *Geometrically-Aware One-Shot Skill Transfer of Category-Level Objects*, Mar. 2025. DOI: 10.48550/arXiv.2503.15371.

[16] A. Simeonov et al., "Neural Descriptor Fields: SE(3)-Equivariant Object Representations for Manipulation," in *2022 International Conference on Robotics and Automation (ICRA)*, Philadelphia, PA, USA: IEEE, May 2022, pp. 6394–6400. DOI: 10.1109/ICRA46639.2022.9812146.

[17] Y. Hou et al., *Adaptive Compliance Policy: Learning Approximate Compliance for Diffusion Guided Control*, Mar. 2025. DOI: 10.48550/arXiv.2410.09309.

[18] H. Xue et al., "Reactive Diffusion Policy:,"

[19] W. Gao and R. Tedrake, "kPAM 2.0: Feedback Control for Category-Level Robotic Manipulation," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2962–2969, Apr. 2021. DOI: 10.1109/LRA.2021.3062315.

[20] K. Crane, U. Pinkall, and P. Schröder, "Robust fairing via conformal curvature flow," *ACM Transactions on Graphics*, vol. 32, no. 4, pp. 1–10, Jul. 2013. DOI: 10.1145/2461912.2461986.

[21] R. Sawhney and K. Crane, "Monte Carlo Geometry Processing:A Grid-Free Approach to PDE-Based Methods on Volumetric Domains," vol. 38, no. 4,

[22] F. L. Markley, Y. Cheng, J. L. Crassidis, and Y. Oshman, "Averaging Quaternions," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 4, pp. 1193–1197, Jul. 2007. DOI: 10.2514/1.28949.

[23] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The YCB object and Model set: Towards common benchmarks for manipulation research," in *2015 International Conference on Advanced Robotics (ICAR)*, Jul. 2015, pp. 510–517. DOI: 10.1109/ICAR.2015.7251504.

[24] K. Shaw, A. Agarwal, and D. Pathak, *LEAP Hand: Low-Cost, Efficient, and Anthropomorphic Hand for Robot Learning*, Sep. 2023. DOI: 10.48550/arXiv.2309.06440.

[25] N. Sharp, S. Attaiki, K. Crane, and M. Ovsjanikov, "DiffusionNet: Discretization Agnostic Learning on Surfaces," *ACM Transactions on Graphics*, vol. 41, no. 3, 27:1–27:16, Mar. 2022. DOI: 10.1145/3507905.

[26] B. Miller, R. Sawhney, K. Crane, and I. Gkioulekas, *Differential Walk on Spheres*, May 2024.