How to Peel with a Knife: Aligning Fine-Grained Manipulation with Human Preference

Toru Lin*, Shuying Deng*, Zhao-Heng Yin, Pieter Abbeel, Jitendra Malik

https://toruowo.github.io/peel

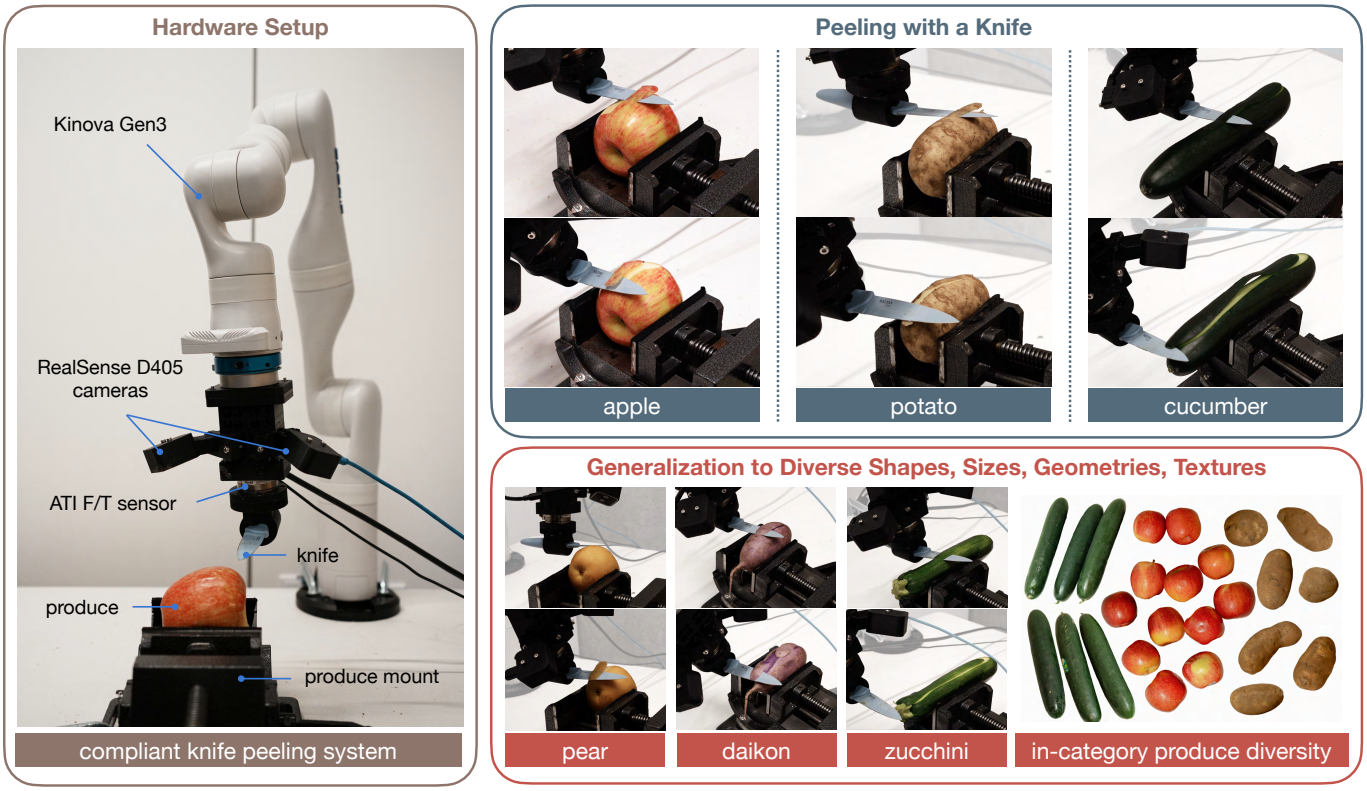

Fig. 1. **An overview of our system setup and learned peeling policies.** We use a 7-DoF Kinova Gen3 arm with impedance control. A custom designed mount holding a knife is attached to the tool end. Two wrist cameras are attached to the tool end and pointing towards the knife and produce. We collect data on three types of produce, train peeling policies that zero-shot generalize to six types of produce with a wide range of geometries and surface physical properties, and finetune the policies to align with human preference of peel quality.

***Abstract*—Many essential manipulation tasks – such as food preparation, surgery, and craftsmanship – remain intractable for autonomous robots. These tasks are characterized not only by contact-rich, force-sensitive dynamics, but also by their "implicit" success criteria: unlike pick-and-place, task quality in these domains is continuous and subjective (e.g. how well a potato is peeled), making quantitative evaluation and reward engineering difficult. We present a learning framework for such tasks, using peeling with a knife as a representative example. Our approach follows a two-stage pipeline: first, we learn a robust initial policy via force-aware data collection and imitation learning, enabling generalization across object variations; second, we refine the policy through preference-based finetuning using a learned reward model that combines quantitative task metrics with qualitative human feedback, aligning policy behavior with human notions of task quality. Using only 50–200 peeling trajectories, our system achieves over 90% average success rates on challenging produce including cucumbers, apples, and potatoes, with performance improving by up to 40% through preference-based finetuning. Remarkably, policies trained on a single produce category exhibit strong zero-shot generalization to unseen in-category instances and to out-of-distribution produce from different categories while maintaining over 90% success rates.**

* Equal contribution.
All authors are affiliated with University of California, Berkeley. Work done while SD was a visiting student researcher from Tsinghua University. Correspondence to toru@berkeley.edu.

I. Introduction

Many essential manipulation tasks – such as food preparation, surgery, and craftsmanship – remain challenging for autonomous robots despite recent progress in learning-based robotic manipulation [1–5]. The fundamental bottlenecks lie in two aspects: (1) *quantity* – the contact-rich and force-sensitive nature of these tasks makes it difficult to collect high-quality demonstration data at scale; (2) *quality* – task success is often continuous, subjective, and difficult to specify mathematically,

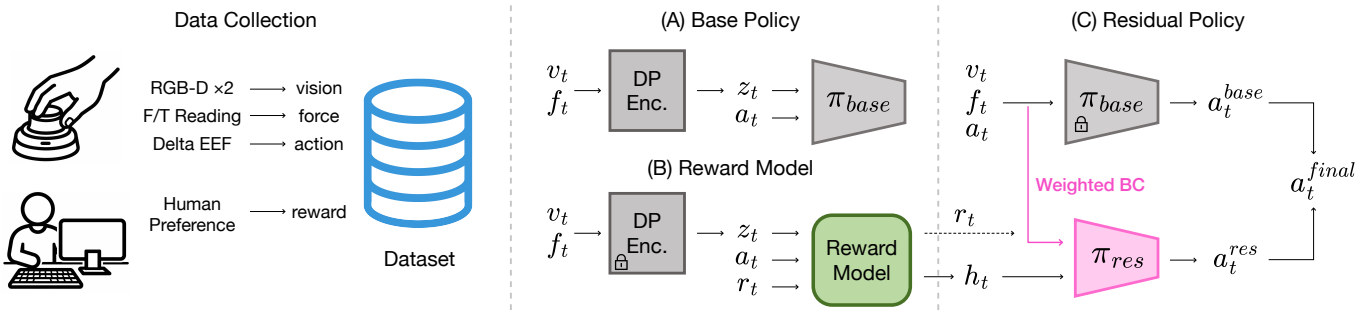

Fig. 2. **A overview of our two-stage learning framework.** This includes details on data and model architecture for compliant data collection, force-aware imitation learning, and preference-based finetuning from a learned reward model.

making it hard to evaluate learning outcomes and optimize for meaningful objectives.

In this work, we study how to address these bottlenecks through the lens of *peeling with a knife*, a representative task in this problem class. Peeling requires precise force regulation under unstable blade–surface contact, accurate real-time tracking of complex geometries, and generalization across variances of natural produce. Furthermore, success in peeling is inherently difficult to quantify: beyond removing the skin, performance is also judged by the cleanliness, evenness, and efficiency of the cut. Prior works address these challenges only partially: model-based controllers are brittle to modeling errors, calibration drift, and object variation; learning-based methods often require large-scale data collection that is expensive or simply infeasible; and evaluation is typically reduced to fixed quantitative metrics that poorly align with human judgments of quality, limiting real-world applicability.

We present a learning framework that addresses both the *quantity* and *quality* bottlenecks by combining efficient data collection, generalizable policy learning, and robust alignment with human preference. Our approach follows a two-stage pipeline. First, we initialize a peeling policy using force-aware imitation learning, providing a strong baseline that generalizes across object variations. Second, we learn a reward model from human feedback and use it to finetune the policy, aligning its behavior with human notions of task quality. We evaluate our method on real-world knife-based peeling across multiple produce categories, including cucumbers, apples, and potatoes. Using only 50–200 trajectories, our approach achieves over 90% average success rates, with performance improving by up to 40% after preference-based finetuning. Remarkably, policies trained on a single produce category generalize zero-shot to unseen instances and to out-of-distribution produce with distinct physical properties.

Our primary contributions are as follows:

- **A two-stage learning framework:** We propose a pipeline that combines compliant data collection, force-aware imitation learning, and preference-based finetuning, and demonstrate its effectiveness toward learning fine-grained manipulation that aligns with human preference.
- **Preference-based reward model:** We show how human preference can be defined in terms of qualitative and quantitative rewards, how a reward model can be learned from the preference labels, and how such a learned reward model can be used to finetune policies to drive substantial policy improvements on real robots.
- **Data-efficient generalization:** We outline a scalable data collection and training pipeline that enables challenging peeling policies from a small amount of real-world data (using as few as 8 fruits). The pipeline integrates visual, proprioceptive, and force sensing into a compact representation for *zero-shot generalizable* policy learning.

Our results demonstrate that robots can acquire highly precise, adaptive, and generalizable contact-rich manipulation skills – such as knife-based peeling – from limited real-world experience when learning is guided by a richer notion of task quality. The proposed framework offers a practical path toward general-purpose manipulation systems capable of mastering a broad class of fine-grained, force-sensitive real-world tasks.

II. Related Work

Learning manipulation from human preference. Modeling and learning from human preference is a long-standing problem in machine learning, with existing work spanning reinforcement learning [6, 7] and supervised learning [8]. Recently, this problem has gained renewed attention through applications to large language models [9], and a growing body of work in robot learning [10–14]. However, existing robotics work largely focuses on simple tasks in simulation or highly constrained real-world settings – e.g. low-dimensional control, short-horizon manipulation, and binary success criteria – where preference modeling and reward learning are relatively straightforward [6, 10, 12, 13]. As a result, these methods do not fully confront the challenges of aligning practical contact-rich manipulation tasks with human preference, where task quality is continuous, subjective, and tightly coupled to subtle force-motion interactions. To our knowledge, our work is the first to investigate learning from human preference on such a challenging manipulation task on real robots.

Peeling with a knife. We are not aware of any prior work that successfully peels multiple types of produce with a knife.

We therefore review adjacent tasks, including knife cutting and peeler-based peeling. Cutting with a knife is substantially more challenging than peeling with a peeler, as it requires precise regulation of force, blade angle, and depth along a continuously evolving contact surface to avoid slippage or breakage. Existing knife-cutting works [15–21] rely primarily on classical model-based approaches, differing mainly in heuristics and analytical models of force dynamics, perception, and knife motion. While these methods demonstrate feasibility, they exhibit limited generalization due to sensitivity to modeling errors and perception noise. Learning-based approaches remain scarce: [22] combines model-based control with a specialized differentiable cutting simulator. Peeler-based peeling has been demonstrated using model-based planning [23, 24], teleoperation data [25], or scripted policies [26, 27], but only on simple geometries with minimal curvature and limited generalization. These limitations highlight the difficulty of collecting high-quality data and learning compliant policies for realistic knife-based peeling.

Force-Based Manipulation. Knife-based peeling is inherently force-sensitive, motivating learning-based approaches for generalization. Prior work incorporates force in the observation space, the action space, or both [25, 28–31]. To use force in observation space, existing works propose architectures to process signals from tactile or force-torque sensors, and adding the encoded feature into observation vector. To use force in action space, a force-based controller such as impedance controller or admittance controller is often used to achieve compliant control. To tackle the challenge of data collection and learning compliant control, ACP [31] and DexForce [30] propose to collect data with kinesthetic teaching and recover the compliance parameters by estimating effective mass and inertia; but these estimations are largely hand-tuned and rule-based, limiting applicability to hard-to-model tasks like peeling. Other works collect data using specialized force-aware teleoperation systems [28, 29], which restrict accessibility. Simulation-based approaches for cutting [22, 32] and sim-to-real transfer [33, 34] have shown promise, but remain difficult to scale beyond simple insertion or cutting tasks due to the complexity of modeling cutting dynamics in deformable, heterogeneous produce [35, 36].

III. How to Peel with a Knife

We present a two-stage framework to learn highly challenging fine-grained manipulation tasks, exemplified by knife-based peeling. This section outlines the three main components of our final system: (1) system design; (2) efficient data collection and policy training to learn a generalizable policy that achieves at least 60% success rates; and (3) preference-based finetuning that uses a learned human preference reward model to improve the policy. The result is a system capable of peeling produce of various physical properties with a knife, achieving 100% success rates on seen produce and over 70% average success rates on unseen produce. Our system and framework are visualized in Figure 1 and 2.

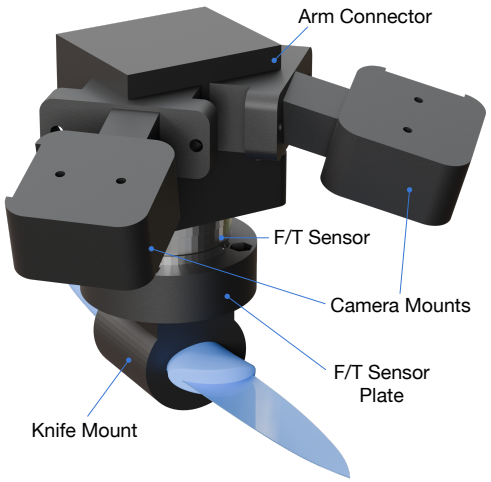

Fig. 3. **End-effector mount.** A CAD visualization of our custom end-effector mount design, including an arm connector, a force-torque sensor plate, two camera mounts, and a knife mount.

A. System Design

Hardware setup. We use a Kinova Gen3 arm which has seven degrees of freedom (DoF). The arm can be torque-controlled which allows for implementation of an impedance controller. We mount an ATI mini45 force-torque sensor between the tool flange and end-effector. The sensor readings are streamed at 500Hz. To stably hold a knife, we custom design an end-effector knife mount as shown in Figure 3. We attach two RealSense D405 wrist cameras near the tool.

Compliant Control. We implement an impedance controller to control the Kinova Gen3 arm. We run the low-level impedance control at 500Hz and send the Python control commands at 10Hz, using an Intel NUC to ensure real-time property. Our implementation is heavily based on open-source projects [37, 38]. The detailed controller parameters to achieve stable compliant control are listed in Appendix A.

B. Data Collection and Policy Training

Infrastructure. We collect high-quality peeling data using human teleoperation. Specifically, we use a 3Dconnexion SpaceMouse to control the 6 DoF end-effector pose of the Kinova arm. To produce smoother end-effector motion in Cartesian space, we implement a weighted least-squares inverse kinematics (IK) solver that dynamically adjusts how much each joint moves. Instead of treating all joints equally, it assigns adaptive weights computed from the Jacobian to penalize joints that cause large leverage, favoring distal joints (elbow/wrist) over proximal ones (base/shoulder). Our solver also uses a weighted null-space projector to stay close to the default pose without reintroducing jitter or violating the smoothness constraint. We collect trajectory data at 10Hz during teleoperation. The collected data include robot joint angles, force-torque sensor readings, and RGBD images from the two wrist cameras.

Data processing. We post-process the data in real time to obtain observations for policy training. Specifically, we standardize the force-torque readings by subtracting the mean of

first 10 samples from all future readings, and segment the RGBD images on knife and object masks separately. The segmentation masks are obtained from running SAM2 [39] online. We record proprioception as delta end-effector pose in the end-effector frame, as previous work [40] has demonstrated that using a relative end-effector trajectory as proprioception enables generalization to arbitrary base position.

Training and inference. With the collected dataset, we learn policies that take vision and force as input and predict proprioception. For visual inputs, we convert colored images to grayscale, multiply clipped depth values with binary segmented masks (both knife mask and object mask), concatenate the processed RGB and depth features, and apply random crop augmentation. For force inputs, we normalize the readings to [-1, 1]. We encode the visual features with ResNet [41] and the force features with MLP. We train the policy using Diffusion Policies [42] with a simple MLP-based denoiser network. During inference, we send actions at 10Hz.

C. Policy Finetuning with Preference-based Reward

Reward design. The quality of a peel is difficult to capture with a single objective metric, and different observers often apply different criteria when judging performance. Human evaluations may consider multiple factors – e.g. peel thickness, continuity and smoothness, and the presence of defects – each weighted differently across individuals. Among these, peel thickness admits a clear geometric interpretation and provides a relatively objective signal, whereas properties such as visual uniformity and overall continuity are inherently holistic and rely strongly on human perceptual judgment. To capture both aspects, we construct a hybrid reward that combines quantitative and qualitative components. The quantitative component measures the relative thickness of the local peel. For this, we temporally segment each trajectory at 2Hz and annotate each segment with one of the six thickness categories shown in Figure 5. The qualitative component captures subjective human preferences based on the overall visual appearance of the peel. These preferences are difficult to express using local or low-level metrics and are typically global in nature; accordingly, we assign a trajectory-level preference score using a Likert-type ordinal scale, as illustrated in Figure 4. We combine the two components using a weighted sum to produce a per-step reward signal that reflects both local geometric precision and global perceptual quality. Implementation details and weighting choices are provided in Appendix B.

Reward-guided policy finetuning. We finetune the peeling policy by freezing the base diffusion policy π_{base} and learning a residual policy that predicts action corrections guided by human preference. We now describe how preference-based rewards are used to supervise the residual policy. To enable preference-aware refinement, we first introduce a learned reward model trained offline prior to policy finetuning. The reward model $r_\psi(z_t, a_t)$ predicts a human preference score for a state–action pair, where a_t denotes the action recorded in the offline dataset, and z_t denotes the encoded latent feature produced by the frozen base policy observation encoder. It is

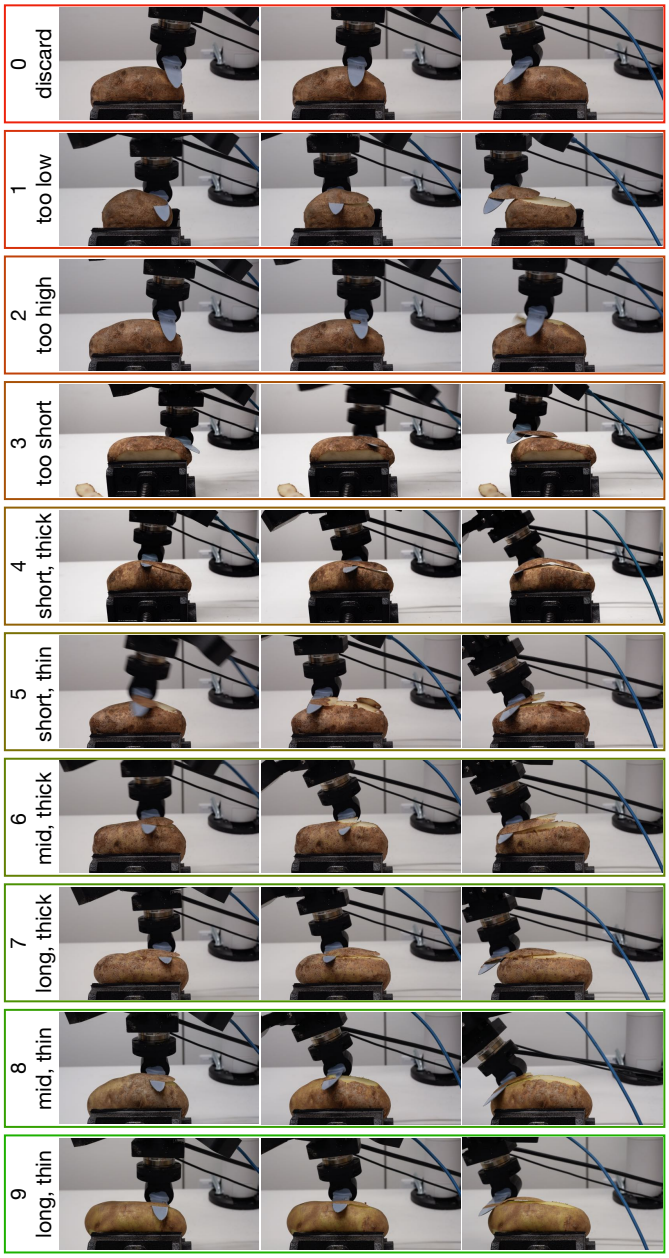

Fig. 4. **Front-view visualization of qualitative score metric.** We use integer scores from 0 to 9 (the higher the better) to capture subjective human preferences based on the overall visual appearance of the peel.

implemented as a three-layer MLP and trained using a mean squared error objective to regress the normalized reward r_t derived from raw human annotations:

$$\mathcal{L}_{\text{reward}} = \mathbb{E}_{(z_t, a_t)} \left[\| r_\psi(z_t, a_t) - r_t \|^2 \right]. \quad (1)$$

In addition to the scalar reward prediction, the reward model exposes an intermediate hidden representation $h_t \in \mathbb{R}^d$, which captures structured aspects of human preference and is later used to condition the residual policy.

The residual policy π_{res}, implemented as a two-layer MLP, predicts an action correction conditioned on the base policy's latent feature z_t, the base action a_t^{base}, and the reward model's

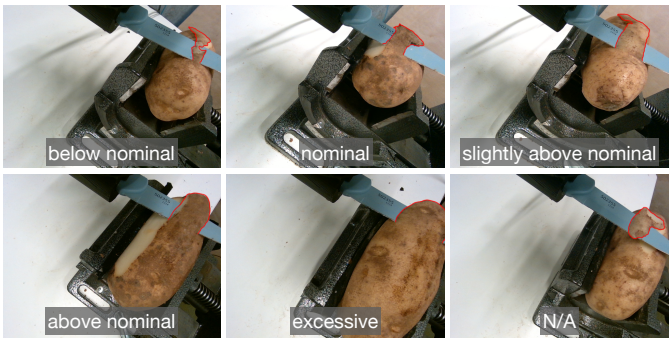

Fig. 5. **Wrist-view visualization of quantitative score metric.** We use six discrete thickness categories, where *nominal* denotes the most desired thickness. Details of how these categories are mapped to normalized scalar rewards can be found in Appendix B.

hidden representation h_t: $a_t^{\text{res}} = \pi_{\text{res}}(z_t, a_t^{\text{base}}, h_t)$. The final action executed by the robot is obtained by adding the residual correction to the base action: $a_t^{\text{final}} = a_t^{\text{base}} + a_t^{\text{res}}$.

We train the residual policy using a reward-weighted behavioral cloning objective that encourages the predicted residual to match the difference between the dataset action and the base action:

$$\mathcal{L}_{\text{res}} = \mathbb{E}_t\Big[w_t \left\|a_t^{\text{res}} - (a_t - a_t^{\text{base}})\right\|^2\Big] + \alpha\, \mathbb{E}_t\big[\|a_t^{\text{res}}\|^2\big]\,. \quad (2)$$

The first term prioritizes imitating high-quality corrections, while the second term regularizes the magnitude of the residual action to prevent overcorrection. The per-step weight is defined as $w_t = \exp(\beta r_t)/\mathbb{E}_t[\exp(\beta r_t)]$ which emphasizes samples with higher predicted preference.

An overview of the full pipeline is shown in Figure 2.

IV. Experiments

We evaluate the proposed framework by demonstrating the learned peeling behaviors on real-world produce and by conducting extensive ablation studies on key system components.

Task definition. The peeling task requires removing a thin, continuous layer of outer skin from a produce item using a handheld knife, while aligning with human preferences over peel thickness, continuity, smoothness, and efficiency. This task is challenging due to the subtle and highly variable boundary between skin and edible flesh, which demands precise force modulation and stable tool–object contact throughout the motion. A high-quality peel removes a consistent layer of skin with minimal thickness, avoids cutting into the underlying flesh, and maintains smooth, energy-efficient knife motion without jitter or discontinuities. To standardize evaluation across irregularly shaped produce, we define a peel segment as a single stroke executed along the principal axis of the object (i.e. the longest line passing through its centroid). Each peeling trial consists of one or more such strokes, executed sequentially around the circumference until a full side of the surface is covered.

Evaluation metrics. We evaluate peeling performance using both qualitative and quantitative metrics derived from human preference and perception, as illustrated in Figures 4 and 5. The qualitative metric is a holistic preference score reflecting human judgments of overall peel thickness, length, and continuity. The quantitative metric measures peel thickness as perceived from the robot's onboard sensory data. We consider a peel with a qualitative score greater than 3 to be successful.

Training details. We collect 50, 150, and 200 demonstrations for cucumber, apple, and potato, respectively, with each demonstration consisting of a single peel stroke. All RGB-D observations are resized to $(120, 160, 4)$ in height, width, and channels. Force–torque measurements from the ATI sensor are represented as a 6-dimensional vector, and proprioceptive input consists of a 7-dimensional joint-angle vector. The visual encoder is a ResNet-18 with a 64-dimensional output. The state encoder is a two-layer feedforward network with hidden dimension 64 and output dimension 64. We apply dropout with rate 0.1 to mitigate overfitting. For the diffusion policy, we use a two-layer feedforward denoising network with hidden dimension 64, which predicts 6-dimensional end-effector actions.

A. Overall Performance

Task success rates and generalization. We evaluate peeling on three produce types: cucumbers, apples, and potatoes. For generalization, we perform two kinds of tests: (1) on same produce type as training data, test generalization of the peeling behavior on different start poses and diverse produce instances with small variations in size, shape, stiffness, and surface texture; (2) on unseen produce types, test generalization to completely out-of-distribution produce instances. For all tests with same produce type, we report a success rate of 100% across cucumber, apple, and potato. Our cucumber policy achieves 50% success rate on zucchini, apple policy 90% on pear, and potato policy 80% on daikon radish. Videos can be found in our supplementary.

B. How to Collect High-Quality Data for Peeling?

Comparison of data collection methods. We compare our SpaceMouse-based teleoperation method with model-based planner, VR-based teleoperation [43], and kinesthetic teaching followed by replay with heuristic-based compliance parameters [31]. For each data collection method, we evaluate the quality of 10 trajectories collected using qualitative metrics defined in Section III-C. We show the success rate, average performance and time taken for each method in Table I. Below, we share implementation details and discuss our experience with each method.

- For heuristic planner, we use a calibrated third-view L515 LiDAR camera to capture RGBD images of the peeling scene, and implement a planner based on visual inputs. Specifically, we first extract a segmented point cloud of the object, plan a dense trajectory on its surface with 20 waypoints, extract surface normal of the object at each waypoint, calculate the desired knife pose trajectory, and solve for the desired joint trajectory using IK. A visualization of the planning procedure is shown in

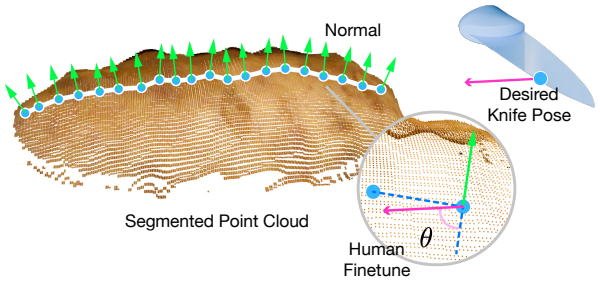

Fig. 6. **Model-based planner.** A visualization of the planning procedure of our heuristic planner. We use a calibrated top-view LiDAR camera to capture the scene, extract a segmented point cloud of the object, and plan a dense trajectory on its surface that represents the peeling path. We then calculate the desired knife pose trajectory based on the surface normal of each waypoint on the path, and execute the planned trajectory with human-in-the-loop correction of knife pose via keyboard control (e.g. +20 degrees in yaw).

Figure 6. It is capable of peeling with human in the loop supervision, where humans can finetune in real time the next desired knife pose. However, it cannot execute fully autonomously due to the large variation of object geometries and surface properties.

- Our VR teleoperation is implemented based on [43]. We find that it performs much worse than SpaceMouse teleoperation on this peeling task. Reasonably, the instability of the physical action of holding the VR controllers and the noise in VR tracking performance both prevent teleoperator from reaching the precision and delicacy required for peeling.
- We evaluate kinesthetic teaching in two stages: first, collection of position trajectories; second, trajectory replay with hand-tuned compliance parameters using heuristics similar to ACP [31]. This is for fairer comparison with other methods, since trajectories collected with kinesthetic teaching have different effective compliance parameters from the original ones used by controller and cannot be simply recovered. From the results, we conclude three interesting findings: (1) it is substantially faster to collect good peeling data with kinesthetic teaching; (2) however, the raw data quality is lower than that with SpaceMouse – likely because kinesthetic teaching requires more challenging (strenuous yet precise) muscle control; (3) the high quality of raw data does not transfer to replay performance, even after extensive parameter tuning – likely due to the challenging variations of produce surfaces.

Surprisingly, we find SpaceMouse teleoperation – without additional modifications like adaptive compliance or haptic feedback – an efficient way to collect high-quality data for the challenging task of peeling with a knife.

C. How to Learn High-Performance Peeling Policies?

Designing observation and action spaces. We compare important design choices in camera placement, number of cameras, and choice of data modalities that enables precise, adaptive and generalizable peeling. For each design choice, we perform ablation experiments by comparing the task success rate and generalization. Results for each policy are obtained from running 5 evaluation trials with the best performing checkpoint on potatoes.

Tab. I. **Quality and efficiency of data collection across different methods.** For each experiment, qualitative scores of 10 trajectories collected from cucumbers are averaged. A trajectory with score above 3 is counted as a success. Average time taken is only calculated from successful trajectories. We treat Kinesthetic Teaching as a special category because the collected data has ambiguous compliant parameters. We therefore show results from both the initial data collection and the best trajectory replay where compliance parameters are tuned using heuristics similar to [31].

Data Collection	Success %	Avg. Score	Avg. Time (s)
Heuristic Planner	20	1.8	50
VR	20	2.1	69.5
SpaceMouse **(ours)**	100	8.5	46
Kinesthetic Teaching	100	7.2	13.8
Replay	0	1.6	N.A.

- Wrist camera: As shown in Figure 3, we attach two wrist cameras to the end effector mount. Subtly, since our peeling direction is fixed, one camera always capture the object and knife view slightly *before* the current peeling action, while the other camera always capture the object and knife view slightly *after*. In Table II, we present an ablation study comparing using both cameras, only the *before* camera, and only the *after* camera. Policy performance where both cameras are used is strictly better than when only a single camera is used; this is reasonable since because two cameras include strictly more information, including implicit 3D information. Furthermore, we find that the *before* camera contributes to policy performance more than the *after* camera. We hypothesize this is because the *before* camera captures a less occluded view of the contact between knife edge and produce than the *after* camera, due to the produce's curved geometry.
- Data modalities: Our policy takes three types of sensing modalities as inputs: proprioception, vision, and touch. For vision, we use RGBD images where the original colored RGB is converted to grayscale. In Table III, we quantitatively investigate how the input modalities affect policy performance. Results show that it is important to (1) use both visual and force-torque observations, and (2) convert RGB to grayscale. We reason that (1) is because of the force-sensitive and position-sensitive nature of our peeling task, and (2) because grayscale image input forces the policy to focus on geometric features rather than produce texture, greatly aiding generalization.

Sample efficiency. We study the efficiency of learning by empirically evaluating the correlation between number of demonstrations and policy performance on potato. The results are shown in Table IV. For potato peeling, 200 trajectories are needed to reach 80% success rate, with an average score of 5.9. This amounts to about 33 potatoes, where each potato contributes about 6 trajectories. Similarly, with 50 trajectories

Tab. II. **Wrist camera.** We study the relative importance of two wrist cameras on potato data. Interestingly, the *before* camera contributes to policy performance more than the *after* camera, suggesting the benefit of learning from a less occluded view.

Num Cam	Success %	Avg. Score
Both	80	5.9
Before Only	100	4.8
After Only	60	3.2

Tab. III. **Data modalities.** We denote grayscale RGB images as gGRB, original RGB images as RGB, depth images as D, and force-torque readings as F. Visual observation with grayscaled RGB and force-torque observation are both important.

Modality	Success %	Avg. Score
gRGB, D, F	80	5.9
gRGB, D	60	5.2
RGB, D, F	0	1.6
F	0	0.6

Tab. IV. **Sample efficiency.** We study the correlation between number of trajectories and policy performance on potato data. We find that performance improves almost linearly in terms of both success rate and qualitative score as number of trajectories increase until success rate reaches 80%.

Num Traj	Success %	Avg. Score
200 (100%)	80	5.9
100 (50%)	60	4.4
40 (20%)	10	2.4

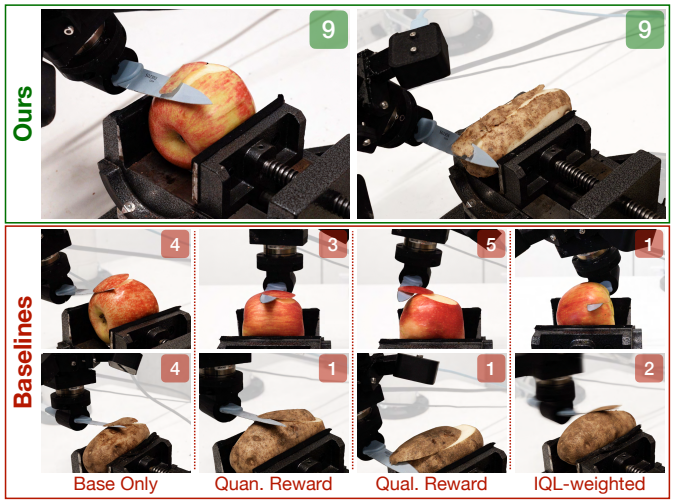

Fig. 7. **Qualitative comparison with baselines.** We show qualitative comparison between our policy and the baselines. The peel quality is substantially higher even when both are counted as success, as indicated by the qualitative score (top right corner of each image).

(about 8 cucumbers) our cucumber peeling policy achieves 100% success rate and an average score of 8.5; with 100 trajectories (about 17 apples) our apple peeling policy achieves 60% success rate and an average score of 3.8. For each reported number, the evaluation is conducted over 10 trials.

D. How to Align Learned Policies with Human Preference?

Comparison with baselines. Our final policy is finetuned with both quantitative and qualitative rewards. We implement the following baselines for comparison. To ensure fairness of comparison, we train all policies with the same dataset.

- Base Only: base policy without finetuning.
- Quantitative Reward Only: base policy finetuned with only quantitative preference reward.
- Qualitative Reward Only: base policy finetuned with only qualitative preference reward.
- IQL-weighted: we first train critic networks following IQL [44] to estimate state-action values Q and V, and compute per-step advantages as $A = Q - V$. We then perform advantage-weighted supervised finetuning of the base policy, where each behavior cloning loss is weighted by a monotonic exponential function of the estimated advantage, emphasizing actions that are relatively better than the policy's expected behavior at the same state.

We show quantitative results in Table V and qualitative comparison in Figure 7. We find that supervised finetuning not only exhibits better training stability and lower infrastructure hurdle, but also leads to better performance than offline RL–style critic-guided finetuning method with implicit Q-learning [44].

Designing human preference scores. We explore two main design choices for representation of human preference: density across time horizon (per-step vs per-60-steps vs per-trajectory) and density in value (binary vs fine-grained). While denser rewards might benefit learning, they come at a higher labor cost. Empirically, we find the right balance lies at per-step, fine-grained reward values – where the labor cost can be offset by learning a reward function from a small number of hand-annotated data (see details in Section III-C). In Table VI, we show the quantitative comparison of finetuned policy performance from different reward designs.

Utilizing learned reward model. Given a reward model that can assign per-step score to a trajectory, a key question is how to improve a policy with it. In addition to our final approach, we consider two alternatives baselines and run experiments to compare these approaches: (A) One-Step Advantage: instead of using the raw reward, we calculate a trajectory-centered "advantage" by first computing a trajectory baseline b_τ, then forming a per-step advantage $A_t = r_t - b_\tau$, enabling more localized credit assignment when weighting behavior cloning updates. (B) Binary Filtering: instead of applying smooth per-step weighting via an exponential preference-based weight, we hard-select high-scoring steps using a binary filter, retaining only a fixed top fraction according to the preference reward, and finetune uniformly on the remaining samples without any additional weighting. Empirical results are shown in Table VII, demonstrating that our approach is the most effective.

Choosing the right training scheme. Finally, we investigate two important design choices regarding training: (1) whether to use a residual network, and (2) whether to finetune on a frozen base policy or train from scratch. Our final policy is obtained by finetuning on a residual network. We compare the performance with two baselines: (1) No Residual: directly finetuning the base policy; (2) From Scratch: learn the policy from scratch with reward weighting. Empirical results are shwon in Table VIII. We find that to achieve stable and robust policy, both separating policy learning into two stages (base policy training and finetuning) and using a residual layer are

Tab. V. **Comparison with baselines for finetuning.** Task success rates (%) and average scores of four reward alignment methods from experiments on apples (A) and potatoes (P).

Method	Success % (A)	Score (A)	Success % (P)	Score (P)
Ours	100	7.1	100	7.3
Base Only	60	3.8	80	5.9
Quan. R.	40	3.8	60	4.0
Qual. R.	0	1.4	60	4.4
IQL-w	0	0.2	0	2.6

Tab. VI. **Ablation studies on reward design.** We study how reward density across time horizon – per-step (PS) vs per-60-steps (P60) vs per-trajectory (PT) – and in value – binary (B) vs fine-grained (FG) – affects effectiveness of finetuning. Empirically, we find the right balance lies at per-step, fine-grained reward values (ours).

Reward	Success % (A)	Score (A)	Success % (P)	Score (P)
Ours	100	7.1	100	7.3
PS+B	0	0.0	0	0.0
P60+FG	0	1.0	20	3.8
PT+FG	40	2.3	0	1.2

Tab. VII. **How to finetune policies with learned reward.** We compare our finetuning method to two other ways to finetune policies given per-step reward: using one-step advantage instead of raw reward (OneStep), and applying reward only on high-scoring steps filtered with a hard threshold (Filter). Neither outperforms our method.

Method	Success % (A)	Score (A)	Success % (P)	Score (P)
Ours	100	7.1	100	7.3
One-Step	0	1.2	60	4.6
Filter	0	1.0	80	5.2

Tab. VIII. **Comparison with different training schemes.** Our final policy is first learned from scratch without reward and then finetune with reward on a residual network. We compare with two alternatives: training with reward from scratch (Scratch), and finetuning without residual network (No Res.). We find that our training scheme is the only that ensures stable learning.

Training	Success % (A)	Score (A)	Success % (P)	Score (P)
Ours	100	7.1	100	7.3
Scratch	0	0.4	0	0.0
No Res.	0	1.2	40	3.0

of key importance.

E. Failure Cases

We systematically collect and study the failure cases. In addition to cutting too low and cutting too high (qualitative score 1 and 2), most failures happen during generalization experiments can be reasonably hypothesized as due to model's inability to generalize. For example, in our supplementary video, we show failures when deploying cucumber policy to apple, apple to potato, and potato to cucumber. While it is unsurprising that a policy trained on one produce cannot generalize to another produce with completely different characteristics, how far this generalization goes depends on a wide range of factors and will make for an interesting topic of study.

V. Conclusion

In this work, we propose a systematic approach to learn end-to-end policies capable of peeling a diverse range of real-world produce with a knife – one of the most challenging manipulation tasks. Our learned policies showcase not only extreme precision, but also zero-shot generalization to completely unseen objects. Our pipeline consists of efficient data collection, robust policy learning, and preference-based reward refinement. Our key idea is to first initialize generalizable peeling skills by learning from force-aware demonstration data, then align the precision and naturalness of policies through learned human preferences, without requiring further collection of expert demonstrations.

VI. Limitations and Future Work

While our framework demonstrates strong performance on an exceptionally challenging task, a key limitation is its reliance on manually collected, high-quality demonstrations. Improving scalability is therefore an important direction for future work. One promising extension is to incorporate online reinforcement learning as a finetuning stage, which recent work has shown to be effective for refining real-world manipulation policies [45–47]. Another direction is to reduce reliance on fully human teleoperation by adopting mixed-autonomy data collection, for example by combining model-based planners with intermittent human intervention, thereby lowering human effort while maintaining data quality.

Beyond scalability, several modifications could further improve overall system performance. First, inspired by advances in large language model alignment, more expressive reward parameterizations – such as ranking-based or listwise rewards – may enable stronger alignment with human notions of task quality. Second, perception remains a bottleneck: augmenting the sensing setup with additional viewpoints (e.g. a front-facing camera) could improve estimation of peel thickness and surface quality. While orthogonal to this work, continued progress in depth sensing hardware and segmentation models would further benefit such systems.

More broadly, by introducing both qualitative and quantitative evaluation metrics for knife-based peeling and demonstrating how to learn an effective reward model from them, this work opens the door to systematic studies of the trade-off between data quality and data quantity in preference-based robot learning.

Finally, a practical limitation of real-world food manipulation research is the generation of food waste. We hope future work will explore reusable "surrogate produce" or improved simulation and sim-to-real transfer methods that enable similar experimentation with reduced environmental cost.

Acknowledgment

We thank the authors of the open-source repositories that informed our implementation of the compliant controller on the Kinova Gen3 [37, 38, 48] for their technical guidance. We are grateful to Yifan Hou for initial advice on compliant controller implementation and for sharing resources related to mount design. We also thank Pingchuan Ma for assistance with photography and videography.

References

[1] H.-S. Fang, C. Wang, H. Fang, M. Gou, J. Liu, H. Yan, W. Liu, Y. Xie, and C. Lu, "Anygrasp: Robust and efficient grasp perception in spatial and temporal domains," *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3929–3945, 2023. 1

[2] T. Lin, K. Sachdev, L. Fan, J. Malik, and Y. Zhu, "Sim-to-real reinforcement learning for vision-based dexterous manipulation on humanoids," *arXiv preprint arXiv:2502.20396*, 2025.

[3] R. Wang, J. Zhang, J. Chen, Y. Xu, P. Li, T. Liu, and H. Wang, "Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation," *arXiv preprint arXiv:2210.02697*, 2022.

[4] T. G. W. Lum, M. Matak, V. Makoviychuk, A. Handa, A. Allshire, T. Hermans, N. D. Ratliff, and K. Van Wyk, "Dextrah-g: Pixels-to-action dexterous arm-hand grasping with geometric fabrics," *arXiv preprint arXiv:2407.02274*, 2024.

[5] P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, *et al.*, "$\pi_{0.5}$: a vision-language-action model with open-world generalization," *arXiv preprint arXiv:2504.16054*, 2025. 1

[6] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," *Advances in neural information processing systems*, vol. 30, 2017. 2

[7] C. Wirth, R. Akrour, G. Neumann, and J. Fürnkranz, "A survey of preference-based reinforcement learning methods," *Journal of Machine Learning Research*, vol. 18, no. 136, pp. 1–46, 2017. 2

[8] J. Fürnkranz and E. Hüllermeier, "Pairwise preference learning and ranking," in *European conference on machine learning*. Springer, 2003, pp. 145–156. 2

[9] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, *et al.*, "A survey on evaluation of large language models," *ACM transactions on intelligent systems and technology*, vol. 15, no. 3, pp. 1–45, 2024. 2

[10] B. Ibarz, J. Leike, T. Pohlen, G. Irving, S. Legg, and D. Amodei, "Reward learning from human preferences and demonstrations in atari," *Advances in neural information processing systems*, vol. 31, 2018. 2

[11] J. Hejna, R. Rafailov, H. Sikchi, C. Finn, S. Niekum, W. B. Knox, and D. Sadigh, "Contrastive preference learning: learning from human feedback without rl," *arXiv preprint arXiv:2310.13639*, 2023.

[12] D. J. Hejna III and D. Sadigh, "Few-shot preference learning for human-in-the-loop rl," in *Conference on Robot Learning*. PMLR, 2023, pp. 2014–2025. 2

[13] K. Lee, L. Smith, and P. Abbeel, "Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training," *arXiv preprint arXiv:2106.05091*, 2021. 2

[14] Y. Chen, D. K. Jha, M. Tomizuka, and D. Romeres, "Fdpp: Fine-tune diffusion policy with human preference," *arXiv preprint arXiv:2501.08259*, 2025. 2

[15] P. Long, W. Khalil, and P. Martinet, "Force/vision control for robotic cutting of soft materials," in *2014 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2014, pp. 4716–4721. 3

[16] X. Mu, Y. Xue, and Y.-B. Jia, "Dexterous robotic cutting based on fracture mechanics and force control," *IEEE Transactions on Automation Science and Engineering*, 2023.

[17] ——, "Robotic cutting: Mechanics and control of knife motion," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3066–3072.

[18] P. Jamdagni and Y.-B. Jia, "Robotic cutting of fruits and vegetables: Modeling the effects of deformation, fracture toughness, knife edge geometry, and motion," *IEEE Transactions on Robotics*, 2024.

[19] B. Yang, H. Wang, W. Chen, and Z. Wang, "Vision-based cutting control of deformable objects," in *2016 IEEE International Conference on Real-time Computing and Robotics (RCAR)*. IEEE, 2016, pp. 650–655.

[20] L. Han, H. Wang, Z. Liu, W. Chen, and X. Zhang, "Vision-based cutting control of deformable objects with surface tracking," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 4, pp. 2016–2026, 2020.

[21] A. Straižys, M. Burke, and S. Ramamoorthy, "Surfing on an uncertain edge: Precision cutting of soft tissue using torque-based medium classification," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4623–4629. 3

[22] Z. Xu, Z. Xian, X. Lin, C. Chi, Z. Huang, C. Gan, and S. Song, "Roboninja: Learning an adaptive cutting policy for multi-material objects," in *Proceedings of Robotics: Science and Systems (RSS)*, 2023. 3

[23] Y. Watanabe, K. Nagahama, K. Yamazaki, K. Okada, and M. Inaba, "Cooking behavior with handling general cooking tools based on a system integration for a life-sized humanoid robot," *Paladyn, Journal of Behavioral Robotics*, vol. 4, no. 2, pp. 63–72, 2013. 3

[24] C. Dong, L. Yu, M. Takizawa, S. Kudoh, and T. Suehiro, "Food peeling method for dual-arm cooking robot," in *2021 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, 2021, pp. 801–806. 3

[25] H. Xue, J. Ren, W. Chen, G. Zhang, Y. Fang, G. Gu, H. Xu, and C. Lu, "Reactive diffusion policy: Slow-fast visual-tactile policy learning for contact-rich manipulation," *arXiv preprint arXiv:2503.02881*, 2025. 3

[26] T. Chen, E. Cousineau, N. Kuppuswamy, and P. Agrawal, "Vegetable peeling: A case study in constrained dexterous manipulation," *arXiv preprint arXiv:2407.07884*, 2024. 3

[27] R. Ye, Y. Hu, Y. A. Bian, L. Kulm, and T. Bhattacharjee, "Morpheus: a multimodal one-armed robot-assisted peeling system with human users in-the-loop," in *2024 IEEE International Conference on Robotics and Automation*

(ICRA). IEEE, 2024, pp. 9540–9547. 3

[28] Z. He, H. Fang, J. Chen, H.-S. Fang, and C. Lu, "Foar: Force-aware reactive policy for contact-rich robotic manipulation," *IEEE Robotics and Automation Letters*, 2025. 3

[29] W. Liu, J. Wang, Y. Wang, W. Wang, and C. Lu, "Forcemimic: Force-centric imitation learning with force-motion capture system for contact-rich manipulation," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 1105–1112. 3

[30] C. Chen, Z. Yu, H. Choi, M. Cutkosky, and J. Bohg, "Dexforce: Extracting force-informed actions from kinesthetic demonstrations for dexterous manipulation," *IEEE Robotics and Automation Letters*, 2025. 3

[31] Y. Hou, Z. Liu, C. Chi, E. Cousineau, N. Kuppuswamy, S. Feng, B. Burchfiel, and S. Song, "Adaptive compliance policy: Learning approximate compliance for diffusion guided control," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 4829–4836. 3, 5, 6

[32] E. Heiden, M. Macklin, Y. S. Narang, D. Fox, A. Garg, and F. Ramos, "DiSECt: A Differentiable Simulation Engine for Autonomous Robotic Cutting," in *Proceedings of Robotics: Science and Systems*, Virtual, July 2021. 3

[33] X. Zhang, C. Wang, L. Sun, Z. Wu, X. Zhu, and M. Tomizuka, "Efficient sim-to-real transfer of contact-rich manipulation skills with online admittance residual learning," in *Conference on Robot Learning*. PMLR, 2023, pp. 1621–1639. 3

[34] X. Zhang, M. Tomizuka, and H. Li, "Bridging the sim-to-real gap with dynamic compliance tuning for industrial insertion," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 4356–4363. 3

[35] H. Yin, A. Varava, and D. Kragic, "Modeling, learning, perception, and control methods for deformable object manipulation," *Science Robotics*, vol. 6, no. 54, p. eabd8803, 2021. 3

[36] G. Sochacki, X. Zhang, A. Abdulali, and F. Iida, "Towards practical robotic chef: Review of relevant work and future challenges," *Journal of Field Robotics*, vol. 41, no. 5, pp. 1596–1616, 2024. 3

[37] A. L. Mitchell, T. Flatscher, and I. Posner, "Task and joint space dual-arm compliant control," 2025. [Online]. Available: https://arxiv.org/abs/2504.21159 3, 8

[38] J. Wu, W. Chong, R. Holmberg, A. Prasad, Y. Gao, O. Khatib, S. Song, S. Rusinkiewicz, and J. Bohg, "Tidybot++: An open-source holonomic mobile manipulator for robot learning," in *Conference on Robot Learning*, 2024. 3, 8

[39] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer, "Sam 2: Segment anything in images and videos," *arXiv preprint arXiv:2408.00714*, 2024. [Online]. Available: https://arxiv.org/abs/2408.00714 4

[40] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song, "Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots," *arXiv:2402.10329*, 2024. 4

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016. 4

[42] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," in *RSS*, 2023. 4

[43] T. Lin, Y. Zhang, Q. Li, H. Qi, B. Yi, S. Levine, and J. Malik, "Learning visuotactile skills with two multifingered hands," *arXiv:2404.16823*, 2024. 5, 6

[44] I. Kostrikov, A. Nair, and S. Levine, "Offline reinforcement learning with implicit q-learning," *arXiv preprint arXiv:2110.06169*, 2021. 7

[45] L. Ankile, Z. Jiang, R. Duan, G. Shi, P. Abbeel, and A. Nagabandi, "Residual off-policy rl for fine-tuning behavior cloning policies," *arXiv preprint arXiv:2509.19301*, 2025. 8

[46] W. Xiao, H. Lin, A. Peng, H. Xue, T. He, Y. Xie, F. Hu, J. Wu, Z. Luo, L. Fan, *et al.*, "Self-improving vision-language-action models with data generation via residual rl," *arXiv preprint arXiv:2511.00091*, 2025.

[47] Y. Li, X. Ma, J. Xu, Y. Cui, Z. Cui, Z. Han, L. Huang, T. Kong, Y. Liu, H. Niu, *et al.*, "Gr-rl: Going dexterous and precise for long-horizon robotic manipulation," *arXiv preprint arXiv:2512.01801*, 2025. 8

[48] R. Madan, R. K. Jenamani, and S. W. Han, "gen3_compliant_controllers: Ros package providing compliant controllers for the kinova gen3 arm," https://github.com/empriselab/gen3_compliant_controllers, 2024. 8

AUTHOR CONTRIBUTIONS

TL conceived the initial idea, conceptualized the algorithmic and system components, and led the project. She procured and set up the hardware, developed the controller and data collection infrastructure, collected all demonstration data, conducted all evaluation experiments, and contributed to debugging throughout the project. She wrote the manuscript and finalized the figures, videos, and project website.

SD set up the hardware, designed the custom connector mounts, implemented the heuristic-based planner baseline for data collection, labeled the human preference dataset, implemented preference-based finetuning pipeline and finetuned the base policies, wrote the appendix, and made significant contribution to the figures, videos, and project website.

ZY implemented the diffusion policy training and contributed to technical discussions and debugging throughout the project.

PA and JM provided high-level guidance and funding support for the project.

APPENDIX

A. Compliant Controller Details

We execute policy commands using a torque-level joint-space impedance controller with internal compliance adaptation. The controller runs at $\Delta t = 0.002\,\mathrm{s}$ (500 Hz) and outputs joint torques at each control step. All gain matrices $(K_p, K_d, K_r, K_l, K_{lp})$ are diagonal and positive definite unless otherwise specified.

Given desired joint trajectories $(q_d^t, \dot{q}_d^t)$ generated by an online trajectory generator and sensed joint states $(q_s^t, \dot{q}_s^t, \tau_s^t)$, we first compute a task tracking torque

$$\tau_{\text{task}}^t = -K_p\big(q_n^t - q_d^t\big) - K_d\big(\dot{q}_n^t - \dot{q}_d^t\big) + g(q_s^t), \tag{3}$$

where $q_n^t, \dot{q}_n^t$ denote internal nominal joint states and $g(q_s^t)$ is the gravity compensation torque computed from the sensed joint configuration.

To induce compliance under external contact, the nominal joint acceleration is updated according to the discrepancy between the commanded torque and the measured joint torque:

$$\ddot{q}_n^t = K_r^{-1}\big(\tau_{\text{task}}^t - \tau_s^{f,t}\big), \tag{4}$$

where K_r is a diagonal stiffness matrix and $\tau_s^{f,t}$ is a low-pass filtered version of the sensed torque,

$$\tau_s^{f,t} = \alpha\,\tau_s^t + (1-\alpha)\,\tau_s^{f,t-1}. \tag{5}$$

This formulation allows the nominal trajectory to adapt when sustained torque discrepancies are observed, effectively introducing admittance-like compliance while retaining torque-level control.

The nominal joint velocity and position are updated using semi-implicit Euler integration,

$$\dot{q}_n^{t+1} = \dot{q}_n^t + \ddot{q}_n^t\,\Delta t, \qquad q_n^{t+1} = q_n^t + \dot{q}_n^{t+1}\,\Delta t. \tag{6}$$

To reduce drift between nominal and sensed trajectories during prolonged contact, we apply a friction-like coupling term

$$\tau_f^t = K_r K_l\Big((\dot{q}_n^t - \dot{q}_s^t) + K_{lp}(q_n^t - q_s^t)\Big), \tag{7}$$

which damps relative motion and improves stability by softly pulling the nominal state toward the sensed state.

The final commanded torque is

$$\tau_c^t = \tau_{\text{task}}^t + \tau_f^t. \tag{8}$$

Based on this formulation, the controller parameters used in our experiments are listed in Table IX.

Tab. IX. Impedance controller parameters.

α	0.01
K_r	[0.3, 0.3, 0.3, 0.3, 0.18, 0.18, 0.18]
K_l	[106.2, 100.8, 106.2, 106.2, 131.4, 106.2, 106.2]
K_{lp}	[11.89, 25.52, 22.0, 22.0, 22.0, 22.0, 22.0]
K_p	[382.2, 296.4, 347.1, 400.0, 200.0, 200.0, 200.0]
K_d	[21.0, 17.5, 10.0, 10.0, 5.0, 5.0, 5.0]

B. Reward Design Details

We use a hybrid reward formulation that combines *quantitative* and *qualitative* components. Each demonstration is annotated at two temporal resolutions: a segment-level quantitative score measuring relative peeling thickness, and a trajectory-level qualitative score capturing overall execution preference.

1) Quantitative reward: Quantitative scores are provided at the segment level (2 Hz). We use six discrete thickness categories (Fig. 5): *below nominal*, *nominal*, *slightly above nominal*, *above nominal*, *excessive*, and *N/A*. These categories are mapped to normalized scalar rewards $\mathcal{R}_{\text{quant}} \in [0, 1]$ via task-specific lookup tables for apple and potato (Table X).

Tab. X. Quantitative reward mapping for apple and potato.

Quantitative label	Apple $\mathcal{R}_{\text{quant}}$	Potato $\mathcal{R}_{\text{quant}}$
below nominal	0.3	0.5
nominal	1.0	1.0
slightly above nominal	0.8	0.5
above nominal	0.3	0.1
excessive	0.0	0.0
N/A	0.0	0.0

Quantitative rewards are converted into per-step signals by uniformly assigning the segment reward to all frames within the segment. To reduce boundary discontinuities, we apply a lightweight linear smoothing over the O overlapping frame pairs between adjacent segments. For the i-th overlap ($i = 0, \ldots, O-1$), we interpolate as

$$\alpha_i = \frac{i+1}{O+1}, \qquad r \leftarrow (1-\alpha_i)\,r_{\text{prev}} + \alpha_i\,r_{\text{next}}, \tag{9}$$

and assign the interpolated value symmetrically to both sides of the segment boundary.

2) Qualitative reward: Each trajectory additionally receives a single qualitative preference score in the range $[0, 9]$ (Fig. 4), reflecting overall execution quality such as consistency, smoothness, and perceived preference. The qualitative score is mapped to a normalized scalar reward $\mathcal{R}_{\text{qual}} \in [0, 1]$ using the task-specific lookup table shown in Table XI.

Tab. XI. Qualitative reward mapping for apple and potato.

Qualitative score	Descriptor	Apple $\mathcal{R}_{\text{qual}}$	Potato $\mathcal{R}_{\text{qual}}$
0	discard	0.0	0.0
1	too low	0.1	0.1
2	too high	0.2	0.2
3	too short	0.3	0.3
4	short, thick	0.4	0.4
5	short, thin	0.5	0.5
6	mid, thick	0.6	0.6
7	long, thick	0.8	0.8
8	mid, thin	0.9	0.9
9	long, thin	1.0	1.0

3) Combined reward: At each timestep, the final reward is computed by combining quantitative and qualitative components:

$$r = \begin{cases} \mathcal{R}_{\text{quant}}, & \text{if } \mathcal{R}_{\text{quant}} < \tau, \\ \alpha\, \mathcal{R}_{\text{quant}} + (1 - \alpha)\, \mathcal{R}_{\text{qual}}, & \text{otherwise.} \end{cases} \tag{10}$$

In all experiments, we set $\tau = 0.1$ and use segment length $L = 15$ with overlap $O = 3$. The weighting parameter α is set to 0.85 for the apple task and 0.75 for the potato task.