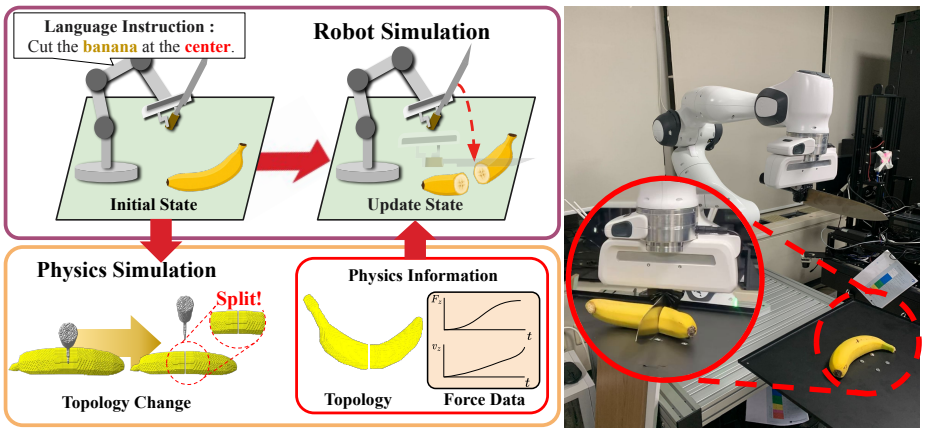
# CulinaryCut: A Physics-Grounded Cutting Benchmark for Vision-Language-Action Models

Hyunseo Koh[*1,3], Chang-Yong Song[*2], Youngjae Choi[*1],
Misa Viveiros[2], David Hyde[2], and Heewon Kim[1,3]

[1]Soongsil University [2]Vanderbilt University [3]Kairoba Inc.




**(a) Physics-grounded Robot Simulation** **(b) Real-world Deployment**

**Fig. 1: Physics-grounded simulation for robotic food cutting.** (a) Our simulator, *IntSim*, integrates physical and topological information from a physics simulator, such as contact forces and topology changes, into robot simulation to generate *CulinaryCut*, a physics-grounded benchmark for cutting. (b) Policies trained with *CulinaryCut* transfer effectively to real-world robotic food cutting, demonstrating the practical applicability of the proposed simulation.

**Abstract.** Food cutting is a representative non-rigid manipulation task involving deformation, contact forces, and material separation, yet it remains largely unexplored in VLA research. Existing robot manipulation datasets are primarily built around rigid objects and struggle to jointly capture the spatial precision, deformation, topology changes, and force interactions required for cutting. To systematically study these challenges, we introduce CulinaryCut, a benchmark that integrates an MPM-based deformable simulator with a robot environment to generate 21,000 physics-grounded cutting trajectories across fifteen food items. By combining visual scene variation across fifteen indoor background scenes with language augmentation, these trajectories are expanded to approximately 1.57M total instances, each paired with language instructions, multi-view observations, continuous control actions, and contact-force profiles. Physical plausibility is assessed against force measurements

* Equal contribution.

from a real Franka arm. Importantly, these forces are embedded in the dataset rather than provided as policy inputs, enabling evaluation of existing VLAs while exposing where physical grounding is needed. Using three representative VLA baselines, we identify two core limitations in non-rigid cutting. First, a geometry gap: VLAs struggle to map ratio- and direction-based instructions onto object geometry, especially when sequential cuts change the object's topology. Second, a physics gap: because policies output motion without a notion of material stiffness or contact resistance, a geometrically accurate path may still fail to sever the object. We show that training VLAs on physics-grounded trajectories, without modifying the policy architecture or adding force inputs, improves cutting success and sim-to-real transfer. Together, CulinaryCut provides a foundation for evaluating spatial reasoning and physical plausibility in deformable manipulation.

**Keywords:** Vision-Language-Action Model · Deformable Object Manipulation · Physics-Based Simulation

## 1 Introduction

Vision-Language-Action (VLA) models have achieved strong results in robotic manipulation by integrating visual perception, language understanding, and action generation into a single policy pipeline [5, 6, 8, 23, 31, 37]. Much of this progress, however, has been demonstrated on manipulation tasks involving rigid objects, where success is often determined by spatial alignment, object pose, or reaching a target configuration. These tasks are well suited for evaluating visual grounding and action execution, but they do not fully capture manipulation problems in which task success depends on the physical interaction generated during the motion.

Food cutting is a representative example of such a contact-rich task. To cut a material, a knife must not only follow a plausible geometric path, but also generate sufficient contact forces relative to the object's stiffness, geometry, and internal structure. When an object deforms, separates, or resists the tool, the same nominal motion can lead to different outcomes depending on the force profile it induces, as illustrated in Fig. 1. Thus, a trajectory that appears geometrically correct may still fail to sever the material. Despite being a fundamental everyday skill, food cutting remains largely unexplored in VLA research.

This limitation is closely tied to the data on which current VLAs are trained and evaluated. Standard robot demonstrations typically encode the executed motion, but not the material-dependent contact interactions required for successful cutting. Existing large-scale robot learning datasets focus mainly on rigid-body manipulation [12, 29, 36, 41, 52] and do not capture the forces, deformations, or topological changes that arise during cutting, as summarized in Table 1. Cutting-specific simulators and datasets, such as DiSECt [19], TopoCut [53], SliceIt! [3], and RoboCook [43], model fine-grained material behavior, but they do not jointly provide the physics-grounded contact signals, language instructions, multi-view

observations, and continuous robot control trajectories required for VLA training and evaluation.

Recent efforts have begun to incorporate physical cues into robot policies. For example, ForceVLA [59] explores force-conditioned policies by using force information as an additional input. In contrast, our goal is not to design a specialized force-conditioned architecture, but to study whether standard VLA models can learn physics-grounded cutting behaviors when physics is incorporated at the dataset generation stage. This requires a benchmark in which physical interaction, not only geometric motion or visual realism, is faithfully represented.

In this paper, we introduce *CulinaryCut*, a physics-grounded food-cutting benchmark generated using *IntSim*, our integrated simulator that couples an MLS-MPM simulator [20] with ManiSkill. *CulinaryCut* contains 21,000 trajectories across fifteen food items and is expanded to ∼1.57M instances through visual and language augmentation. Each instance includes multi-view observations, continuous actions, language instructions, and contact-force profiles, with physical plausibility validated using a real Franka arm.

Using *CulinaryCut*, we evaluate three representative VLA models: RDT [37], Octo [49], and OpenVLA [31]. Our analysis reveals two core limitations. First, models exhibit a geometry gap: they struggle to perceive object extent and ground ratio- and direction-based instructions, especially for small objects and multi-object scenes. Second, models exhibit a physics gap: even when the predicted trajectory is geometrically plausible, it may fail to cut the material because the policy does not account for material stiffness or contact resistance. We show that this physics gap can be reduced without modifying the policy architecture or providing force inputs at inference time, by training on physics-grounded cutting data, improving both cutting success and sim-to-real transfer.

Based on these observations, our contributions are as follows:

1. **Physics-grounded cutting benchmark.** We introduce a benchmark that integrates an MPM-based physics simulator with a robot simulator, providing a large-scale food-cutting dataset and evaluation protocol with force profiles, multi-view observations, continuous actions, and language instructions.
2. **Analysis and data-level mitigation of geometry and physics gaps.** We identify a geometry gap in spatial instruction grounding and a physics gap in material-dependent cutting dynamics, and show that training on physics-grounded trajectories improves cutting success without modifying the policy architecture or providing force inputs at inference time.
3. **Scalable data generation pipeline.** We combine LLM-based language-instruction synthesis with simulation-driven trajectory augmentation to support large-scale dataset construction.
4. **Physical plausibility assessment.** We compare simulated contact-force curves against measurements from a real Franka robotic arm on a representative cutting task.

## 2 Related Work

**Table 1:** Comparison of representative robot manipulation and cutting datasets relevant to VLA-based food cutting. Large-scale manipulation datasets may provide language-conditioned or continuous robot trajectories but generally lack cutting support, while cutting-specific simulators model material behavior but do not jointly provide language instructions, multi-view observations, and continuous actions required for VLA training and evaluation.

| Dataset | Physics Sim | Robot Sim | Data Origin | Multi Camera | Language Instruction | Continuous Action | Cutting Support |
|---|---|---|---|---|---|---|---|
| DROID [29] | – | – | Real | ✓ | ✓ | ✓ | – |
| Arnold [16] | – | IsaacSim | Synthetic | ✓ | – | ✓ | – |
| RoboChop [14] | – | – | Real | – | – | – | ✓ |
| RoboCook [43] | MPM | – | Hybrid | ✓ | – | – | ✓ |
| Jamdagni et al. [25] | FEM | – | Synthetic | – | – | – | ✓ |
| SliceIt! [3] | FEM | Gazebo | Synthetic | – | – | – | ✓ |
| DiSECt [19] | FEM | – | Synthetic | ✓ | – | – | ✓ |
| TopoCut [53] | MPM | – | Synthetic | ✓ | – | – | ✓ |
| **CulinaryCut (Ours)** | **MPM** | **ManiSkill** | **Synthetic** | ✓ | ✓ | ✓ | ✓ |

### 2.1 VLA Models for Robotic Manipulation

VLA (Vision-Language-Action) models have rapidly evolved toward unifying perception, language, and action within a single policy backbone. Early approaches primarily focused on behavior imitation [51], and later expanded toward language-conditioned generalist agents [42]. Recent models such as RT-2 [6], OpenVLA [31], RDT [37], Octo [49], $\pi_0$ [5], and GR00T N1 [4] have advanced VLA architectures toward large-scale data-driven learning and cross-embodiment generalization. In parallel, reasoning-augmented VLAs incorporate self-talk, VLM-to-VLA transfer, and long-horizon task planning [21,34,48,60,61], and trajectory-centric designs directly map language to motion and control token sequences [9, 27, 56]. Despite this progress, existing VLAs and visuomotor baselines such as Diffusion Policy [11] and ACT [62] have been primarily trained and evaluated on rigid-object manipulation, and their capabilities remain insufficiently validated on contact-rich tasks such as cutting, where complex physical interactions, deformation, and material separation are central. ForceVLA [59] recently explored force-conditioned policies by using force cues as additional inputs; in contrast, we incorporate physics into the dataset generation process, enabling standard VLAs to learn and be evaluated on physics-grounded cutting behaviors without requiring a specialized force-conditioned architecture.

### 2.2 Benchmarks and Datasets for Manipulation

This limitation is partly attributable to the lack of suitable training and evaluation data for contact-rich cutting tasks. Large-scale robot learning datasets, such

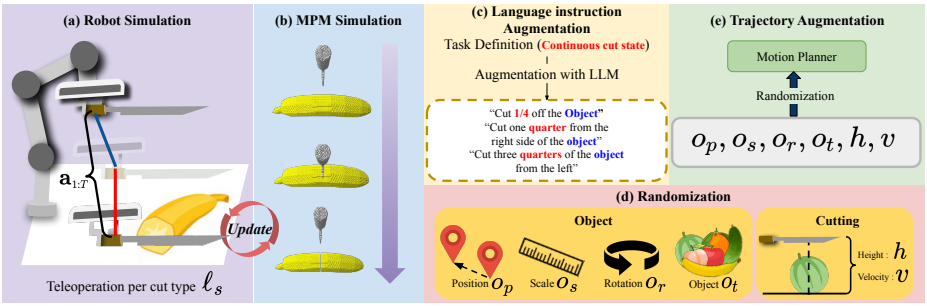



**Fig. 2:** Overview of CulinaryCut data generation pipeline. (a) Teleoperation generates initial cutting demonstrations for each cut style. (b) Physics simulation updates material deformation and cutting geometry. (c) Language instructions are augmented using LLMs with continuous cut state variations. (d, e) Trajectories are augmented via motion planning with object and cutting randomization.

as Open X-Embodiment [41], DROID [29], BridgeData V2 [52], LIBERO [36], and RoboMIND [57], provide diverse real-robot trajectories but do not include cutting; moreover, collecting large-scale real cutting data is difficult due to safety concerns, material variability, and the need for precise force control. Simulation offers an alternative, yet widely used benchmarks such as ManiSkill [17] do not address cutting, and deformable-manipulation benchmarks such as PlasticineLab [22] and SoftGym [35] do not cover cutting or material separation. Cutting-specific datasets and simulation studies, including DiSECt [19], TopoCut [53], SliceIt! [3], and RoboCook [43], focus on physics simulation and material behavior but do not jointly provide the language instructions, multi-view observations, and continuous control actions required for VLA training.

### 2.3 Simulation Realism and Physics-Grounded Cutting

Since our dataset is simulation-based, it inevitably involves a sim-to-real gap. A major component is the visual domain gap [50], for which recent foundation world models offer tools for improving visual realism and diversity. Models such as Cosmos [1], Genie [7], UniSim [58], and GenAug [10] improve visual realism by generating photorealistic or diverse scenes from structured inputs.

However, visual realism alone cannot resolve the physics gap. In cutting, contact forces, deformation, and topology changes are central to success, so realistic imagery is insufficient unless the simulation also encodes physically plausible contact-force signals and material state transitions. Cutting simulation commonly relies on physics-based methods such as the Material Point Method (MPM) and the Finite Element Method (FEM). MPM naturally handles large deformation and topology changes by coupling Lagrangian particles with an Eulerian grid [45, 46], with recent extensions supporting fracture modeling through continuum damage mechanics [54, 55]. FEM-based approaches have been explored for differentiable cutting and robotic slicing [3, 19, 25], and related work

in surgical robotics addresses deformable tissue cutting with similar challenges in contact, topology change, and force-sensitive interaction [15, 18].

Among these approaches, we adopt MLS-MPM [20] for its stable handling of large deformation and topology change while preserving rotational and shear responses inherited from APIC [26], and couple it with a robot simulator to build *CulinaryCut*.

## 3 CulinaryCut Benchmark

### 3.1 Overview

CulinaryCut is not a new model architecture, but a benchmark designed to diagnose the structural limitations of VLA models in non-rigid cutting tasks. In this paper, we target two such limitations. The ***geometry gap*** denotes the difficulty of grounding spatial, ratio-based, and directional instructions, such as "cut one-quarter from the right," in an object's actual geometry, especially when each cut changes the object's topology and requires subsequent cut positions to be recomputed. The ***physics gap*** captures the challenge of reasoning about the material-dependent physical interaction required to sever a deformable object. Many existing VLAs are primarily trained and evaluated as visual or geometric trajectory-following systems, without explicitly modeling force feedback, contact resistance, or material failure; as a result, a geometrically accurate cutting path alone does not guarantee a successful cut.

To identify these limitations, CulinaryCut encodes physics at the dataset level: each trajectory carries contact-force signals from our MLS-MPM simulator, embedded in the dataset rather than fed to the policy. Combining the MLS-MPM simulator, the ManiSkill environment, and an LLM-based instruction pipeline, CulinaryCut isolates the geometry gap through its task structure (Sec. 3.2) and operationalizes the physics gap through force-labeled trajectories. Fig. 2 shows the full generation pipeline.

### 3.2 Task Design

Tasks in CulinaryCut are designed with two main objectives: evaluating ratio-based spatial grounding and assessing sequential planning under topology changes.

**Task for spatial grounding.** This task requires the agent to translate language-specified proportions into precise cutting positions. Given commands such as "cut the object in half" or "cut one-quarter from the right," the agent must perceive object size and convert ratio expressions (1/2, 1/4, 1/6, etc.) into a cut location, while grounding directional references—the same ratio yields different positions for "left" versus "right" (defined w.r.t. the camera viewpoint). This directly tests whether a VLA can map ratios onto object geometry.

**Table 2: Material properties used in CulinaryCut.** Values are collected from the food-engineering literature and used to parameterize deformable objects in the MLS-MPM simulator.

| Object | Density ($\rho$) | Young's Modulus ($E$) | Poisson's Ratio ($\nu$) |
|---|---|---|---|
| Apple [30] | 850 $kg/m^3$ | $3.0 \times 10^6$ Pa | 0.33 |
| Cucumber [38] | 980 $kg/m^3$ | $1.0 \times 10^6$ Pa | 0.35 |
| Melon [39] | 1100 $kg/m^3$ | $1.2 \times 10^6$ Pa | 0.45 |
| Banana [24] | 960 $kg/m^3$ | $7.0 \times 10^3$ Pa | 0.35 |
| Orange [33] | 1020 $kg/m^3$ | $5.84 \times 10^5$ Pa | 0.24 |
| Peach [28] | 968 $kg/m^3$ | $8.9 \times 10^5$ Pa | 0.29 |
| Strawberry(Berry) [2] | 1010 $kg/m^3$ | $2.0 \times 10^5$ Pa | 0.40 |

**Task for sequential planning.** This task requires dividing an object into a specified number of equal pieces (e.g., "divide the cucumber into four pieces"). Because each cut changes the object's topology, the remaining cut positions must be recomputed, demanding both long-horizon planning and adaptation to geometric state changes. Success is measured by the number and uniformity of the resulting pieces.

### 3.3 Physics-grounded Simulation

Our simulator is built on MLS-MPM [20], which naturally handles large deformations and topology changes inherent to cutting. Each particle carries position, velocity, deformation gradient, and a damage scalar; corotated elasticity [44] is combined with J2 plasticity and a scalar damage model to represent per-material stiffness and cutting resistance. The fifteen food items are assigned material properties drawn from the food-engineering literature [2,24,28,30,33,38,39]. For brevity, Table 2 reports representative material parameters, including density, Young's modulus, Poisson's ratio, and their sources. Simulation parameters are initialized from these literature values and adjusted for numerical stability.

Crucially, this formulation makes cutting outcomes depend on the contact forces a trajectory induces, not merely on the path it follows. We estimate these forces from the momentum exchange between tool and material: the knife and board are represented as signed distance fields (SDFs), and the velocity change imposed on MPM particles at each grid node is converted into a reaction impulse, yielding the contact force. Because each trajectory thus carries a physics-grounded force-over-time signal, the dataset exposes the physics gap as a measurable property: a geometrically correct path that violates the material's force requirements fails to sever the object, as we show quantitatively in Sec. 4.4 (Table 4). These force signals also inform the velocity guardrail (Sec. 3.5) and are validated against real Franka measurements. Full equations and setup are in the Supplementary.

### 3.4 Data Generation

**Trajectory generation.** Demonstrations for each task are generated by an AABB-based motion planner that dynamically computes the target cutting po-

sition from the language instruction. The planner first estimates the spatial extent of the object from its bounding box, then determines the cut location according to the specified ratio and direction. The moment the tool first contacts the object surface is recorded as the contact phase; from that point onward, the physics simulator updates forces, deformation, and topology changes in real time. This process maintains physical consistency between visual observations, actions, and the resulting contact forces, providing physics-grounded supervision for imitation learning.

**Dataset statistics.** The dataset contains 21,000 cutting trajectories across fifteen food items (banana, cucumber, apple, peach, melon, orange, strawberry, pear, kiwi, tomato, lemon, grape, shine muscat, plum, and cherry), capturing object-level diversity through per-item geometric and material variations. We expand each trajectory along two augmentation axes: visual scene diversity, by rendering it across fifteen indoor background scenes drawn from ReplicaCAD [47], AI2-THOR [32], and ProcTHOR [13]; and language diversity, via at least five instruction variants each. Together these yield approximately 1.57M total instances ($21{,}000 \times 15 \times 5 = 1{,}575{,}000$).

**Language instructions.** A template-based language-generation engine synthesizes diverse textual commands for the same manipulation. Each template contains slots for object, position, ratio, and direction, which are randomly filled from a synonym pool (e.g., "50%," "half," "two quarters") to increase linguistic diversity. LLM-generated paraphrases [40] further broaden the range of expressions while preserving semantic content. Because the initial state is not given in text, the agent must infer the current situation directly from visual observation. The full list of templates and lexical variations is provided in the Supplementary Material.

**Evaluation protocol.** We evaluate VLAs along two complementary dimensions. First, we assess **geometric generalization** along three axes: (1) *unseen random seeds*: object positions and scales not seen during training; (2) *multi-object scenes*: multiple objects coexisting in the same workspace; (3) *unseen language instructions*: commands reserved exclusively for testing. Cross-domain evaluation (e.g., training on apple and evaluating on banana) further measures cross-category transfer. Second, we assess **physical severing** under MPM-coupled dynamics, measuring whether a trajectory induces the material-dependent force required to actually sever the object rather than merely following a geometrically correct path (IntSim; Sec. 4.4).

### 3.5 Baselines and Physics-grounded Comparison

**VLA baselines.** We evaluate three representative VLA models: RDT [37], Octo [49], and OpenVLA [31]. These models follow the standard VLA paradigm

of predicting robot actions from language instructions and visual or multimodal observations, while differing in architecture and training strategy. RDT models action trajectories with a diffusion-based transformer, Octo provides an open generalist policy architecture for cross-embodiment robot learning, and OpenVLA augments a vision-language backbone with an action prediction head. We use these models as representative baselines to test whether current VLA pipelines can handle the geometric and physical requirements of non-rigid cutting. All baselines are fine-tuned on the CulinaryCut training split before evaluation, and none receives force, contact, or damage-state information as policy input.

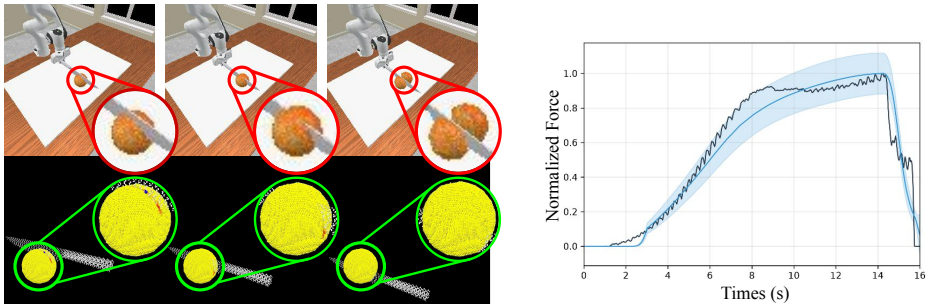
**Physics-grounded data.** Our goal is to test whether physics-grounded demonstrations improve cutting behavior without changing the VLA architecture or providing force observations at inference time. To this end, we compare policies trained on kinematic trajectory-only demonstrations against policies trained on our MPM-grounded demonstrations, whose actions are generated under deformable material dynamics and annotated with contact-force signals. The force signal is used to construct and analyze the dataset, but it is not provided as an input to the policy. Although explicit force values are omitted during training, these physics-grounded trajectories inherently encode material resistance into the action space, allowing the policy to implicitly learn the velocity micro-adjustments and sustained effort required to sever the object. This comparison isolates whether the physical consistency of the training data, rather than a force-conditioned policy architecture, improves actual cutting success (Table 4).

**Velocity guardrail.** For fair and safe task-level evaluation, we apply the same velocity guardrail to all baselines at execution time. The guardrail clips commands that would exceed the predefined safe operating range of the Franka arm, but it does not provide additional observations to the policy or re-plan the trajectory. Thus, the policy itself remains force-unaware; the guardrail acts only as an external safety filter applied uniformly across models. Implementation details are provided in the Supplementary Material.

## 4 Experiments

### 4.1 Experimental Setup

**Experiment detail.** We evaluate RDT [37], Octo [49], and OpenVLA [31] after fine-tuning them on the CulinaryCut training split. All evaluations are conducted in a simulated tabletop cutting environment with randomized object poses, positions, and scales. At the beginning of each episode, the target food item is randomly initialized within the workspace, and the policy receives either a standard cutting instruction or a ratio-based language instruction. For each object and task configuration, we conduct 20 randomized trials using seeds excluded from training and report the corresponding success rate.




**(a) Visual results** **(b) Real-world force validation**

**Fig. 3: Simulation results of IntSim.** (a) *IntSim* controls the robot using ManiSkill and updates food topology changes using an MLS-MPM simulator. The visual samples show the progressive topology change of an orange during cutting. (b) Real-world force validation between Franka F/T sensor measurements (black line) and *IntSim* contact-force estimates (blue line) during banana cutting. The normalized overlay shows close temporal agreement between real and simulated force profiles.

**Success criteria.** A cutting attempt is considered successful if all of the following conditions are satisfied: (1) The knife tip reaches the table surface with a positional error less than 0.05 m. (2) The blade accurately intersects the predefined cutting region of the object. (3) The blade–object contact angle is within $90° \pm 10°$. (4) The cutting location lies within one-tenth of the object size from the designated target position.

**Real robot setup.** A Franka Emika Panda arm equipped with a kitchen knife executes a single downward cut on a banana at constant speed. The wrist-mounted F/T sensor records the vertical reaction force at 60 Hz, with a pre-contact baseline of $\sim$25 N from the static end-effector load and sensor offset.

**MPM simulation.** The MLS-MPM framework adopts grid resolution $120^3$, $\Delta t = 2\times10^{-5}$ s, and 6 substeps to generate cutting scenarios. The knife collider moves downward at the prescribed cutting speed, and the contact force is computed from the accumulated impulse on material particles per output interval.

### 4.2 IntSim Validation

**Visual results.** Fig. 3(a) illustrates the role of the MPM component in *IntSim*. While ManiSkill provides the robot execution and visual observation, the MLS-MPM simulator tracks particle-level deformation and topology change of the food. This distinction is important because a trajectory that appears correct in the robot simulator does not necessarily produce material separation. The orange-cutting example shows that *IntSim* can directly evaluate whether the executed knife motion induces actual severing under material dynamics, rather than only geometric path following.

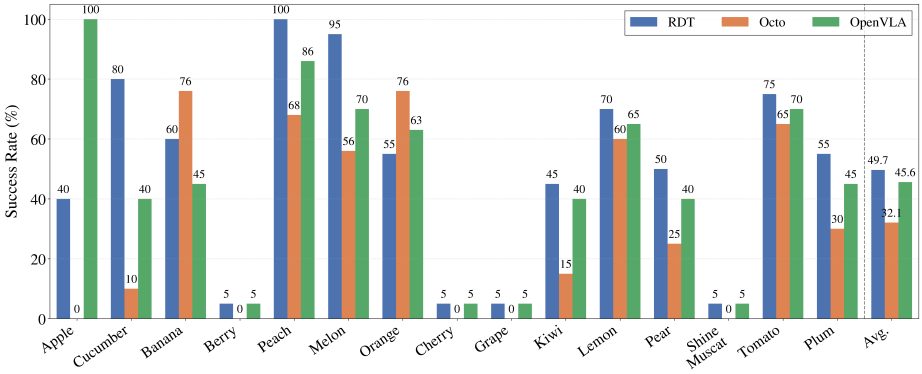



**Fig. 4: Single-object results (I).** The bar chart reports the success rate of each model across individual object categories under the single-object setting.

**Real-world force validation.** We further assess whether *IntSim* produces physically plausible contact-force profiles by comparing it with a real Franka Emika Panda cutting a banana. As shown in Fig. 3(b), the real F/T signal has a non-zero pre-contact baseline of approximately 25 N due to the tool weight, sensor bias, and robot-side force offset, whereas *IntSim* directly reports contact force without the robot-arm contribution. We therefore subtract the pre-contact baseline from the real measurement before comparing peak cutting forces. After baseline subtraction, the real robot produces a peak force of $\sim$12.5 N, while *IntSim* estimates $\sim$12.0 N, yielding a relative error within 5%. We then normalize both profiles by their respective peaks to compare temporal shape. The normalized curves agree well during the ramp-up and peak-force regions, indicating that the MPM contact model captures the dominant cutting dynamics. Remaining discrepancies near force release are likely caused by effects not fully modeled in the simulator, such as adhesion, fiber tearing, and compliance in the real robot control loop.

### 4.3 Geometry Generalization Results

We first evaluate the geometry gap of current VLA models, focusing on object-scale variation, target selection in multi-object scenes, unseen-object transfer, ratio-based grounding, and sequential split cuts.

**Performance under single-object scene (I).** Under the single-object setting, models achieve reasonable success on large and visually distinctive objects, but performance varies substantially across categories as visualized in Fig. 4. In particular, smaller objects and objects with limited visual extent lead to larger localization errors. This result indicates that even without distractors, current VLAs are sensitive to object scale and geometry when the task requires precise cutting-point localization.

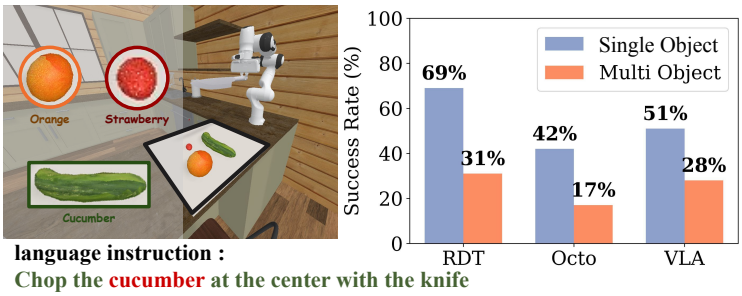



**Fig. 5: Multi-object results (II).** Additional objects significantly reduce cutting success rates compared to the single-object setting.

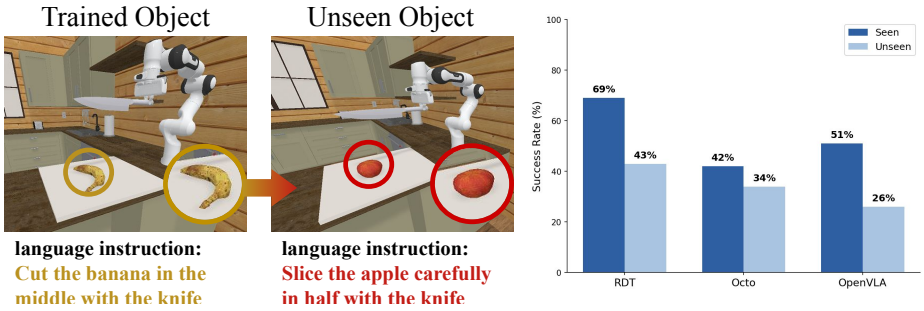



**Fig. 6: Generalization results to unseen objects with novel language instructions (III).** The bar chart presents success rates of RDT, Octo, and OpenVLA, showing a clear performance drop on unseen objects.

**Performance under multi-object scene (II).** When multiple food items are placed in the workspace, all models exhibit a clear drop in success rate, summarized in Fig. 5. RDT decreases from 69% to 31%, Octo from 42% to 17%, and OpenVLA from 51% to 28%. This suggests that target-object grounding under visual clutter remains fragile, even when the language instruction explicitly specifies the object to cut.

**Transfer-object performance (III).** We evaluate cross-category generalization using a leave-one-object-out setting, where each model is fine-tuned after excluding one target object category and evaluated only on that held-out category. This setting tests whether the learned cutting policy transfers to objects with unseen geometry, appearance, and material properties. As shown in Fig. 6, performance drops consistently on unseen objects: RDT decreases from 69% to 43%, Octo from 42% to 34%, and OpenVLA from 51% to 26%. The models retain partial transfer ability, but the substantial degradation shows that current VLAs remain sensitive to object-specific visual and geometric cues, revealing a clear object-generalization gap in food cutting.

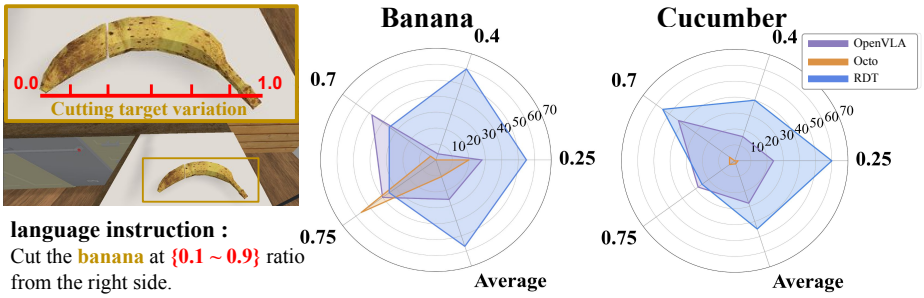



**Fig. 7: Evaluation of continuous cut (IV).** Success rates (%) of RDT, Octo, and OpenVLA under continuous ratio-based cutting instructions. The radar charts show model performance across cut ratios (0.1–0.9) for banana and cucumber.

**Numeric ratio grounding performance in continuous cut (IV).** The continuous-ratio benchmark exposes a systematic weakness in grounding numeric expressions to object-centric coordinates. Compared with standard center-cut instructions, ratio-based commands require the model to estimate object extent and convert expressions such as "0.4 from the right" into an accurate cutting position. As shown in Fig. 7, all models show unstable performance across nearby ratios, and Octo/OpenVLA often collapse at mid-range ratios. These results suggest that current VLA policies do not yet represent ratio and direction as continuous geometric variables.

**Sequential division performance in split cut (V).** Table 3 reports step-wise split-cut performance using RDT. Although RDT occasionally places the first cut within the target tolerance, success rapidly decreases in later steps. This failure mode is expected because each cut changes the remaining object geometry, so the next target position must be recomputed from the updated topology rather than from the original object extent. The near-zero success in later steps indicates that split cutting requires sequential geometric reasoning beyond local ratio grounding.

**Table 3: Split cut evaluation with RDT (V).** Each cell reports the success rate for each step, measured by whether the target cut position falls within the ±10% tolerance range.

| Object | Split Cut | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| Banana | 1/3 | 20% | 0% | – | – |
| | 1/4 | 10% | 5% | 0% | – |
| | 1/5 | 5% | 0% | 0% | 0% |
| Cucumber | 1/3 | 30% | 0% | – | – |
| | 1/4 | 10% | 0% | 0% | – |
| | 1/5 | 0% | 0% | 0% | 0% |

## 4.4 Ablation Study

**Physics gap analysis.** The evaluations above measure whether the predicted trajectory satisfies the intended cutting geometry. We now examine the *physics gap*: even when the knife reaches the target region with the correct pose, the object may not be severed once material dynamics are considered. To isolate the effect of physics at the data level, rather than as an explicit policy input,

**Table 4: Effect of physics-grounded training** (RDT, orange cutting). The geometry-only metric (ManiSkill) hides the physics gap, which appears only once contact dynamics are simulated (IntSim).

| | Evaluation environment | | |
|---|---|---|---|
| Training environment | ManiSkill | IntSim | Real |
| ManiSkill | 62% | 23% | 1/10 |
| IntSim (Ours) | **68%** | **59%** | **3/10** |

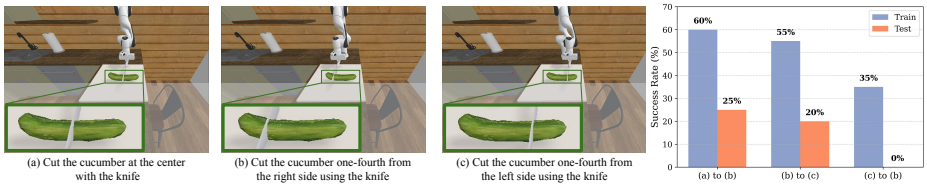



**Fig. 8: Directional and ratio generalization results.** (a), (b), and (c) illustrate ratio-based cutting tasks, while the bar chart on the right compares model performance between training and testing conditions, highlighting a clear performance drop on unseen ratio configurations.

we compare two RDT policies on orange cutting (Table 4). The first policy is trained on ManiSkill trajectories that satisfy the geometric cutting criteria, whereas the second is trained on *IntSim* trajectories rolled out and filtered under MLS-MPM-coupled material dynamics. Neither policy receives force, contact, or damage-state information as input. Both policies perform similarly under the ManiSkill evaluation, achieving 62% and 68% success, respectively. However, when evaluated in *IntSim*, where the trajectory must also physically sever the object, the ManiSkill-trained policy drops to 23%, while the *IntSim*-trained policy retains 59% success. This result shows that a geometrically valid cutting motion can still fail without the material-dependent interaction required for severing, such as sufficient velocity and sustained contact force. Since the model architecture and policy inputs are unchanged, the improvement from 23% to 59% indicates that the physics gap can be reduced through physics-grounded training data alone. The preliminary real-robot deployment is limited in scale, but follows the same trend, improving from 1/10 to 3/10 successful cuts.

**Directional and ratio-based generalization.** We further evaluate the RDT model, which achieved the highest overall performance, to analyze its ability to generalize across directional flips and cutting ratios. As summarized in Fig. 8, the model shows a significant degradation when trained on one configuration and evaluated on its directional or proportional counterpart. When the model is trained on 0.25-ratio cuts from the right side and then evaluated on the mirrored left-side instruction (e.g., "cut the banana at a 0.25 ratio from the left"), its success rate drops from 55% to 20%. Similarly, in the reverse direction (0.75 $\rightarrow$ 0.25), the success rate decreases from 35% to 0%, and when transferring from

the 0.5-ratio to 0.25-ratio instruction, performance drops from 60% to 25%. Even the strongest diffusion-based policy fails to generalize symmetrically across directional and ratio-based instructions. To bridge this gap, we include a *continuous-cut* split within CulinaryCut that provides dense ratio supervision (0.1–0.9) with explicit left/right labels, enabling continuous grounding of cutting geometry.

## 5 Conclusion

We presented CulinaryCut, a physics-grounded benchmark for food cutting that integrates an MLS-MPM simulator with a robot simulator to provide 21,000 cutting trajectories spanning fifteen food items, paired with contact-force profiles and language instructions. Evaluating three representative VLA models reveals two core limitations. The *geometry gap*: models struggle to map ratio- and direction-based instructions onto object geometry. The *physics gap*: because policies output motion without modeling material stiffness or contact resistance, a geometrically accurate path may still fail to sever the object. We further show that this physics gap can be narrowed not by modifying the policy architecture or adding force inputs, but by training on physics-grounded data, which improves both cutting success and sim-to-real transfer. CulinaryCut thus offers a foundation for evaluating and beginning to close the spatial and physical gaps that separate current VLAs from reliable deformable manipulation.

*Limitations and Future Directions.* CulinaryCut currently covers fifteen food items; extending it to broader food categories (e.g., bread, meat) and non-food deformable objects would strengthen generality. Although validated against real-robot forces, the MLS-MPM simulator does not model viscoelastic relaxation or moisture-dependent changes, and our validation so far covers only a single representative item; broader multi-material validation remains future work. Finally, while physics-grounded data improves the realism of contact dynamics, a visual sim-to-real gap remains. A natural next step is to combine physics-grounded simulation with visually enhanced world models such as Cosmos [1], jointly bridging the visual and physics gaps to generate data that is both photorealistic and physically consistent.

## Acknowledgements

This research was supported by the grants of IITP (the Advanced GPU Utilization Support Program, No. 2024-0-00071, No. RS-2024-00430997, No. RS-2024-00426853), NRF (No. RS-2025-24803335), MOE (No. RS-2025-25441317), SGM (2025-RISE-01-020-04), the KIST Institutional Program (Project No. 26E0052), the Basic Science Research Program (No. RS-2021-NR060140), and KISA (No. RS-2026-25530576) funded by Korea. D.H. was supported by the United States National Science Foundation Award #2450401. NVIDIA Corporation is acknowledged for providing GPUs used in this work.

## References


1. Agarwal, N., Ali, A., Bala, M., Balaji, Y., Barker, E., Cai, T., Chattopadhyay, P., Chen, Y., Cui, Y., Ding, Y., Dworakowski, D., Fan, J., Fenzi, M., Ferroni, F., Fidler, S., Fox, D., Ge, S., Ge, Y., Gu, J., Gururani, S., He, E., Huang, J., Huffman, J., Jannaty, P., Jin, J., Kim, S.W., Klár, G., Lam, G., Lan, S., Leal-Taixe, L., Li, A., Li, Z., Lin, C.H., Lin, T.Y., Ling, H., Liu, M.Y., Liu, X., Luo, A., Ma, Q., Mao, H., Mo, K., Mousavian, A., Nah, S., Niverty, S., Page, D., Paschalidou, D., Patel, Z., Pavao, L., Ramezanali, M., Reda, F., Ren, X., Sabavat, V.R.N., Schmerling, E., Shi, S., Stefaniak, B., Tang, S., Tchapmi, L., Tredak, P., Tseng, W.C., Varghese, J., Wang, H., Wang, H., Wang, H., Wang, T.C., Wei, F., Wei, X., Wu, J.Z., Xu, J., Yang, W., Yen-Chen, L., Zeng, X., Zeng, Y., Zhang, J., Zhang, Q., Zhang, Y., Zhao, Q., Zolkowski, A.: Cosmos world foundation model platform for physical ai (2025), arXiv preprint arXiv:2501.03575
2. An, X., Kopka, J., Rode, M., Zude-Sasse, M.: Relationship of cell size distribution and biomechanics of strawberry fruit under varying ca and n supply. Food Bioprocess Technol. (2025)
3. Beltran-Hernandez, C.C., Erbetti, N., Hamaya, M.: Sliceit! – a dual simulator framework for learning robot food slicing. In: ICRA (2024)
4. Bjorck, J., Castañeda, F., Cherniadev, N., Da, X., Ding, R., Fan, L.J., Fang, Y., Fox, D., Hu, F., Huang, S., Jang, J., Jiang, Z., Kautz, J., Kundalia, K., Lao, L., Li, Z., Lin, Z., Lin, K., Liu, G., Llontop, E., Magne, L., Mandlekar, A., Narayan, A., Nasiriany, S., Reed, S., Tan, Y.L., Wang, G., Wang, Z., Wang, J., Wang, Q., Xiang, J., Xie, Y., Xu, Y., Xu, Z., Ye, S., Yu, Z., Zhang, A., Zhang, H., Zhao, Y., Zheng, R., Zhu, Y.: Gr00t n1: An open foundation model for generalist humanoid robots (2025), arXiv preprint arXiv:2503.14734
5. Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Shi, L.X., Tanner, J., Vuong, Q., Walling, A., Wang, H., Zhilinsky, U.: $\pi_0$: A vision-language-action flow model for general robot control (2024), arXiv preprint arXiv:2410.24164
6. Brohan, A.: Rt-2: Vision-language-action models transfer web knowledge to robotic control. In: CoRL (2023)
7. Bruce, J., Dennis, M.D., Edwards, A., Parker-Holder, J., Shi, Y., Hughes, E., Lai, M., Mavalankar, A., Steigerwald, R., Apps, C., et al.: Genie: Generative interactive environments. In: ICML (2024)
8. Bu, Q., Yang, Y., Cai, J., Gao, S., Ren, G., Yao, M., Luo, P., Li, H.: Univla: Learning to act anywhere with task-centric latent actions (2025), arXiv preprint arXiv:2505.06111
9. Bucker, A., Figueredo, L., Haddadin, S., Kapoor, A., Ma, S., Vemprala, S., Bonatti, R.: Latte: Language trajectory transformer. In: ICRA (2023)
10. Chen, Z., Kiami, S., Gupta, A., Kumar, V.: Genaug: Retargeting behaviors to unseen situations via generative augmentation. In: RSS (2023)
11. Chi, C., Xu, Z., Feng, S., Cousineau, E., Du, Y., Burchfiel, B., Tedrake, R., Song, S.: Diffusion policy: Visuomotor policy learning via action diffusion. In: RSS (2023)
12. Dasari, S., Ebert, F., Tian, S., Nair, S., Bucher, B., Schmeckpeper, K., Singh, S., Levine, S., Finn, C.: Robonet: Large-scale multi-robot learning. In: CoRL (2019)
13. Deitke, M., VanderBilt, E., Herrasti, A., Weihs, L., Ehsani, K., Salvador, J., Han, W., Kolve, E., Kembhavi, A., Mottaghi, R.: Procthor: Large-scale embodied ai using procedural generation. Advances in Neural Information Processing Systems **35**, 5982–5994 (2022)

14. Dikshit, A., Bartsch, A., George, A., Farimani, A.B.: Robochop: Autonomous framework for fruit and vegetable chopping leveraging foundational models (2023), arXiv preprint arXiv:2307.13159
15. Ge, J., Kilmer, E., Mady, L.J., Opfermann, J.D., Krieger, A.: Enhancing surgical precision in autonomous robotic incisions via physics-based tissue cutting simulation. In: IROS (2024)
16. Gong, R., Huang, J., Zhao, Y., Geng, H., Gao, X., Wu, Q., Ai, W., Zhou, Z., Terzopoulos, D., Zhu, S.C., et al.: Arnold: A benchmark for language-grounded task learning with continuous states in realistic 3d scenes. In: ICCV (2023)
17. Gu, J., Xiang, F., Li, X., Ling, Z., Liu, X., Mu, T., Tang, Y., Tao, S., Wei, X., Yao, Y., Yuan, X., Xie, P., Huang, Z., Chen, R., Su, H.: Maniskill2: A unified benchmark for generalizable manipulation skills. In: ICLR (2023)
18. He, S., Qian, Y., Zhu, X., Liao, X., Heng, P.A., Feng, Z., Wang, Q.: Versatile cutting fracture evolution modeling for deformable object cutting simulation. Comput. Methods Programs Biomed. (2022)
19. Heiden, E., Macklin, M., Narang, Y., Fox, D., Garg, A., Ramos, F.: Disect: A differentiable simulator for parameter inference and control in robotic cutting. In: RSS (2021)
20. Hu, Y., Fang, Y., Ge, Z., Qu, Z., Zhu, Y., Pradhana, A., Jiang, C.: A Moving Least Squares Material Point Method with Displacement Discontinuity and Two-way Rigid Body Coupling. ACM TOG (2018)
21. Huang, W., Xia, F., Xiao, T., Chan, H., Liang, J., Florence, P., Zeng, A., Tompson, J., Mordatch, I., Chebotar, Y., Sermanet, P., Brown, N., Jackson, T., Luu, L., Levine, S., Hausman, K., Ichter, B.: Inner monologue: Embodied reasoning through planning with language models. In: CoRL (2022)
22. Huang, Z., Hu, Y., Du, T., Zhou, S., Su, H., Tenenbaum, J.B., Gan, C.: Plasticinelab: A soft-body manipulation benchmark with differentiable physics. In: ICLR (2021)
23. Intelligence, P., Black, K., Brown, N., Darpinian, J., Dhabalia, K., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Galliker, M.Y., Ghosh, D., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., LeBlanc, D., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Ren, A.Z., Shi, L.X., Smith, L., Springenberg, J.T., Stachowicz, K., Tanner, J., Vuong, Q., Walke, H., Walling, A., Wang, H., Yu, L., Zhilinsky, U.: $\pi_{0.5}$: a vision-language-action model with open-world generalization (2025), arXiv preprint arXiv:2504.16054
24. Jahanbakhshi, A., Yeganeh, R., Shahgoli, G.: Determination of mechanical properties of banana fruit under quasi-static loading in pressure, bending, and shearing tests. Int. J. Fruit Sci. (2020)
25. Jamdagni, P., Jia, Y.B.: Robotic slicing of fruits and vegetables: Modeling the effects of fracture toughness and knife geometry. In: ICRA (2021)
26. Jiang, C., Schroeder, C., Teran, J.: An Angular Momentum Conserving Affine-Particle-in-Cell Method. J. Comput. Phys. (2017)
27. Jiang, Y., Gupta, A., Zhang, Z., Wang, G., Dou, Y., Chen, Y., Fei-Fei, L., Anandkumar, A., Zhu, Y., Fan, L.: Vima: General robot manipulation with multimodal prompts. In: ICML (2023)
28. Kabas, O., Vladut, V.: Determination of drop-test behavior of a sample peach using finite element method. Int. J. Food Prop. (2015)
29. Khazatsky, A., et al.: Droid: A large-scale in-the-wild robot manipulation dataset. In: RSS (2024)

30. Kheiralipour, K., Tabatabaeefar, A., Mobli, H., Rafiee, S., Sahraroo, A.S., Rajabipour, A., Jafari, A., et al.: Some physical properties of apple. Pak. J. Nutr. (2008)
31. Kim, M.J., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., Vuong, Q., Kollar, T., Burchfiel, B., Tedrake, R., Sadigh, D., Levine, S., Liang, P., Finn, C.: Openvla: An open-source vision-language-action model. In: CoRL (2024)
32. Kolve, E., Mottaghi, R., Han, W., VanderBilt, E., Weihs, L., Herrasti, A., Deitke, M., Ehsani, K., Gordon, D., Zhu, Y., et al.: Ai2-thor: An interactive 3d environment for visual ai (2017), arXiv preprint arXiv:1712.05474
33. Kurniawati, E., Rakhmadevi, A.G., Fadhila, P.T., Lailah, S.: Determination of physical properties of siamese oranges (citrus nobilis var. microcarpa): sphericity, roundness, volume, and density. J. Agritechnol. Food Process. (2023)
34. Li, X., Liu, M., Zhang, H., Yu, C., Xu, J., Wu, H., Cheang, C., Jing, Y., Zhang, W., Liu, H., Li, H., Kong, T.: Vision-language foundation models as effective robot imitators. In: ICLR (2024)
35. Lin, X., Wang, Y., Olkin, J., Held, D.: Softgym: Benchmarking deep reinforcement learning for deformable object manipulation. In: CoRL (2021)
36. Liu, B., Wang, Z., Ding, Z., et al.: Libero: Benchmarking knowledge transfer for lifelong robot learning. In: NeurIPS (2023)
37. Liu, S., Wu, L., Li, B., Tan, H., Chen, H., Wang, Z., Xu, K., Su, H., Zhu, J.: Rdt-1b: a diffusion foundation model for bimanual manipulation. In: ICLR (2025)
38. Mattos, F., Fasina, O., Reina, L., Fleming, H., Breidt Jr, F., Damasceno, G., Passos, F.V.: Heat transfer and microbial kinetics modeling to determine the location of microorganisms within cucumber fruit. J. Food Sci. (2005)
39. Nourain, J., Ying, Y.b., Wang, J.p., Rao, X.q., Yu, C.g.: Firmness evaluation of melon using its vibration characteristic and finite element analysis. J. Zhejiang Univ. Sci. B (2005)
40. OpenAI: Gpt-4 technical report (2023), arXiv preprint arXiv:2303.08774
41. O'Neill, A., Rehman, A., Maddukuri, A., Gupta, A., Padalkar, A., Lee, A., Pooley, A., Gupta, A., Mandlekar, A., Jain, A., et al.: Open x-embodiment: Robotic learning datasets and rt-x models. In: ICRA (2024)
42. Reed, S., Zolna, K., Parisotto, E., Colmenarejo, S.G., Novikov, A., Barth-Maron, G., Gimenez, M., Sulsky, Y., Kay, J., Springenberg, J.T., Eccles, T., Bruce, J., Razavi, A., Edwards, A., Heess, N., Chen, Y., Hadsell, R., Vinyals, O., Bordbar, M., de Freitas, N.: A generalist agent. TMLR (2022)
43. Shi, H., Xu, H., Clarke, S., Li, Y., Wu, J.: Robocook: Long-horizon elasto-plastic object manipulation with diverse tools. In: CoRL (2023)
44. Stomakhin, A., Howes, R., Schroeder, C.A., Teran, J.M.: Energetically consistent invertible elasticity. In: SCA (2012)
45. Stomakhin, A., Schroeder, C., Chai, L., Teran, J., Selle, A.: A Material Point Method for Snow Simulation. ACM TOG (2013)
46. Sulsky, D., Chen, Z., Schreyer, H.: A Particle Method for History-Dependent Materials. Comput. Methods Appl. Mech. Eng. (1994)
47. Szot, A., Clegg, A., Undersander, E., Wijmans, E., Zhao, Y., Turner, J., Maestre, N., Mukadam, M., Chaplot, D.S., Maksymets, O., et al.: Habitat 2.0: Training home assistants to rearrange their habitat. In: NeurIPS (2021)
48. Team, G.R., Abeyruwan, S., Ainslie, J., Alayrac, J.B., Arenas, M.G., Armstrong, T., et al.: Gemini robotics: Bringing ai into the physical world (2025), arXiv preprint arXiv:2503.20020

49. Team, O.M., Ghosh, D., Walke, H., Pertsch, K., Black, K., Mees, O., Dasari, S., Hejna, J., Kreiman, T., Xu, C., Luo, J., Tan, Y.L., Chen, L.Y., Sanketi, P., Vuong, Q., Xiao, T., Sadigh, D., Finn, C., Levine, S.: Octo: An open-source generalist robot policy. In: RSS (2024)
50. Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P.: Domain randomization for transferring deep neural networks from simulation to the real world. In: IROS (2017)
51. Torabi, F., Warnell, G., Stone, P.: Behavioral cloning from observation. In: IJCAI (2018)
52. Walke, H., Black, K., Lee, A., Kim, M.J., Du, M., Zheng, C., Zhao, T., Hansen-Estruch, P., Vuong, Q., He, A., Myers, V., Fang, K., Finn, C., Levine, S.: Bridgedata v2: A dataset for robot learning at scale. In: CoRL (2023)
53. Wang, L., Bian, J., Heiden, E., Garg, A.: Topocut: Learning multi-step cutting with spectral rewards and discrete diffusion policies (2025), arXiv preprint arXiv:2509.19712
54. Wang, S., Ding, M., Gast, T.F., Zhu, L., Gagniere, S., Jiang, C., Teran, J.M.: Simulation and visualization of ductile fracture with the material point method. Proc. ACM Comput. Graph. Interact. Tech. (2019)
55. Wolper, J., Fang, Y., Li, M., Lu, J., Gao, M., Jiang, C.: CD-MPM: Continuum Damage Material Point Methods for Dynamic Fracture Animation. ACM TOG (2019)
56. Wu, H., Jing, Y., Cheang, C., Chen, G., Xu, J., Li, X., Liu, M., Li, H., Kong, T.: Unleashing large-scale video generative pre-training for visual robot manipulation. In: ICLR (2024)
57. Wu, K., Hou, C., Liu, J., Che, Z., Ju, X., Yang, Z., Li, M., Zhao, Y., Xu, Z., Yang, G., Fan, S., Wang, X., Liao, F., Zhao, Z., Li, G., Jin, Z., Wang, L., Mao, J., Liu, N., Ren, P., Zhang, Q., Lyu, Y., Liu, M., He, J., Luo, Y., Gao, Z., Li, C., Gu, C., Fu, Y., Wu, D., Wang, X., Chen, S., Wang, Z., An, P., Qian, S., Zhang, S., Tang, J.: Robomind: Benchmark on multi-embodiment intelligence normative data for robot manipulation. In: RSS (2025)
58. Yang, Z., Chen, Y., Wang, J., Manivasagam, S., Ma, W.C., Yang, A.J., Urtasun, R.: Unisim: A neural closed-loop sensor simulator. In: CVPR (2023)
59. Yu, J., Liu, H., Yu, Q., Ren, J., Hao, C., Ding, H., Huang, G., Huang, G., Song, Y., Cai, P., et al.: ForceVLA: Enhancing VLA models with a force-aware MoE for contact-rich manipulation (2025), arXiv preprint arXiv:2505.22159
60. Zawalski, M., Chen, W., Pertsch, K., Mees, O., Finn, C., Levine, S.: Robotic control via embodied chain-of-thought reasoning. In: CoRL (2024)
61. Zhao, Q., Lu, Y., Kim, M.J., Fu, Z., Zhang, Z., Wu, Y., Li, Z., Ma, Q., Han, S., Finn, C., Handa, A., Liu, M.Y., Xiang, D., Wetzstein, G., Lin, T.Y.: Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In: CVPR (2025)
62. Zhao, T.Z., Kumar, V., Levine, S., Finn, C.: Learning fine-grained bimanual manipulation with low-cost hardware. In: RSS (2023)