MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Find the Fruit: Zero-Shot Sim2Real RL for Occlusion-Aware Plant Manipulation

Subedi, Nitesh; Yang, Hsin-Jung; Jha, Devesh K.; Sarkar, Soumik; Raghunathan, Arvind

TR2026-097 July 01, 2026

Abstract

Autonomous harvesting in the open presents a complex manipulation problem. In most scenarios, an autonomous system has to deal with significant occlusion and require interaction in the presence of large structural uncertainties (every plant is different). Perceptual and modeling uncertainty make design of reliable manipulation controllers for harvesting challenging, resulting in poor performance during deployment. We present a sim2real reinforcement learning (RL) framework for occlusion-aware plant manipulation, where a policy is learned entirely in simulation to reposition stems and leaves to reveal target fruit(s). In our proposed approach, we decouple high-level kinematic planning from low-level compliant control which simplifies the sim2real transfer. This decomposition allows the learned policy to generalize across multiple plants with different stiffness and morphology. In experiments with multiple real-world plant setups, our system achieves up to 86.7% success in exposing target fruits, demonstrating robustness to occlusion variation and structural uncertainty.

IEEE International Conference on Robotics and Automation (ICRA) 2026

© 2026 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Find the Fruit: Zero-Shot Sim2Real RL for Occlusion-Aware Plant Manipulation

Nitesh Subedi[1*], Hsin-Jung Yang[1*], Devesh K. Jha[2], Soumik Sarkar[1]

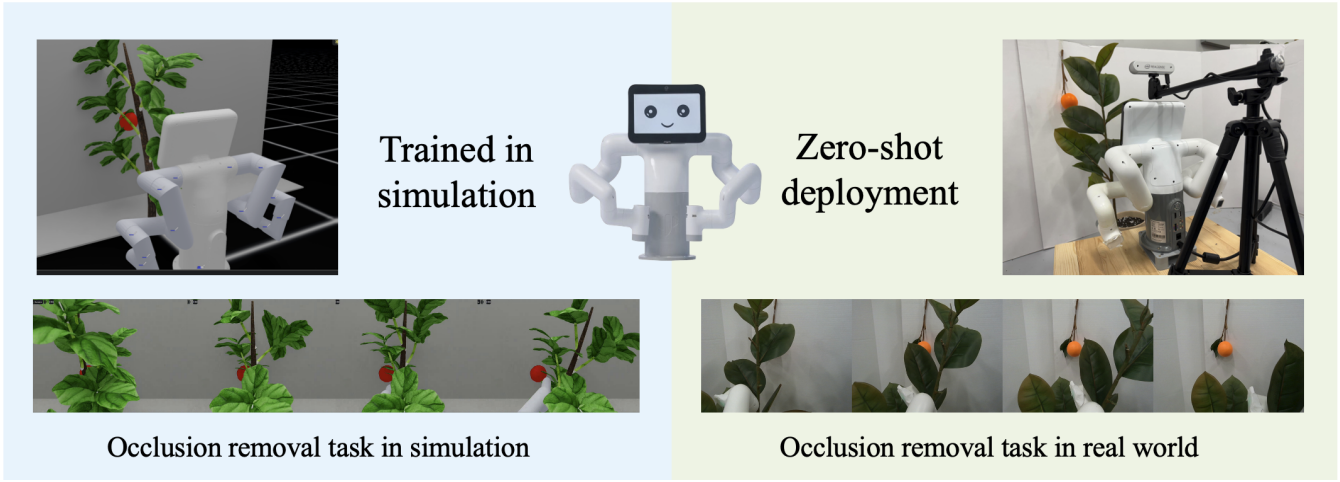

Fig. 1: **Top:** We present a sim2real approach for occlusion-aware plant manipulation to reveal fruits in cluttered environments. We perform end-to-end RL training on a generic deformable plant in simulation (left), and then deploy the trained policy in a zero-shot manner on a real stemmed plant (right) using the MyBuddy 280 robot (center). **Bottom:** Frame-by-frame sequences of occlusion removal in simulation (left) and real-world trials (right), showing that the policy trained on an abstract model generalizes to experimental plants.

***Abstract*— Autonomous harvesting in the open presents a complex manipulation problem. In most scenarios, an autonomous system has to deal with significant occlusion and require interaction in the presence of large structural uncertainties (every plant is different). Perceptual and modeling uncertainty make design of reliable manipulation controllers for harvesting challenging, resulting in poor performance during deployment. We present a sim2real reinforcement learning (RL) framework for occlusion-aware plant manipulation, where a policy is learned entirely in simulation to reposition stems and leaves to reveal target fruit(s). In our proposed approach, we decouple high-level kinematic planning from low-level compliant control which simplifies the sim2real transfer. This decomposition allows the learned policy to generalize across multiple plants with different stiffness and morphology. In experiments with multiple real-world plant setups, our system achieves up to 86.7% success in exposing target fruits, demonstrating robustness to occlusion variation and structural uncertainty.**

I. Introduction

Autonomous harvesting robots must do more than grasping a fruit – they must first *find* it. In most real world scenarios, fruits are generally hidden deep within dense foliage, partially or completely occluded by leaves, stems, or branches. These occlusions slow down harvesting, increase missed detection rates, and often require manual repositioning before automated grasping can even begin. While recent advances have led to robust perception for visible fruits and reliable manipulation for grasping, the problem of occlusion-resolution for harvesting remains underexplored. Most existing harvesting systems either assume fruits are already visible [1], [2], [3], or adjust the robot itself to get a clearer view of the fruit [4], [5], [6], [7]. However, such strategies fail to scale in real scenarios, where visibility could vary with growth stage, fruit distribution, and foliage density. For robust harvesting, a robot needs to actively manipulate the plant to reveal fruits before grasping — we refer it as *occlusion-aware fruit discovery*. Addressing this problem is essential for enabling downstream systems to function robustly in real-world conditions.

Plant manipulation, at its core, relies on three tightly coupled ingredients: *perception*, to perceive the environment; *planning*, to determine actions based on perception; and *control*, to execute these actions. Each of these components present unique challenges in field environments, where fruits are embedded within dense, deformable vegetation.

Perception for harvesting is challenging due to significant occlusions: leaves, stems, and branches often hide fruits partially or completely, breaking the assumptions of standard detection pipelines [8], [9], [10]. Planning is complicated by cluttered and variable plant geometry [11], [12]. Interaction during manipulation is difficult due to deformable structure of plants. Flexible stems bend, leaves fold, and branches recoil unpredictably under contact. Designing controllers in such settings is difficult since stiffness, damping, and

[1]Department of Mechanical Engineering, Iowa State University, USA {nitesh, hjy, soumiks}@iastate.edu
[2]Mitsubishi Electric Research Laboratories, USA devesh.dkj@gmail.com
*Equal contribution, Appendix: https://tinyurl.com/ftfappendix

mass distribution vary across species and even between individuals [13], [14]. These uncertainties make open-loop strategies and model-based control brittle, especially when occlusions change dynamically during interaction.

Existing research on deformable object manipulation does not fully transfer to agricultural settings. Prior work has largely focused on ropes and chains [15], [16], [17], fabrics [18], [19], [20], [21], or biological tissues [22], often under controlled conditions with strong modeling assumptions. Agricultural studies have explored leaf removal [9], [23], navigating through the branches to reach parts of the plant [24], sequential exploration to predict newly revealed spaces [25], or branch repositioning prior to harvesting [26], [27], [28]. Other efforts target collision-free grasp planning [29], [30]. While valuable, many of these methods rely on explicit plant-part detection or detailed 3D reconstruction [31], [32], [33], which are computationally costly and hard to scale to highly deformable, variable foliage.

We believe that a closed-loop learning approach offers a promising alternative. As occlusions evolve during manipulation, classical motion planning methods often struggle [34], [35], [36]. Reinforcement learning (RL) adaptively couples perception and action, enabling agents to respond to plant deformation without brittle intermediate models. However, sim-to-real transfer remains challenging [37], [38], [39], particularly for deformable objects with difficult-to-model mechanics [40].

Our Approach: We propose an end-to-end RL framework for occlusion-aware manipulation of deformable plants with zero-shot sim-to-real transfer (see Fig. 1). The policy is trained entirely in simulation using a generic plant model built in NVIDIA Isaac Lab [41], with deformable dynamics and randomized occlusion patterns. In our simulation setup, the plant is represented using a finite element method (FEM) model rather than simplified kinematic chains or reduced-order approximations. This formulation directly captures bending, twisting, and nonlinear deformation of stems and leaves under contact, without relying on lumped stiffness parameters or geometric heuristics. By avoiding lower-order assumptions, the model preserves the continuous deformation behavior of foliage, providing the RL policy with interaction dynamics that are structurally faithful to real plants [13], [42]. This high-fidelity representation serves as a key inductive bias, enabling the learned policy to transfer more reliably to physical systems [43]. This model trains the policy entirely in simulation using PPO [44].

To enable real-world transfer, we combine extensive domain randomization with a compliant low-level controller on the real robot, decoupling high-level planning from uncertain contact dynamics. This design maintains robustness even when real plants differ significantly from the simulated model in stiffness or morphology. In addition to single-target discovery, we also validate the policy in scenarios with multiple fruits, where sequential uncovering is required, further highlighting the generality of the approach.

To the best of our knowledge, this is the first end-to-end RL framework for occlusion-aware fruit discovery that transfers zero-shot to real-world conditions. Table I provides a comparison of our approach with representative prior works. More specifically, we:

Ref.	Task	No Recons.	Visual	RL
Jacob et al. [24]	Reach	✓	✗	✓
Rijal et al. [28]	Branch	✗	✓	✗
Zhang et al. [25]	Explore	✓	✗	✗
Yao et al. [9]	Leaf	✗	✓	✗
Jenkins et al. [32]	Stalk	✗	✓	✗
Ours	Adaptive	✓	✓	✓

TABLE I: Comparison of representative approaches in agricultural or deformable manipulation. "No Recons." indicates no use of explicit 3D reconstruction or geometric modeling; "Visual" denotes visual servoing; "RL" refers to reinforcement learning. Task labels indicate the primary focus: *Branch* and *Leaf* involve respective manipulation, while *Stalk* refers to its detection.

- Introduce a task formulation for fruit discovery as a prerequisite to harvesting, targeting plant manipulation without explicit 3D reconstruction,
- Propose a decoupled architecture that combines RL-based kinematic planning with a compliant execution layer, improving robustness across plant morphologies,
- Demonstrate successful sim-to-real transfer, achieving up to 86.7% occlusion-resolution success in physical trials (vs. 96.1% in simulation) using only a generic plant model, and
- Highlight policy generalization, showing robustness across individuals of the same morphological class and extending naturally to multiple-fruit scenarios where sequential discovery is required.

II. Methodology

Our objective is to train a policy that, given visual and proprioceptive inputs, produces a sequence of manipulations to expose a designated fruit hidden by plant foliage. The key challenge lies in bridging the gap between high-level task intent ("expose the fruit") and low-level contact execution in an environment with uncertain, deformable dynamics.

To address this, we adopt a hierarchical control strategy: **(1) High-level RL policy:** Operates in configuration space, outputting incremental joint-angle commands that plan how to reduce occlusion over time. **(2) Low-level controller:** Executes these commands on the real robot, handling contact forces safely and adapting to plant stiffness without requiring the policy to model exact dynamics.

This decoupling enables the policy to learn generalizable strategies in simulation, where physical parameters are randomized, and transfer them to hardware without re-tuning for each plant. The policy leverages structural cues such as the central stem and adapts motion to the target's visibility state, producing compliant, task-directed interaction.

A. Reinforcement Learning Problem Formulation

We present the MDP formulation for training the RL policy, including state, action, and reward design as follows:

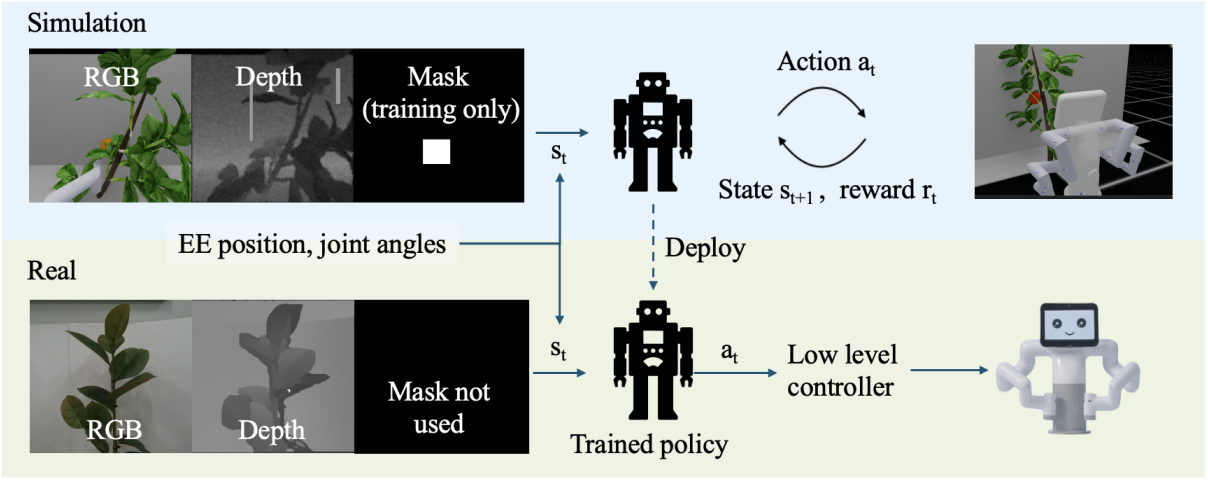

Fig. 2: Overview of the framework. **(a) Sim:** The robot interacts with an abstract plant in Isaac Lab, receiving RGB, depth, and fruit mask inputs (mask used only in training). The RL agent exchanges actions and rewards with the simulator to learn occlusion-aware manipulation. **(b) Real:** The trained policy is deployed on the MyBuddy 280 robot. An RGB-D camera provides observations, and a low-level controller executes high-level commands for safe interaction with real plants.

- State: The state s_t at time t is a composite of multi-modal inputs and proprioceptive data. Each state s_t includes an RGBA-D image ($I_{\text{RGBA-D},t}$, RGB image, alpha fruit mask, depth), joint angles for the waist ($J_{b,t}$) and the left arm ($J_{i,t}$, where $i = 1 \sim 5$, representing the different joints), and end-effector position ($EE_{\text{pos},t}$). It can be formally represented as $s_t = [I_{\text{RGBA-D},t}, J_{b,t}, J_{i,t}, EE_{\text{pos},t}]$.
- Action: The action $a_t = [\Delta j_{1\text{-}5}, \Delta j_b]$ are delta joint angles for the same joints, clipped to $[-1, 1]$ and scaled at runtime.
- Reward: The total reward r_t is a weighted sum of:
  (a) Occlusion reward, r_{occ} = (1 - occluded / fruit pixels)
  (b) Full visibility reward, $r_{\text{fv}} = 3$, upon satisfying the success criterion for 1 timestep (Section II-C).
  (c) Sustained visibility, $r_{sus} = 20$ if full visibility is maintained ≥ 10 steps; 0 otherwise.
  (d) Post-visibility penalty, r_{pv} = action magnitude after full visibility, penalizing post-visibility motion.
  (e) Self-collision penalty, $r_{sc} = -5$ if self collision; 0 otherwise.

Together, with the selected weights for each term, the reward function becomes:

$$r_i = 10.0 \cdot \left(1 - \frac{P_{\text{occ},i}}{40 \times 40}\right) + 3.0 \cdot \mathbb{1}_{[P_{\text{occ},i} \leq 160]} + 20.0 \cdot \mathbb{1}_{\text{sus},i} - 0.06 \cdot \|\mathbf{a}_i\|_2 \cdot \mathbb{1}_{[P_{\text{occ},i} \leq 160]} - 5.0 \cdot \mathbb{1}_{sc,i} \quad (1)$$

B. Simulation Environment Setup

The simulation environment, as shown in top right of Fig. 2, is developed using Isaac Lab from NVIDIA Omniverse, ensuring high-fidelity replication of the physical setup. The MyBuddy 280 robot is imported as an URDF model derived from SolidWorks, with joint positions controlled to mirror the behavior of the physical robot closely. Artificial plants are modeled as OBJ meshes converted into deformable bodies using Isaac Lab's API. Primitive shapes like spheres and cubes were randomly placed behind the plant to mimic the variable fruit location. Domain randomization is applied to improve robustness, including 360° variations in plant orientation and perturbed lighting conditions. The PhysX engine operates at a 1/60 s timestep with eight solver iterations. A white wall is placed behind the fruits to minimize background distractions. All simulations are conducted in Isaac Lab 1.4.0 with an NVIDIA GeForce RTX 4090 GPU.

Although simplified, the simulated plant captures essential structural features: a central stem, occluding leaves, and a fruit target. This design encodes an inductive bias toward realistic manipulation strategies, encouraging the policy to discover affordances such as sliding along the stem or pushing leaves aside, which transfer naturally to real plants. Randomization of appearance, stiffness, and fruit placement prevents overfitting and ensures generalization.

C. Evaluation Metrics

To assess the RL agent's performance, we define the following evaluation metrics:

- Success rate: the percentage of trials in which at least 90% of the fruit is evident for more than 10 timesteps.
- Steps to goal: The number of steps required to achieve the visibility objective.
- Generalization: The percentage of successful trials across various configurations and environmental setups.

D. Training and Evaluation of the RL Agent

a) Training: We train the policy using the PPO [44] implementation from skrl [45] in simulation (please see Appendix for training configurations, the link to appendix is present in footer of page 1), enabling agent to learn continuous control of the robot's 6-DOF arm. Each episode begins with a randomized robot configuration and fruit location, requiring the agent to adapt to varying views and occlusion. To promote generalization, we apply domain randomization

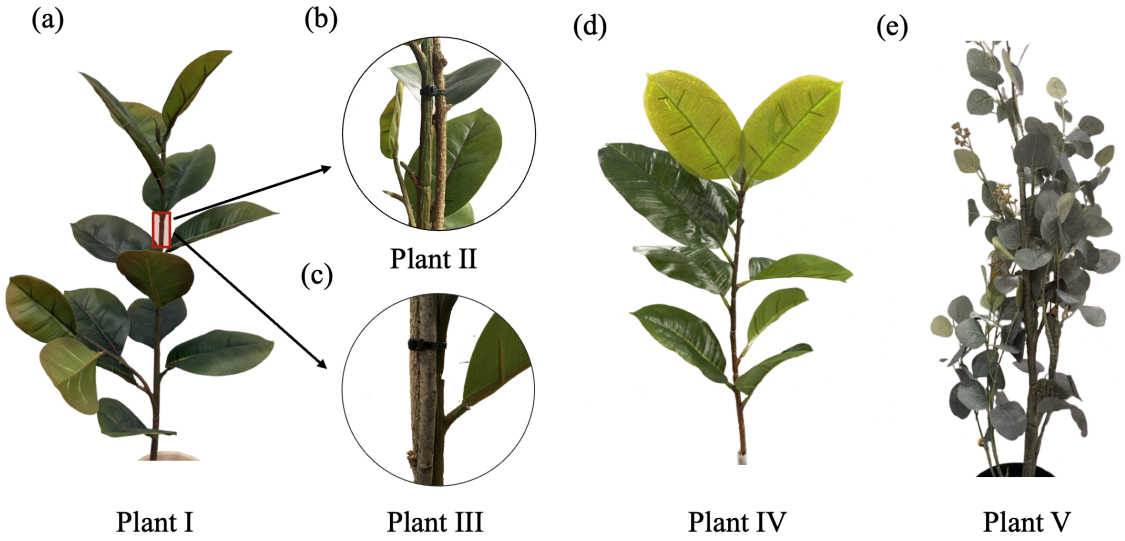

Fig. 3: Plant used for real-world experiments. **(a)** and **(d)** show plants with a clearly visible central stem and broad foliage, similar to the simulated morphology. **(b)** and **(c)** show reinforced versions of **(a)**: **(b)** is the doubled reinforced version of **(a)**, and **(c)** is the tripple reinforced version of **(a)**. The zoom-in highlights the reinforcement detail. **(e)** depicts a dense, bushy plant lacking a distinct central stem, presenting a more severe occlusion challenge. Together, these variations enable testing of the policy's robustness across a range of plant geometry and mechanical properties. All plants are artificial.

over physical properties (torque), visual conditions (lighting, textures), and sensor noise. These variations prevent overfitting to a specific environment and bolster robustness to real-world deployment.

To facilitate efficient learning, we provide the agent with privileged information during training: a ground-truth binary mask highlighting the fruit's location. This auxiliary input guides the agent toward identifying the fruit, especially when it is heavily occluded, and accelerates the acquisition of effective behaviors. Importantly, the mask is used only during training; at evaluation and deployment, the policy relies solely on sensory observations (see Section IV-C).

After approximately 50 million simulation steps, the agent converges to a policy that reliably finds and exposes fruit by strategically interacting with the plant: locate the stem and fruit, move the end-effector along the stem, and push aside occluding structure until the fruit becomes visible. This behavior, shaped by training in diverse randomized environments, generalizes well to real plants with similar stems and occlusion patterns.

b) Evaluation: During post-training evaluation in simulation, 20,000 randomized episodes mirroring training conditions, the RL agent achieved an overall success rate of **96.09%** (see Appendix's Fig. 1 for the plots). In most trials, the agent revealed the fruit with minimal occlusion, demonstrating the robustness and effectiveness of the learned policy. The agent also exhibited efficient behavior, requiring as few as 200 control steps in the easiest scenarios and up to 600 in the most challenging ones (mean episode length: $\approx$ 500 steps). This range indicates adaptability and consistency: even from unfavorable starting positions or highly occluded setups, the policy reliably finds a sequence of actions to achieve visibility.

E. Deployment Framework

The deployment framework (Fig. 2) integrates the trained RL controller with a low-level servo position controller. The end-effector pose and joint angles are published to ROS 2 and subscribed to by the control PC. Simultaneously, an RGB-D image is captured from the sensor. These observations are fed into the policy, which generates an action. The action, converted to target joint angles, is then published and received by the robot. At this stage, only kinematic control is used. The servo controller tracks the target position in a controlled manner without exerting excessive force on the plant. This process runs in a continuous loop, ensuring safe and effective control.

III. Experiments

After training exclusively in simulation, we deployed the policy zero-shot on a physical setup (Fig. 1 top) consisting of a 6-DoF MyBuddy arm, a RealSense camera [46], and a selection of potted, off-the-shelf artificial plants. Camera height, distance, and tilt as well as the plant's base pose-were fixed to mirror the simulation.

To evaluate generalization, we conducted experiments across a structured grid of initial conditions. Five plant instances, presented in Fig. 3, were considered: two morphologically similar (Plant I and IV), two structurally reinforced variant of one of the former two with increased stiffness (Plant II and Plant III), and a visually distinct, bushy plant with the stiffest stem (Plant V). Each plant exhibits unique stiffness characteristics, enabling evaluation of the policy's robustness across diverse physical and visual conditions. For each plant, the fruit was placed at nine discrete locations, arranged in a 3×3 lattice spanning left/center/right and low/mid/high positions, 20 cm behind the plant (Fig. 4(a)).

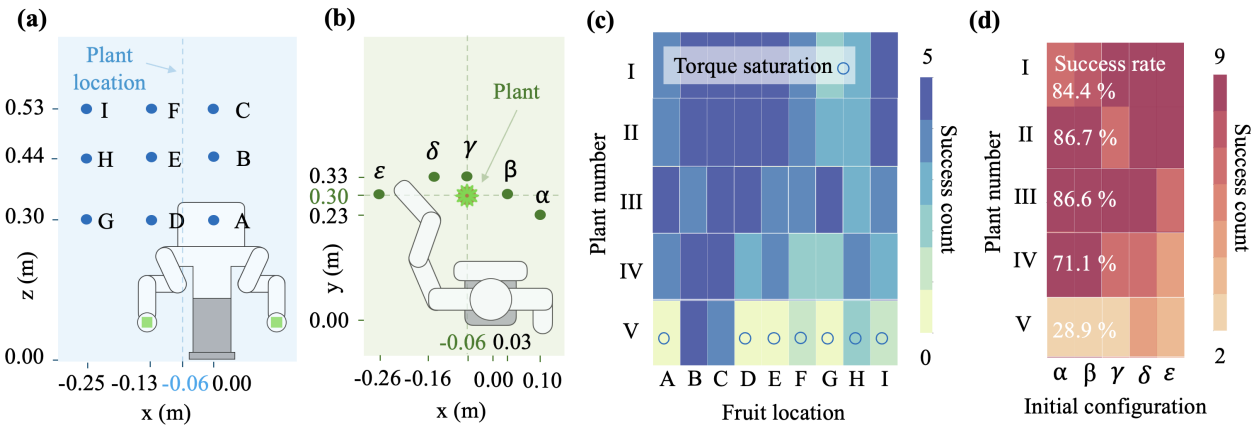

Fig. 4: **(a)** Experimental setup showing predefined fruit locations relative to the robot and the plant, viewed from the rear of the robot (x-z plane). **(b)** Top-down view (x-y plane) of the five different initial configurations, presented in end-effector positions. **(c)** Heatmap of successful trials across all plants (I-V) and fruit locations (A-I), where success count represents the number of successful attempts out of five trials for each initial configuration. **(d)** Heatmap of successful trials across all plant (I-V) and initial configurations (α-ε), where success count represents the number of successful attempts out of nine trials for each fruit location The percentage numbers at the lower left are the success rates across each plant.

The robotic arm was initialized from five pre-selected, self-collision-free joint configurations (Fig. 4(b)), resulting in $5 \times 9 \times 5 = 225$ unique trials. In each trial, the arm was reset to its designated pose, the fruit positioned at the target grid cell, and the trained policy was executed for up to 600 control steps at a frequency of 1 Hz, under low-level compliance control. A trial was considered successful if the robot achieved at least 90% fruit visibility for five consecutive frames. The number of control steps required to achieve success was also recorded.

IV. Results

This section presents results from deploying our learned policy on the real setup, focusing on observed behaviors, success rates, and failure modes. We first analyze generalization across plant morphologies, followed by evaluation in sequential multi-fruit scenarios and ablations on privileged information and simulation fidelity. Finally, we discuss key failure cases and limitations.

A. Policy Behavior and Generalization

In most trials, the robot exhibited purposeful and consistent behavior closely resembling what was observed in simulation. Based on our observations, the policy inferred likely fruit locations and anticipated occlusions. It then guided the end-effector toward occluded regions using compliant, stem-leveraging motions: the robot often maneuvered around the central stem, pushing it aside to expose hidden fruit. Once visibility exceeded the success threshold, the robot typically ceased further movement, demonstrating task-directed interaction.

As summarized in Fig. 4(c) and (d), the policy achieved high success rates on Plant I (84.4%), Plant II (86.7%), and Plant III (86.6%), Plant IV (71.1%) confirming strong generalization across different mechanical properties. Performance dropped on Plant V (28.9%), a bushy plant with no visible central stem and increased stiffness. The two key contributing factors for this drop are: (1) perception challenges due to the different morphology of the testing plant, and (2) increased stem stiffness, the dominant limitation, since the required motion exceeded the robot's torque capacity, leading to torque saturation. Our additional experiments corroborate this attribution (see Modified stiffness test in Section IV-D).

B. Multiple Fruit Generalization

Our policy is trained primarily in a single-fruit setup. Nevertheless, it could generalizes to multiple targets without architectural changes by following a *sequential exposure* protocol: once one fruit is uncovered, it is removed from the scene, and the same policy continues searching for the remaining fruit(s). This removal reflects a practical harvesting scenario in which one arm exposes the fruit while another arm can perform the picking task. Importantly, the control loop runs continuously, we do not reset the robot or environment after the first fruit is found. In scenarios where the first fruit is not to be harvested, e.g., unripe, though not implemented in our experiments, we could apply an ignore mask in the observation stream to prevent re-targeting while proceeding to the next fruit. The testing procedures are:

Setup: Two fruits were placed behind foliage in Plant I across two initial arm configurations (α and ϵ). The observation/action spaces were unchanged. Once a fruit reached the success criterion ($\geq 90\%$ visible for 10 frames), it was physically removed and the same policy continued operating without resetting the robot or environment.

Results: Across 18 trials, both fruits were exposed in 16 cases (88.9%). Failures occurred only in heavily occluded placements under the ϵ initialization, where torque limits prevented full exposure. On average, the first fruit required 40–400 steps, while the second required only 50–150 steps, due to the reduced occlusion.

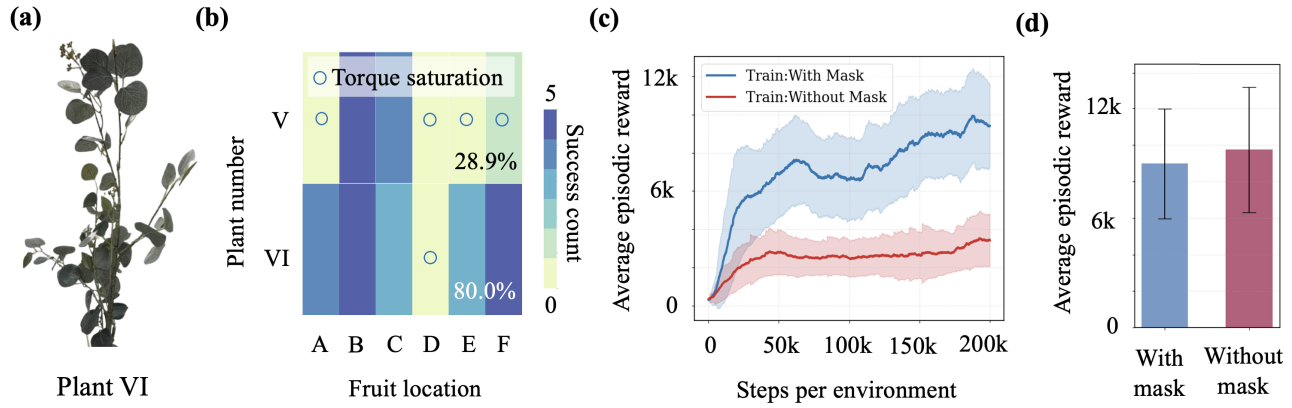

Fig. 5: **(a)** The modified version of Plant IV: mechanically more compliant while preserving its overall structure and visual complexity **(b)** Heatmap of successful trials between Plant IV and V and fruit locations (A-F), where success count represents the number of successful attempts out of five trials for each initial configuration. **(c)** Training performance with and without access to the ground-truth fruit mask. The mask accelerates learning and improves final returns **(d)** Evaluation results when deploying the policy trained with the mask: performance remains consistent even when the mask is removed at test time

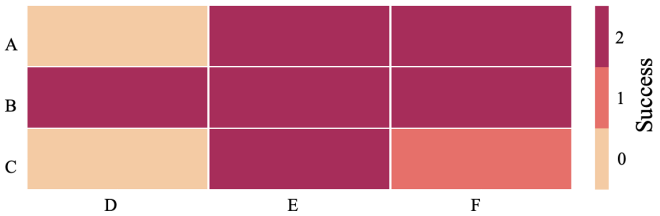

Fig. 6: Multi-fruit exposure success across different fruit location pairs. Rows indicate the first fruit location (A–C) and columns indicate the second (D–F). Each cell shows the number of fruits successfully uncovered (0–2) in sequential trials. The policy achieves high success for most pairs, with failures occurring primarily when both fruits are heavily occluded (pairs A–D and C–D).

In 18 two-fruit trials on Plant I, the policy successfully exposed both fruits in 16 cases (88.9%). Failures occurred only when both fruits were placed in heavily occluded positions (pairs D–A and D–C) under the ϵ initialization, where torque saturation prevented full exposure. In all other fruit pairs, success was achieved across both initial configurations. On average, the first fruit required 300–400 steps to expose, while the second fruit was exposed within 50–150 steps, since removal of the first target reduced occlusion for the next. These results confirm that the single-fruit policy extends naturally to sequential multi-fruit exposure without retraining. In practice, we observe the same stem-leveraging behavior as in the single-target case: the end-effector follows the stem, displaces occluding leaves, and halts once visibility is achieved. Removing the first fruit often relaxes local occlusions, reducing the steps needed for the second target. Failures typically arise from (1) re-occlusion due to plant relaxation after removal and (2) torque saturation on stiff stems, mirroring single-target limits. Please see Fig. 6 for full results of the multi-fruit tests.

C. Ablation Studies

To better understand the critical factors behind successful training and sim-to-real transfer, we conducted two ablations:

Privileged fruit mask during training: During training, we supplied a ground-truth fruit mask as an additional input channel. As shown in Fig. 5(c) and (d), policies trained with this privileged signal converged faster and reached higher returns. Importantly, removing the mask at test time had negligible effect, confirming that the mask accelerates learning but the final behavior relies only on RGB-D and proprioceptive inputs. This validates our training-time design choice.

Simulation fidelity: We trained two policies: one in a high-fidelity simulator with realistic textures and lighting, and another in a reduced-fidelity variant with flat textures and simplified illumination. The high-fidelity policy transferred successfully, achieving up to 86.7% real-world success. In contrast, the low-fidelity policy exhibited erratic behavior and failed to generalize, preventing real-world deployment. This demonstrates that while precise digital twins are unnecessary, sufficient visual realism in simulation is critical for effective sim-to-real transfer.

D. Failure Modes

Perception artifacts: We observed reduced performance on Plant II despite its similarity to Plant I. RealSense depth images occasionally produced ghosted or duplicated stems, leading to ambiguous depth cues and misinterpretation by the policy. These cases underscore the importance of sensor fidelity alongside policy robustness. Representative RGB-D frames are shown in Appendix's Figs. 7 - 10.

Torque saturation: The dominant failure mode on Plant IV was mechanical: the robot often saturated its torque limits, preventing execution of the commanded trajectory. Even in failed trials, partial exposure of 70–80% was achieved, just below the 90% success threshold.

Modified stiffness test: To test whether failures were due to mechanical resistance rather than policy limitations, we created Plant VI by softening Plant IV while preserving its visual complexity (Fig. 5(a)). On Plant VI, the policy achieved 80% success across valid fruit positions, com-

parable to Plants I–III and Plant V. Including a torque-limited fruit location (D) reduced the overall success to 67%. These findings confirm that the learned policy generalizes to visually distinct plants as long as actuation demands remain within hardware limits.

E. Summary of Findings

Overall, our results demonstrate that:

- The policy generalizes well across stemmed plants with varying stiffness, achieving up to 86.7% success in real trials.
- Failures arise primarily from mechanical resistance exceeding actuation limits and from sensor artifacts in depth perception.
- Privileged fruit masks accelerate training but are unnecessary at test time.
- High-fidelity finite element but abstract plant models suffice for sim-to-real transfer; exact digital twins are not required.

These results suggest that a simulated plant with a central stem and occluding leaves offers enough structural cues to support robust real-world deployment.

V. Conclusion and Future Work

We presented an end-to-end RL framework for occlusion-aware fruit discovery, trained entirely in simulation and deployed zero-shot to real plants. The approach combines a high-level kinematic policy with a compliant low-level controller, allowing the robot to uncover occluded fruits safely and adaptively while leveraging the central stem as a structural cue. Robust sim-to-real transfer is achieved through extensive domain randomization and the use of privileged signals during training, without requiring fine-tuning or very accurate digital twins. A key factor in this transfer is our high-fidelity finite element (FEM) plant model, which captures realistic contact dynamics without reduced-order approximations. By exposing the policy to nonlinear deformations of stems and leaves, the FEM model enables strategies that generalize across plant morphologies and multi-fruit scenarios, providing a critical bridge between abstract simulation and the complex behavior of real foliage.

In real-world trials, the learned policy achieved up to 86.7% success on stemmed plants, closely matching simulation performance (96.1%). Notably, our simulation relied on a generic morphological model rather than plant-specific reconstruction, demonstrating that abstract but structurally representative models are sufficient for transfer. These results mark a step toward scalable, perception-driven agricultural robots capable of reliable operation in cluttered, deformable environments.

Looking forward, we aim to expand the simulation framework to multi-branch canopies and fruit clusters, providing richer training scenarios. Beyond occlusion resolution, a key next step is adding a downstream fruit-picking task using the robot's second arm. Training in more realistic scenarios; such as fruits located deep inside foliage where both arms may need to coordinate to first clear occlusions and then grasp the target; represents an exciting future direction toward fully autonomous harvesting.

VI. Limitations

Our approach shows strong performance, but several limitations remain:

(a) Simplified simulation: Training relied on a single-stem, single-fruit abstraction. While adequate for simple plants, it does not capture complex morphologies like multi-branch canopies or fruit clusters. We also assumed static fruit state and a fixed overhead camera. Extending the simulator to relax these assumptions is a natural next step.

(b) Stem dependence: The policy assumes access to the central stem. In dense or constrained foliage this may not hold. Incorporating tactile feedback, contact-aware planning, or alternative strategies could improve robustness.

(c) Sim-to-real gap: Depth artifacts, lighting variation, and unmodeled dynamics (e.g., wind) still affect performance. Real-to-sim adaptation and more invariant visual encoders are promising remedies.

(d) Hardware limits: Failures on stiff plants highlight torque saturation. Stronger or variable-compliance actuators could expand applicability.

In summary, while precise digital twins are not required for effective sim-to-real transfer, addressing these limitations will be critical for scaling to field-deployable systems. Broader plant morphologies, richer occlusion scenarios, and enhanced perception robustness represent promising directions for enabling autonomous harvesters to operate reliably in diverse agricultural settings.

Acknowledgment

This work is jointly funded by the NSF–USDA COALESCE program (NSF award #1954556; USDA award #2021-6702134418)

References

[1] X. Song and K. Karydis, "GS-NBV: a Geometry-based, Semantics-aware Viewpoint Planning Algorithm for Avocado Harvesting under Occlusions," in *2025 IEEE 21st International Conference on Automation Science and Engineering (CASE)*, pp. 690–695, 2025.

[2] A. Abouzeid, M. Mansour, C. Hu, and D. Song, "A Point Cloud Completion Approach for the Grasping of Partially Occluded Objects and Its Applications in Robotic Strawberry Harvesting," *arXiv preprint arXiv:2506.14066*, 2025.

[3] A. Velasquez, C. Grimm, and J. R. Davidson, "Compact robotic gripper with tandem actuation for selective fruit harvesting," *arXiv preprint arXiv:2408.06674*, 2024.

[4] L. Li and H. Kasaei, "Enhanced View Planning for Robotic Harvesting: Tackling Occlusions with Imitation Learning," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13146–13152, 2025.

[5] J. Jun, J. Kim, J. Seol, J. Kim, and H. I. Son, "Towards an efficient tomato harvesting robot: 3d perception, manipulation, and end-effector," *IEEE access*, vol. 9, pp. 17631–17640, 2021.

[6] B. Arad, J. Balendonck, R. Barth, O. Ben-Shahar, Y. Edan, T. Hellström, J. Hemming, P. Kurtser, O. Ringdahl, T. Tielen, *et al.*, "Development of a sweet pepper harvesting robot," *Journal of field robotics*, vol. 37, no. 6, pp. 1027–1039, 2020.

[7] Z. Weng, P. Zhou, H. Yin, A. Kravberg, A. Varava, D. Navarro-Alarcon, and D. Kragic, "Interactive Perception for Deformable Object Manipulation," *IEEE Robotics and Automation Letters*, vol. 9, no. 9, pp. 7763–7770, 2024.

[8] P. Chu, Z. Li, K. Zhang, D. Chen, K. Lammers, and R. Lu, "O2RNet: Occluder-occludee relational network for robust apple detection in clustered orchard environments," *Smart Agricultural Technology*, vol. 5, p. 100284, 2023.

[9] S. Yao, S. Pan, M. Bennewitz, and K. Hauser, "Safe Leaf Manipulation for Accurate Shape and Pose Estimation of Occluded Fruits," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 16795–16802, 2025.

[10] H. Kang, H. Zhou, and C. Chen, "Visual Perception and Modeling for Autonomous Apple Harvesting," *IEEE Access*, vol. 8, p. 62151–62163, 2020.

[11] A. Navone, M. Martini, and M. Chiaberge, "Autonomous robotic pruning in orchards and vineyards: A review," *Smart Agricultural Technology*, vol. 12, p. 101283, 2025.

[12] S. Selvam, "Self-Supervised Learning for Robotic Leaf Manipulation: A Hybrid Geometric-Neural Approach," *arXiv preprint arXiv:2505.03702*, 2025.

[13] F. Gu, Y. Zhou, Z. Wang, S. Jiang, and B. He, "A survey on robotic manipulation of deformable objects: Recent advances, open challenges and new frontiers," *arXiv preprint arXiv:2312.10419*, 2023.

[14] J. Sanchez, J.-A. Corrales, B.-C. Bouzgarrou, and Y. Mezouar, "Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey," *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 688–716, 2018.

[15] C. Chi, B. Burchfiel, E. Cousineau, S. Feng, and S. Song, "Iterative residual policy: For goal-conditioned dynamic manipulation of deformable objects," *The International Journal of Robotics Research*, vol. 43, no. 4, pp. 389–404, 2024.

[16] A. Nair, D. Chen, P. Agrawal, P. Isola, P. Abbeel, J. Malik, and S. Levine, "Combining self-supervised learning and imitation for vision-based rope manipulation," in *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 2146–2153, IEEE, 2017.

[17] J. Luo, C. Xu, X. Geng, G. Feng, K. Fang, L. Tan, S. Schaal, and S. Levine, "Multistage Cable Routing Through Hierarchical Imitation Learning," *IEEE Transactions on Robotics*, vol. 40, pp. 1476–1491, 2024.

[18] H. Ha and S. Song, "FlingBot: The Unreasonable Effectiveness of Dynamic Manipulation for Cloth Unfolding," in *Proceedings of the 5th Conference on Robot Learning* (A. Faust, D. Hsu, and G. Neumann, eds.), vol. 164 of *Proceedings of Machine Learning Research*, pp. 24–33, PMLR, 08–11 Nov 2022.

[19] Z. Huang, X. Lin, and D. Held, "Mesh-based dynamics with occlusion reasoning for cloth manipulation," *arXiv preprint arXiv:2206.02881*, 2022.

[20] X. Lin, Y. Wang, J. Olkin, and D. Held, "Softgym: Benchmarking deep reinforcement learning for deformable object manipulation," in *Conference on Robot Learning*, pp. 432–448, PMLR, 2021.

[21] T. Boroushaki, J. Leng, I. Clester, A. Rodriguez, and F. Adib, "Robotic grasping of fully-occluded objects using rf perception," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 923–929, IEEE, 2021.

[22] P. M. Scheikl, N. Schreiber, C. Haas, N. Freymuth, G. Neumann, R. Lioutikov, and F. Mathis-Ullrich, "Movement Primitive Diffusion: Learning Gentle Robotic Manipulation of Deformable Objects," *IEEE Robotics and Automation Letters*, vol. 9, no. 6, pp. 5338–5345, 2024.

[23] S. Yao and K. Hauser, "Estimating Tactile Models of Heterogeneous Deformable Objects in Real Time," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 12583–12589, 2023.

[24] J. Jacob, S. Cai, P. V. K. Borges, T. Bandyopadhyay, and F. Ramos, "Gentle Manipulation of Tree Branches: A Contact-Aware Policy Learning Approach," in *8th Annual Conference on Robot Learning*, 2024.

[25] X. Zhang and S. Gupta, "Push past green: Learning to look behind plant foliage by moving it," *arXiv preprint arXiv:2307.03175*, 2023.

[26] C. H. Kim, M. Lee, O. Kroemer, and G. Kantor, "Towards Robotic Tree Manipulation: Leveraging Graph Representations," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11884–11890, 2024.

[27] J. Jacob, S. Cai, P. V. K. Borges, T. Bandyopadhyay, and F. Ramos, "Gentle Manipulation of Tree Branches: A Contact-Aware Policy Learning Approach," in *Proceedings of The 8th Conference on Robot Learning* (P. Agrawal, O. Kroemer, and W. Burgard, eds.), vol. 270 of *Proceedings of Machine Learning Research*, pp. 631–648, PMLR, 06–09 Nov 2025.

[28] M. Rijal, R. Shrestha, T. Smith, and Y. Gu, "Force Aware Branch Manipulation To Assist Agricultural Tasks," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1217–1222, 2025.

[29] C. Schuetz, J. Baur, J. Pfaff, T. Buschmann, and H. Ulbrich, "Evaluation of a direct optimization method for trajectory planning of a 9-dof redundant fruit-picking manipulator," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2660–2666, 2015.

[30] A. Tafuro, B. Debnath, A. M. Zanchettin, and E. Amir Ghalamzan, "dPMP-Deep Probabilistic Motion Planning: A use case in Strawberry Picking Robot," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8675–8681, 2022.

[31] T. Parhar, H. Baweja, M. Jenkins, and G. Kantor, "A deep learning-based stalk grasping pipeline," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6161–6167, IEEE, 2018.

[32] M. Jenkins and G. Kantor, "Online detection of occluded plant stalks for manipulation," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5162–5167, IEEE, 2017.

[33] L. Luo, H. Wen, Q. Lu, H. Huang, W. Chen, X. Zou, and C. Wang, "Collision-Free Path-Planning for Six-DOF Serial Harvesting Robot Based on Energy Optimal and Artificial Potential Field," *Complexity*, vol. 2018, no. 1, p. 3563846, 2018.

[34] J. Kwon, L. Yang, R. Nowak, and J. Hanna, "An Empirical Study on the Power of Future Prediction in Partially Observable Environments," *arXiv preprint arXiv:2402.07102*, 2024.

[35] R. T. Icarte, R. Valenzano, T. Q. Klassen, P. Christoffersen, A.-m. Farahmand, and S. A. McIlraith, "The act of remembering: A study in partially observable reinforcement learning," *arXiv preprint arXiv:2010.01753*, 2020.

[36] Q. Liu, A. Chung, C. Szepesvari, and C. Jin, "When Is Partially Observable Reinforcement Learning Not Scary?," in *Proceedings of Thirty Fifth Conference on Learning Theory* (P.-L. Loh and M. Raginsky, eds.), vol. 178 of *Proceedings of Machine Learning Research*, pp. 5175–5220, PMLR, 02–05 Jul 2022.

[37] A. Wagenmaker, K. Huang, L. Ke, K. Jamieson, and A. Gupta, "Overcoming the Sim-to-Real Gap: Leveraging Simulation to Learn to Explore for Real-World RL," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[38] H. I. Ugurlu, X. H. Pham, and E. Kayacan, "Sim-to-real deep reinforcement learning for safe end-to-end planning of aerial robots," *Robotics*, vol. 11, no. 5, p. 109, 2022.

[39] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: a survey," in *2020 IEEE symposium series on computational intelligence (SSCI)*, pp. 737–744, IEEE, 2020.

[40] H.-J. Yang, M. Khosravi, B. Walt, G. Krishnan, and S. Sarkar, "Zero-shot Sim-to-Real Transfer for Reinforcement Learning-based Visual Servoing of Soft Continuum Arms," in *Proceedings of the 7th Annual Learning for Dynamics & Control Conference* (N. Ozay, L. Balzano, D. Panagou, and A. Abate, eds.), vol. 283 of *Proceedings of Machine Learning Research*, pp. 1300–1312, PMLR, 04–06 Jun 2025.

[41] M. Mittal and et al., "Isaac lab: A gpu-accelerated simulation framework for multi-modal robot learning," *arXiv preprint arXiv:2511.04831*, 2025.

[42] M. Pozzi, E. Miguel, R. Deimel, M. Malvezzi, B. Bickel, O. Brock, and D. Prattichizzo, "Efficient fem-based simulation of soft robots modeled as kinematic chains," in *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 4206–4213, IEEE, 2018.

[43] D. F. Leguizamo, H.-J. Yang, X. Y. Lee, and S. Sarkar, "Deep Reinforcement Learning for Robotic Control with Multi-Fidelity Models," *IFAC-PapersOnLine*, vol. 55, no. 37, pp. 193–198, 2022. 2nd Modeling, Estimation and Control Conference MECC 2022.

[44] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[45] A. Serrano-Muñoz, D. Chrysostomou, S. Bøgh, and N. Arana-Arexolaleiba, "skrl: Modular and Flexible Library for Reinforcement Learning," *Journal of Machine Learning Research*, vol. 24, no. 254, pp. 1–9, 2023.

[46] L. Keselman, J. Iselin Woodfill, A. Grunnet-Jepsen, and A. Bhowmik, "Intel realsense stereoscopic depth cameras," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 1–10, 2017.