PanoVine: Whole-Body Visuomotor Control for Soft Growing Vine Robot

Yimeng Qin* **Xiaomeng Xu*** **William Heap** **Aditi Oak** **Shuran Song†** **Allison Okamura†**

Stanford University

https://panovine-bot.github.io

Abstract: Vine robots, a class of soft, growing robots, are suitable for navigating complex and confined environments due to their compliant bodies and self-supporting growth mechanism. However, hysteresis, tether interactions, and deformations make them difficult to predict and model, which in turn limits the effectiveness of conventional planning and control approaches. In this work, we present a data-driven, vision-based control framework for the first autonomous vine robot system. Our system integrates 19 cameras distributed along the robot's body to provide comprehensive feedback of both the robot state and the surrounding environment. Using this rich whole-body vision feedback, we train an end-to-end visuomotor policy from demonstrations for closed-loop autonomous control in complex environments. The policy efficiently aggregates information from distributed sensing while maintaining robustness to inaccurate robot states and actuation. Experimental results demonstrate that the learned policy enables robust navigation and manipulation in challenging scenarios, including steering through branched structures, climbing up slopes, traversing unsupported terrain, reaching objects precisely, and maneuvering through confined spaces and obstacles.

Keywords: Vine robot, soft robot, whole-body sensing, imitation learning

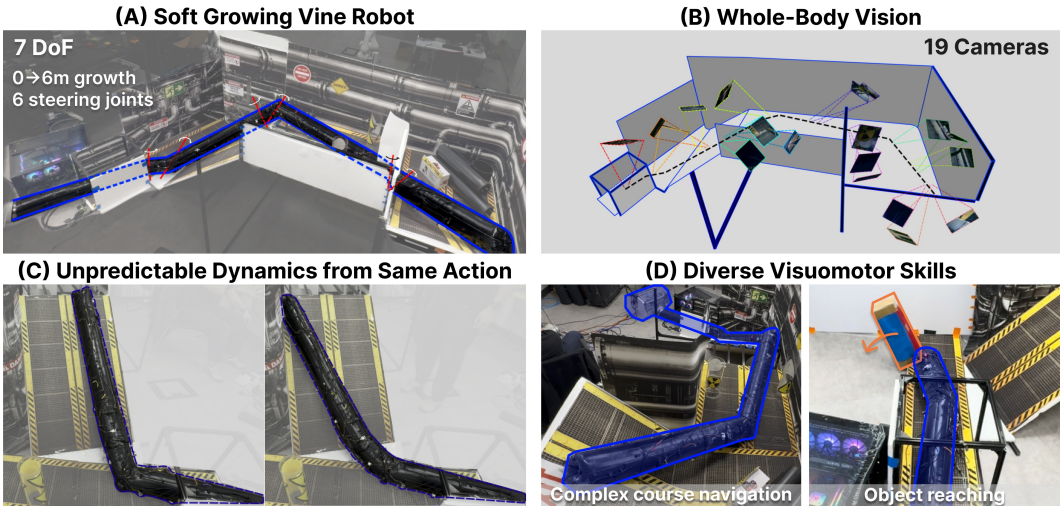

Figure 1: **PanoVine System** features (A) a 6 m soft growing vine robot with (B) 19 cameras distributed along the robot's body. (C) The system is challenging to control due to its unpredictable robot dynamics, where same action command can lead to drastically different robot configurations due to unpredictable buckling locations, soft-material hysteresis, and interactions with the environment. (D) The PanoVine System addresses the challenges and enables diverse navigation and manipulation skills by learning a whole-body visuomotor policy that leverages whole-body vision feedback.

*Equal Contributions
†Equal Advising

1 Introduction

Maintenance and inspection of industrial assets such as pipes, pressure vessels, and underground infrastructure remain challenging due to their confined geometry, limited communication access, and long deployment distances. Narrow passages, sharp bends, branches, and unsupported structures make these environments difficult to navigate for rigid robots that rely on wall-contact locomotion [1, 2]; limited communication access also makes drones that rely on GPS signals infeasible [3].

Vine robots, a class of pneumatic soft growing continuum robots, are well suited to these scenarios [4, 5]. They **grow** in length and extend at the tip by using internal fluid pressure to evert body material supplied from a fixed base. This growth mechanism allows vine robots to extend along tortuous paths to significant lengths, generate large propulsion forces while experiencing negligible sliding friction w.r.t. the environment, squeeze through apertures smaller than their body diameter, and have a lightweight self-supporting body capable of carrying sensors and trailing payloads.

The vine robot we employ can actively control the angle of multiple joints distributed throughout its body to **steer** towards desired directions or avoid obstacles, select branches, climb slopes, and pass through gaps, providing the dexterity needed to navigate confined, unstructured environments.

However, the same properties that make vine robots attractive also make it difficult to control their complex growing and steering motions for intricate navigation and manipulation:

- **Unpredictable robot dynamics.** The robot behavior is shaped by material compliance, hysteresis, buckling, body-environment interaction, and tension in the un-everted tail, all of which are difficult to predict and model. As shown in Fig. 1 C, open-loop replay of the exact action trajectory can lead to significantly different robot configurations across different trials [5].
- **Limited onboard sensing.** The robot undergoes intricate whole-body motions and deformations and spans a large and complex workspace (Fig. 1 A), making limited onboard sensing (e.g., IMU, single camera) insufficient for reliable control [6, 7].

To tackle these challenges, in this work, we propose *a whole-body vision and visuomotor policy learning system for controlling vine robots in complex environments.* To our knowledge, this is the first vine robot system capable of autonomous closed-loop control using on-board sensing alone.

To address unpredictable robot dynamics, we learn a visuomotor policy from demonstrations. High-dimensional visual feedback directly captures body deformation, contact state, and surrounding geometry, allowing a **whole-body visuomotor policy** trained end-to-end to compensate for uncertainty in robot states and dynamics. The policy is specifically designed to handle three challenges: aggregating information from distributed sensing, maintaining robustness to unreliable state estimation and actuation, and learning long-horizon behaviors where sparse steering actions are critical.

Such a policy requires rich sensory coverage across the entire robot body. Because the robot undergoes intricate whole-body motions over large workspaces, a single camera is insufficient to capture the observations a whole-body policy needs. We therefore equip the robot with a **whole-body vision** suite of 19 cameras attached along its body. The cameras are gradually revealed and exposed to the environment as the robot grows during eversion, and they collectively provide comprehensive feedback of both the robot and its environment (Fig. 1 B). When the robot is short, few cameras are revealed; more are revealed as it lengthens and gains degrees of freedom.

In summary, our contributions are:

- A vine robot platform with distributed whole-body vision to provide multi-perspective feedback of both the robot state and its environment.
- A learning-based visuomotor control framework for vine robots with design choices that improve robustness to perception and actuation uncertainty.
- Demonstration of challenging long-horizon navigation and manipulation tasks, including steering through branched structures, traversing unsupported terrain, maneuvering through cluttered obstacles, and reaching objects in the environment. Result videos are best viewed on https://panovine-bot.github.io.

2 Related Work

Control and Planning for Soft Growing Robots. Recent research in control and planning for soft growing robots, particularly vine robots, can be broadly categorized into model-based and data-driven approaches. Model-based methods typically rely on kinematic models, such as the piecewise constant curvature assumption [8, 9, 10, 11], to plan tip trajectories by controlling the robot's curvature. For instance, Greer et al. combined a kinematic model with visual servoing to regulate steering based on image-space error, and Watson et al. introduced a Jacobian-based adaptive control framework to enable deployment in contact-rich environments. Other works exploit obstacle contact for navigation [14, 15]. However, these methods often depend on prior knowledge of obstacle geometry or onboard sensing for real-time SLAM, which is difficult to provide in confined, cluttered environments — particularly on everting robots, where tip-mounted sensors add mechanical complexity and compromise reliability. The nonlinear, hysteretic behavior of soft materials further frustrates physics-based modeling: buckling and wrinkling during growth produce discontinuous deformations [9], and as the robot lengthens, accumulated tail tension can cause the body to buckle or deviate from the intended path in unsupported sections [16]. In contrast, PanoVine learns an end-to-end visuomotor policy from distributed whole-body vision that directly observes contact, deformation, and surrounding geometry, bypassing both the obstacle-geometry and body-modeling assumptions that limit model-based pipelines on long, tortuous deployments.

Learning-Based Methods for Soft Robots. To address the modeling limitations inherent in soft robotic systems, recent works have explored learning-based control framework. El-Hussieny and Hameed developed a deep reinforcement learning framework for obstacle-aware navigation, but the policy was only demonstrated in simulation. Kalibala et al. proposed a learning-based model predictive control framework trained on simulated trajectories, but this was also unverified in the real world. Jitosho et al. demonstrated reinforcement learning–based dynamic control on short vine-robot–like soft arms, but their reliance on finite-element discretization limits scalability to long, continuously growing robots. Haggerty et al. use a Koopman-based approach to learn an explicit dynamical model that enables model-based control; however, it relies on carefully chosen state representations and does not scale well to long, multi-segment systems. In contrast, we learn an end-to-end visuomotor policy directly from demonstrations, which bypasses the challenge of modeling system dynamics and planning high-DoF actions.

Whole-Body Sensing for Robotics. Prior work in whole-body sensing investigates distributed range, force, tactile, and vision sensing to enhance robot perception, addressing challenges in collision avoidance, contact detection, compliant motion, and whole-body manipulation. Range sensing enables semi-autonomous navigation for snake robots [21] and safe human–robot interaction [22]. Distributed force-torque sensors can enable compliant motions [23, 24, 25, 26, 27], while tactile sensing provides dense contact feedback [28, 29, 30, 31]. While effective for detecting nearby obstacles and physical contacts, these sensing modalities provide limited semantic understanding of the environment and task context. Vision offers richer geometric and semantic information beyond immediate contact. Recent works [32, 33, 34] employ egocentric and wrist cameras for bimanual mobile robots, but still suffer from occlusions and are limited to end-effector-only manipulation. RoboPanoptes [35] explores whole-body vision on a rigid serial robot by leveraging distributed visual sensing for whole-body manipulation. Our work applies the concept of whole-body vision to a soft growing vine robot, which enables continuous, multi-perspective observation of both the robot body and the surrounding environment throughout growth and steering.

3 Vine Robot System

3.1 Robot Design and Fabrication

Robot Design. The robot has *seven controllable degrees of freedom*, one from extension at the tip, and six from revolute joints. For the *tip extension*, similar to prior vine robot designs, the body of the vine robot used in this work consists of an airtight fabric tube flexible enough to be inverted

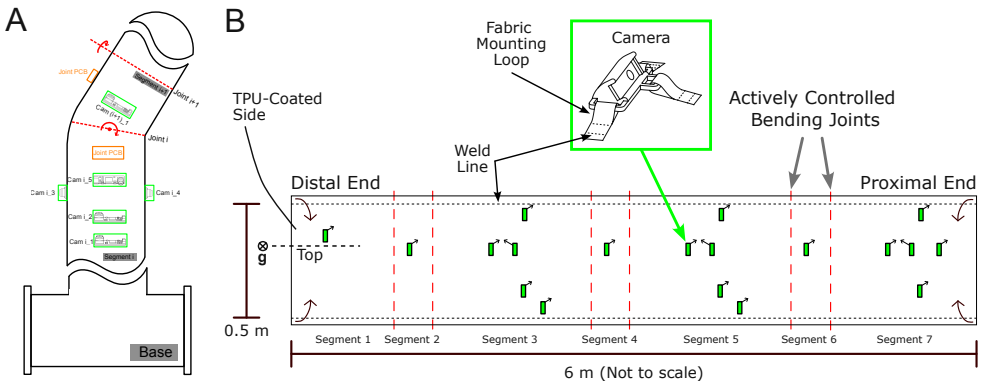

Figure 2: **PanoVine Robot Design.** (A) Scalable design of a single segment, showing the locations of the cameras and joints. (B) Placement of the 6 revolute joints and 19 RGB cameras distributed across the 7 segments of a 6 m long, 0.5 m diameter robot. Each joint is actively controlled to bend, changing the relative angle of adjacent segments. Cameras are attached to the TPU-coated side via welded fabric mounting loops.

into itself. When the inside of the tube is pressurized, the inverted material is everted at the tip of the robot body, and the robot body lengthens. We controlled the lengthening rate by spooling the inverted robot body material on motorized drum. For this work, we built a 6 m long cylindrical body 0.16 m in diameter. To enable robot *re-configuration and steering*, we attached 3 pairs of approximately revolute joints along the body of the robot. The joints within each pair are orthogonal to each other and the robot body's central axis, and parallel to a joint in the other pairs. Each joint is actively controlled in bending. Orthogonal joints within a pair are spaced 0.15 m apart, and each pair of joints is spaced 1 m apart. The design allows for high scalability and customizability for different environments by fabricating a longer body tube or adding more revolute joints.

Robot Fabrication. The primary material is 70-denier nylon ripstop fabric with one side coated with thermoplastic polyurethane (TPU). The vine robot body was fabricated by using an ultrasonic welder (Vetron 5064) to weld fabric attachment features for the actuators, local computing units, cameras, and cables to a fabric sheet.

Camera Placement. The motors, computing units, and cameras are housed in 3D-printed cases with tabs, and secured to the vine robot body with fabric loops. Nineteen cameras are mounted to the robot body at a 60 degree angle relative to the axis of the robot body (Fig. 2), facing either the distal or proximal end of the robot to ensure that a wide range of the environment is observed. One camera is mounted between the proximal end of the vine robot body and the first joint, and six are mounted after every joint pair. Six computing units are mounted before and after each joint pair.

3.2 Data Collection Interface

As shown in Fig. 3A, the operator teleoperates the multi-segment vine robot using a joystick (Logitech G F710). The active robot segment is switched using the RT and LT buttons. Joint motion of the selected segment is controlled via rate control on the right joystick, while the growth velocity is mapped to the left joystick. The robot body pressure, and thus stiffness, is adjusted using the X and A buttons to decrease and increase the value, respectively. Joystick commands are communicated through the Robot Operating System (ROS) at 30 Hz; joint angle and IMU data over RS-485 at 10 Hz; base-station motor encoder and torque measurements via USB at 50 Hz;

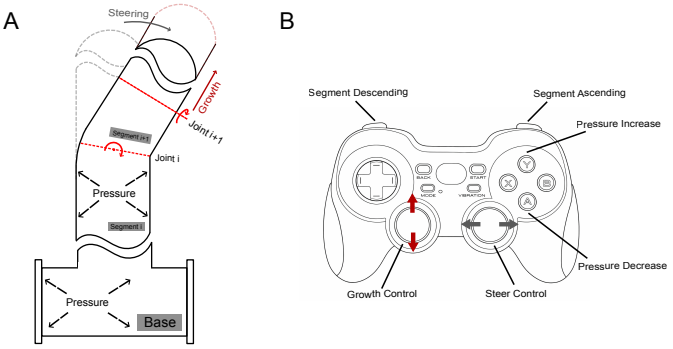

Figure 3: **Data Collection Interface.** A joystick controls the robot by independently commanding joint-space steering and axial growth.

and camera data via USB at 19.5 fps. We pre-characterize the relationships between joint angles and joint sensor readings, as well as between the robot odometer and deployed robot length, using calibrated mapping functions.

4 Whole-Body Visuomotor Policy

We learn a whole-body visuomotor policy from teleoperated demonstrations, as shown in Fig. 4. At each control step, the policy maps a history of sensory observations to an action chunk that specifies how the robot should grow and steer over the next several steps.

The policy receives observations $o = (\mathcal{I}, q)$, where $\mathcal{I}$ denotes the RGB streams from the 19 body-mounted cameras, and q denotes proprioception, comprising the joint angles and a base length signal derived from the growth spool encoder. The policy predicts an action chunk $a \in \mathbb{R}^{\tau \times d}$ consisting of the command base length and joint angles over a horizon τ ($\tau = 8$ in our experiments). Both observations and actions are sampled at 5 Hz. We adopt the diffusion transformers policy [36, 35] as our backbone, and introduce three design choices that target the specific control challenges of a long, deformable, whole-body-sensed vine robot: (i) aggregating distributed vision efficiently, (ii) remaining robust to unreliable proprioception and actuation, and (iii) preventing sparse but decisive steering events from being drowned out by long stretches of pure growth.

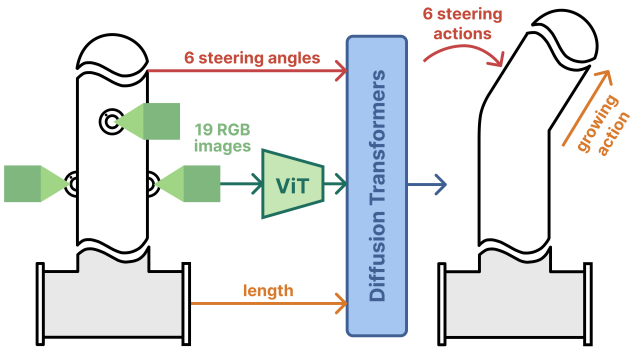

Figure 4: **PanoVine Whole-Body Visuomotor Policy.** The environment and robot states are observed through PanoVine's 19 cameras and growing and steering sensors. The 19 RGB images are represented by the class token of a vision foundation model. The vision tokens along with proprioception are taken by a diffusion transformers policy to predict growing and steering actions.

4.1 Learning Multi-View Correspondences with a Cross-Attention Transformers

Each body-mounted camera observes only a fragment of the full robot-and-environment state, and their relative poses change continuously as the body grows, buckles, and deforms under contact. Explicit extrinsic calibration is unreliable in this regime, so we instead let the network learn implicit cross-view correspondences. Specifically, we apply color jittering and random dropouts [35] on the images (each resized to $224 \times 224 \times 3$) and feed them into a pretrained CLIP Vision Transformers model [37, 38, 39], where we take the predicted class token for each image as a compact per-view embedding. Low-dimensional proprioceptive streams (length and steering angles) are concatenated with the vision tokens, and the diffusion transformers decoder cross-attends actions to this observation. This cross-attention learns task-specific semantic correspondences between multi-modal observations (multi-view images and proprioception) and robot actions.

4.2 Relative Proprioception and Action Representation for Robust Control

We express proprioception and the action chunk *relative to the latest observation frame* in the window: each channel is subtracted from its value at the current observation time. We also retain *absolute* channels for length and joints so the policy still receives unambiguous global context as inputs. At inference time, we subtract the most recent proprioceptive reference from the observation buffer before inference and convert predicted relative actions back to absolute commands by adding the same reference. This relative action representation ensures smooth transitions between action chunks and increases robustness to uncertainty in the robot's absolute state under actuation uncertainties and hysteresis.

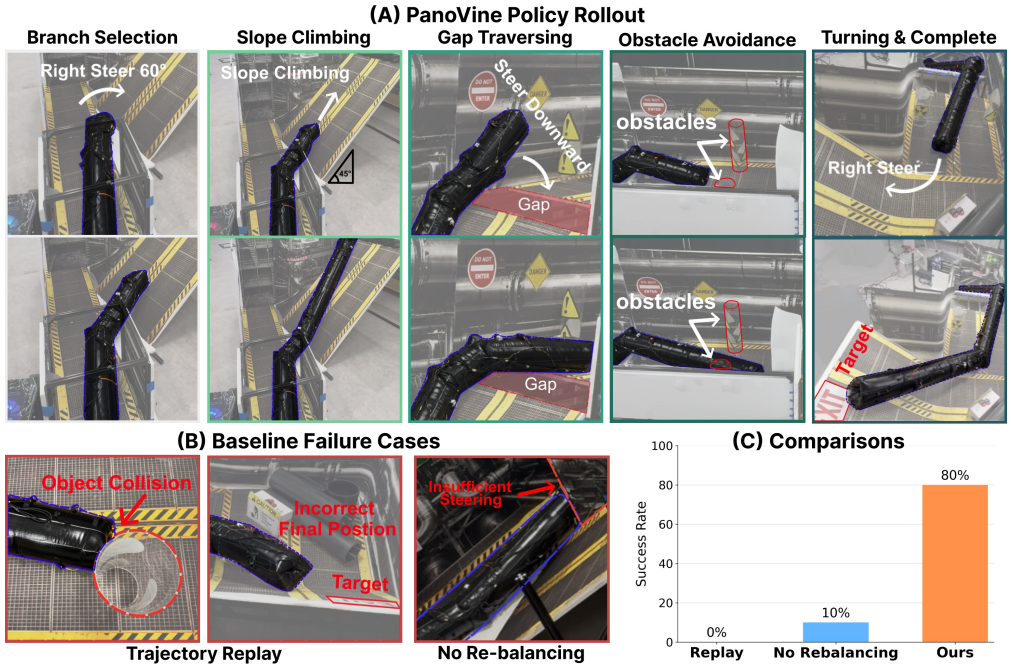

Figure 5: **Complex Course Navigation.** (A) Autonomous policy rollout of PanoVine, demonstrating long-horizon navigation skills in a complex environment, including steering through branched structures, climbing slopes, traversing unsupported gaps, avoiding obstacles, and making sharp turns. (B) Typical baseline failure cases. `Trajectory Replay` often collides with obstacles and fails to reach the correct final position. The `No Re-balancing` baseline undergoes insufficient steering. (C) Quantitative comparisons.

4.3 Rebalancing Steering and Growing Windows for Reactive Control

Teleoperated trajectories are dominated by long segments of nearly pure growth with zero joint angle changes, while steering motion is comparatively sparse. We label each training sequence window according to whether joint angles change over the action horizon relative to a threshold. Then we rebalance the training dataset towards the same ratio of steering windows and pure growth windows. This ensures the policy can learn critical and reactive steering actions for turning, climbing, bending etc. and avoid overfitting to growing actions.

5 Evaluation

We evaluate PanoVine on two challenging real-world tasks that stress different aspects of whole-body vision and visuomotor policy: long-horizon navigation through a complex course and precise reaching toward objects placed at unknown locations.

5.1 Complex Course Navigation

Task: As shown in Fig 5 (A), the robot needs to navigate through a complex course that spans 6 m long and 1.5 m tall. First, after growing for 0.6 m, the robot steers to the right for approximately 90° to enter a branch structure. Then, the robot climbs up a 45° slope, which is followed by an unsupported gap structure at the end, requiring it to lift up and then bend down to traverse the gap. Next, based on the placements of the obstacles in the environment, the robot needs to steer at precise angles to the right or left to pass through the obstacles without colliding. Finally, the robot turns at the correct moment at a narrow, sharp turn and climbs up another slope to reach the final course exit.

Capability: *Long-horizon control:* The course chains together five different skills—branch selection, slope climbing, unsupported gap traversal, obstacle avoidance, and sharp-turn maneuvering—over long continuous growth. Small steering errors early could compound and affect downstream subtasks, so the policy must sustain accurate whole-body actions across the entire trajectory.

Reactive steering from visual feedback: The robot's configuration at any given moment is only loosely predictable from the action history due to actuator force, material compliance, hysteresis, base buckling, and complex environmental interactions. The policy must therefore close the loop on its own body state at every step, using distributed visual feedback for reactive correction rather than committing to a precomputed trajectory. For example, in Fig. 6, the robot steers right when observing the branched structure, steers to avoid obstacles when they come into view, and stops growing after it observes the exit sign.

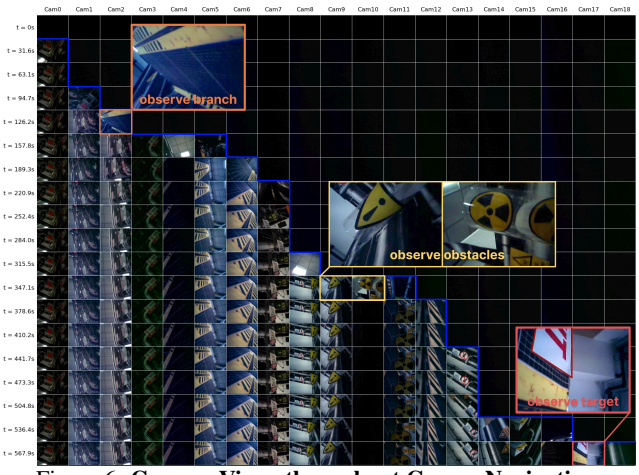

Figure 6: **Camera Views throughout Course Navigation.**

Data Collection: We collected 41 demonstrations with randomized obstacle placements and robot initial configurations (base buckling and initial length, initial orientation, and state of the joint actuators) with each demonstration taking an average duration of 12 minutes.

Test Scenarios: We ran 10 rollouts with randomized obstacle locations and robot initial configurations for all methods.

Performance: `Ours` whole-body visuomotor policy achieves an 80% success rate on the full course, demonstrating precise and reactive control from visual feedback that is robust to uncertainty in robot states and dynamics. Occasional failure cases include getting stuck in the unsupported gap when the bending angle is insufficient to clear it, and colliding with obstacles in the final narrow corner when the last turn is initiated with an insufficient turning angle.

`Trajectory Replay` baseline, where we replay the recorded action trajectory of a successful teleoperated demonstration open-loop on the same course, fails on every trial (0% success rate). Typical failure cases (Fig. 5 (B)) include colliding with obstacles whose positions differ from the demonstration, and falling short of the goal because base buckling reduces the deployed length below what the replayed growth commands assume. This confirms that the course is not solvable open-loop and that closed-loop visual feedback is required throughout the task.

`No Re-balancing` baseline, where we train the whole-body visuomotor policy on raw demonstrations without rebalancing steering and growing windows (Sec. 4.3), achieves only 10% success rate. As shown in Fig. 5 (B), it fails at the first branch selection phase with insufficient steering, demonstrating that our rebalancing strategy is critical for reactive and accurate steering control.

5.2 Object Reaching

Task: As shown in Fig. 8 (A), the robot needs to reach and touch objects 2 m away at different locations w.r.t. the robot. The robot first extends forward for a distance, then when seeing the object in view, it steers left or right towards the object to reach it precisely and knock it down.

Capability: *Precise steering:* Successful contact requires aligning the tip with the object to within a small angular tolerance after 2 m of growth, where even small joint-angle errors translate into large lateral offsets at the tip.

Generalization across objects and placements: Test objects include an object not seen during training and five distinct starting locations, so the policy cannot memorize a fixed trajectory. It must instead ground the steering command on the current visual appearance of the object across multiple cameras (Fig. 7), regardless of which specific object is present or where it is placed in the workspace.

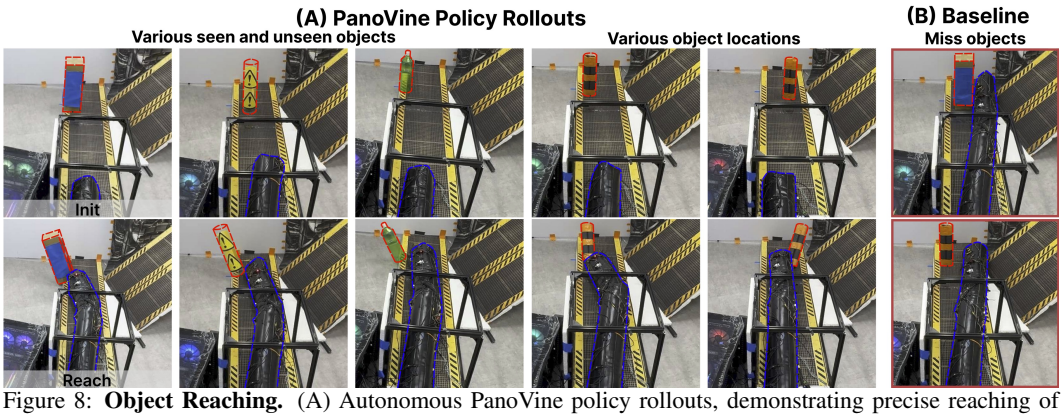

Figure 8: **Object Reaching.** (A) Autonomous PanoVine policy rollouts, demonstrating precise reaching of various objects placed at different locations, achieving **85%** success rate. (B) `Single Camera Policy` baseline always misses objects, resulting in a **0%** success rate.

Data Collection: We collected 80 demos on 4 objects with randomized locations. Average demo duration is 3 minutes.

Test Scenarios: We ran 20 rollouts on 4 objects (three seen and one unseen) starting from 5 different locations, and using the exact same test cases for all methods.

Performance: `Ours` achieves an 85% success rate, demonstrating precise visually-grounded steering under uncertainty in both target location and object geometry and appearance. The policy reactively adjusts its bending angle incrementally as the object becomes visible to successively more body cameras (Fig. 7), indicating that the policy successfully learned multi-view correspondences.

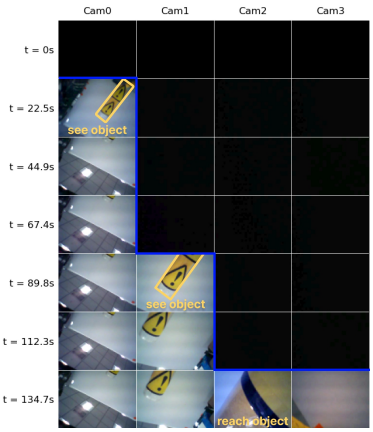

Figure 7: **Camera Views throughout Object Reaching.**

`Single Camera Policy` baseline, where we only input the first camera's image into the policy, achieves 0% success rate. The base camera cannot observe the object when the robot extends out and occludes it, or when the robot steers and the object goes out of view. The policy fails to turn and adjust towards the correct direction; thus, it always misses the object and grows past it (Fig. 8 (B)).

6 Limitations and Future Work

The current system only employs RGB cameras operating at a small field of view (FOV), and lacks contact and force signals; higher-FOV cameras and distributed force and tactile sensing would provide more information on environmental context. Each task is learned from tens of teleoperated demonstrations on a single embodiment; training with more data would make the policy more generalizable and robust. Finally, the robot and sensing designs are based on engineering experience, automatically optimizing them based on task-specific requirements would enhance performance [40, 41].

7 Conclusion

We presented PanoVine, a vine robot platform and learning framework that brings whole-body vision and visuomotor policy learning to long, deformable soft growing robots. The system combines a 6 m seven-DoF eversion-based body with 19 distributed cameras and a diffusion policy for whole-body visuomotor control. On real-world deployments the policy achieves 80% success on a six-meter complex course chaining branch selection, slope climbing, unsupported-gap traversal, obstacle avoidance, and sharp turns, and 85% success on precise object reaching, while single-camera and open-loop baselines fail completely. To our knowledge, this is the first system to demonstrate autonomous closed-loop control of a vine robot using only on-board sensing. We hope our system will encourage and facilitate future research on learning-based control for soft robots.

Acknowledgments

The authors would like to thank the CHARM Lab and REALab members for their helpful discussions and feedback on the manuscript. Xiaomeng Xu is supported by the Stanford Interdisciplinary Graduate Fellowship, and Yimeng Qin is supported by the Stanford Woods Institute for the Environment. This work was supported in part by NSF Awards #2143601, #2037101, and #2132519, an Amazon Research Gift, Stanford System-X, the Stanford Woods Institute for the Environment, and the Stanford University Sustainability Accelerator. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the sponsors.

References

[1] Y. G. Kim, D. H. Shin, J. I. Moon, and J. An. Design and Implementation of an Optimal In-pipe Navigation Mechanism for a Steel Pipe Cleaning Robot. *International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pages 772–773, 2011.

[2] T. Ren, Y. Zhang, Y. Li, Y. Chen, and Q. Liu. Driving mechanisms, motion, and mechanics of screw drive in-pipe robots: A review. *Applied Sciences*, 9(12), 2019.

[3] M. Tranzatto, T. Miki, M. Dharmadhikari, L. Bernreiter, M. Kulkarni, F. Mascarich, O. Andersson, S. Khattak, M. Hutter, R. Siegwart, and K. Alexis. Cerberus in the darpa subterranean challenge. *Science Robotics*, 7(66):eabp9742, 2022.

[4] E. W. Hawkes, L. H. Blumenschein, J. D. Greer, and A. M. Okamura. A soft robot that navigates its environment through growth. *Science Robotics*, 2(8):3028, 2017.

[5] L. H. Blumenschein, M. M. Coad, D. A. Haggerty, A. M. Okamura, and E. W. Hawkes. Design, modeling, control, and application of everting vine robots. *Frontiers in Robotics and AI*, 7: 548266, 2020.

[6] Y. Qin, J. Grinberg, W. Heap, and A. M. Okamura. 3d steering and localization in pipes and burrows using an externally steered soft growing robot. *arXiv*, 2025.

[7] L. Chen, Y. Gao, S. Wang, F. Fuentes, L. H. Blumenschein, and Z. Kingston. Physics-grounded differentiable simulation for soft growing robots. In *IEEE International Conference on Soft Robotics (RoboSoft)*, 2025.

[8] R. J. Webster III and B. A. Jones. Design and kinematic modeling of constant curvature continuum robots: A review. *The International Journal of Robotics Research*, 29(13):1661–1683, 2010.

[9] L. H. Blumenschein, A. M. Okamura, and E. W. Hawkes. Modeling of bioinspired apical extension in a soft robot. In *Living Machines*, 2017.

[10] L. H. Blumenschein, M. Koehler, N. S. Usevitch, E. W. Hawkes, C. D. Rucker, and A. M. Okamura. Geometric solutions for general actuator routing on inflated-beam soft growing robots. *IEEE Transactions on Robotics*, 38:1820–1840, 2020.

[11] A. Ataka, T. Abrar, F. Putzu, H. Godaba, and K. Althoefer. Model-based pose control of inflatable eversion robot with variable stiffness. *IEEE Robotics and Automation Letters*, 5(2): 3398–3405, 2020.

[12] J. D. Greer, T. K. Morimoto, A. M. Okamura, and E. W. Hawkes. A soft, steerable continuum robot that grows via tip extension. *Soft Robotics*, 6(1):95–108, 2019.

[13] C. Watson, R. Obregon, and T. K. Morimoto. Closed-loop position control for growing robots via online Jacobian corrections. *IEEE Robotics and Automation Letters*, 6(4):6820–6827, 2021.

[14] J. D. Greer, L. H. Blumenschein, R. Alterovitz, E. W. Hawkes, and A. M. Okamura. Robust navigation of a soft growing robot by exploiting contact with the environment. *The International Journal of Robotics Research*, 39(14):1724–1738, 2020.

[15] M. Selvaggio, L. A. Ramirez, N. D. Naclerio, B. Siciliano, and E. W. Hawkes. An obstacle-interaction planning method for navigation of actuated vine robots. *IEEE International Conference on Robotics and Automation (ICRA)*, 2020.

[16] M. M. Coad, R. P. Thomasson, L. H. Blumenschein, N. S. Usevitch, E. W. Hawkes, and A. M. Okamura. Retraction of soft growing robots without buckling. *IEEE Robotics and Automation Letters*, 5(2):2115–2122, 2020.

[17] H. El-Hussieny and I. A. Hameed. Obstacle-aware navigation of soft growing robots via deep reinforcement learning. *IEEE Access*, 12:38192–38201, 2024.

[18] A. Kalibala, A. A. Nada, H. Ishii, and H. El-Hussieny. Real-time force/position control of soft growing robots: A data-driven model predictive approach. *Nonlinear Engineering*, 14(1): 20250099, 2025.

[19] R. Jitosho, T. G. W. Lum, A. Okamura, and K. Liu. Reinforcement learning enables real-time planning and control of agile maneuvers for soft robot arms. In *Conference on Robot Learning*, 2023.

[20] D. A. Haggerty, M. J. Banks, E. Kamenar, A. B. Cao, P. C. Curtis, I. Mezić, and E. W. Hawkes. Control of soft robots with inertial dynamics. *Science Robotics*, 8(81):eadd6864, 2023.

[21] M. Tanaka, K. Kon, and K. Tanaka. Range-sensor-based semiautonomous whole-body collision avoidance of a snake robot. *IEEE Transactions on Control Systems Technology*, 23(5): 1927–1934, 2015.

[22] K. Qi, Z. Song, and J. S. Dai. Safe physical human-robot interaction: A quasi whole-body sensing method based on novel laser-ranging sensor ring pairs. *Robotics and Computer-Integrated Manufacturing*, 75:102280, 2022.

[23] M. Kollmitz, D. Büscher, T. Schubert, and W. Burgard. Whole-body sensory concept for compliant mobile robots. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5429–5435. IEEE, 2018.

[24] A. Goncalves, N. Kuppuswamy, A. Beaulieu, A. Uttamchandani, K. M. Tsui, and A. Alspach. Punyo-1: Soft tactile-sensing upper-body robot for large object manipulation and physical human interaction. In *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*, pages 844–851. IEEE, 2022.

[25] M. Murooka, T. Hoshi, K. Fukumitsu, S. Masuda, M. Hamze, T. Sasaki, M. Morisawa, and E. Yoshida. Tact: Humanoid whole-body contact manipulation through deep imitation learning with tactile modality. *IEEE Robotics and Automation Letters*, 2025.

[26] H. Choi, Y. Hou, C. Pan, S. Hong, A. Patel, X. Xu, M. R. Cutkosky, and S. Song. In-the-wild compliant manipulation with umi-ft. *arXiv preprint arXiv:2601.09988*, 2026.

[27] X. Xu, Y. Hou, Z. Liu, and S. Song. Compliant residual dagger: Improving real-world contact-rich manipulation with human corrections. *Advances in Neural Information Processing Systems*, 38:139559–139581, 2026.

[28] R. S. Dahiya, G. Metta, M. Valle, and G. Sandini. Tactile sensing—from humans to humanoids. *IEEE transactions on robotics*, 26(1):1–20, 2009.

[29] Y. Liu, X. Xu, W. Chen, H. Yuan, H. Wang, J. Xu, R. Chen, and L. Yi. Enhancing generalizable 6d pose tracking of an in-hand object with tactile sensing. *IEEE Robotics and Automation Letters*, 2023.

[30] E. Dean-Leon, J. R. Guadarrama-Olvera, F. Bergner, and G. Cheng. Whole-body active compliance control for humanoid robots with robot skin. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 5404–5410. IEEE, 2019.

[31] S. Jiang and L. L. Wong. A Hierarchical Framework for Robot Safety using Whole-body Tactile Sensors. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8021–8028. IEEE, 2024.

[32] X. Xu, J. Park, H. Zhang, E. Cousineau, A. Bhat, J. Barreiros, D. Wang, and S. Song. Hommi: Learning whole-body mobile manipulation from human demonstrations. *arXiv preprint arXiv:2603.03243*, 2026.

[33] R. Punamiya, S. Kareer, Z. Liu, J. Citron, R.-Z. Qiu, X. Cai, A. Gavryushin, J. Chen, D. Liconti, L. Y. Zhu, et al. Egoverse: An egocentric human dataset for robot learning from around the world. *arXiv preprint arXiv:2604.07607*, 2026.

[34] H. Xiong, X. Xu, J. Wu, Y. Hou, J. Bohg, and S. Song. Vision in action: Learning active perception from human demonstrations. *arXiv preprint arXiv:2506.15666*, 2025.

[35] X. Xu, D. Bauer, and S. Song. RoboPanoptes: The All-Seeing Robot with Whole-body Dexterity. In *Proceedings of Robotics: Science and Systems*, 2025.

[36] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023.

[37] A. Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[38] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[39] X. Xu, Y. Yang, K. Mo, B. Pan, L. Yi, and L. Guibas. Jacobinerf: Nerf shaping with mutual information gradients. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16498–16507, 2023.

[40] X. Xu, H. Ha, and S. Song. Dynamics-guided diffusion model for sensor-less robot manipulator design. In *Conference on Robot Learning*, pages 4446–4462. PMLR, 2025.

[41] S. Yi, X. Bai, A. Singh, J. Ye, M. T. Tolley, and X. Wang. Co-design of soft gripper with neural physics. *arXiv preprint arXiv:2505.20404*, 2025.

[42] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[43] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Appendix

1 Sensing and Electronics

The sensing and electronics system is divided into on-body components and base components. The on-body components comprise 19 USB webcams (GC0307 sensors), 12 magnetic rotary encoders (Pololu), and six custom local computing units. Each local computing unit integrates a Teensy 4.0 microcontroller (MCU), dual motor drivers (DRV8838), and an onboard power management system.

At the base, a pressure regulator (QB3, Proportion-Air) is used to control the internal body pressure, and a high-torque brushless DC motor (CubeMars AK80-9) drives a motorized spool to regulate the robot growth velocity. A PCIe-to-USB interface board supports the high-bandwidth USB camera system, while a lead MCU mediates communication between the base computer and the distributed local computing units. Full-duplex RS-485 communication is employed to enable bidirectional data exchange. In the forward direction, the lead MCU transmits motor control commands to individual local MCUs, specifying pulse-width modulation (PWM) duty cycle and pulse duration. In the reverse direction, each local MCU acquires motor encoder measurements and IMU data via analog and I^2C interfaces, respectively, and transmits the sensor data back to the base. To prevent data collisions on the shared RS-485 bus, a circular token-passing protocol is implemented, allowing only one local MCU to transmit at any given time. All USB cameras stream visual data via USB 2.0 at a resolution of 320 × 240 pixels and 30 fps, with the base computer interfacing through the PCIe expansion board. Notably, the maximum robot length of 4.2 m remains within the practical USB 2.0 communication distance (< 5 m), ensuring reliable data transmission.

2 Policy Training Details

Observations and actions. We use a short observation history of T_o=2 steps and predict an action horizon of T_p=8 steps at 5 Hz (downsampled by a factor of 3 from 15 Hz demonstrations). Observations include 19 RGB camera streams ($3\times224\times224$ each) and proprioception. Proprioception comprises the robot's 6 segment joint angles and the base extension length, each provided in both an absolute and a latest-frame-relative form. Actions are 7-dimensional, consisting of the 1-dimensional base extension length and the 6 segment joint angles ($7 = 1 + 6$), predicted in the relative representation.

Model. We use Diffusion Policy with a Diffusion Transformer backbone [36]. The diffusion model is conditioned on global observation embeddings and predicts noise (ε-prediction) with a DDIM scheduler [42] using a squared-cosine β schedule. We use 50 training timesteps, 16 inference steps, and input perturbation 0.1. The DiT uses embedding dimension 768, depth 7, 8 heads, and attention dropout 0.1. The observation encoder finetunes a CLIP-pretrained `ViT-B/16` backbone [37] (`vit_base_patch16_clip_224.openai`) shared across all camera streams, aggregating the `CLS` token, with ColorJitter augmentation. We maintain an exponential moving average (EMA) of the weights (power 0.75, max decay 0.9999).

Optimization. We use AdamW [43] with a cosine learning rate schedule (2000 warmup steps) starting at a learning rate of 3×10^{-4} for the diffusion model and 3×10^{-5} for finetuning the vision backbone, weight decay 1×10^{-6}, and betas $(0.95, 0.999)$. We use a batch size of 32 and train our policy and all baselines for 500 epochs.

3 Additional Camera Visualization

Fig. 9 shows visualization of camera views on the complex course navigation task.

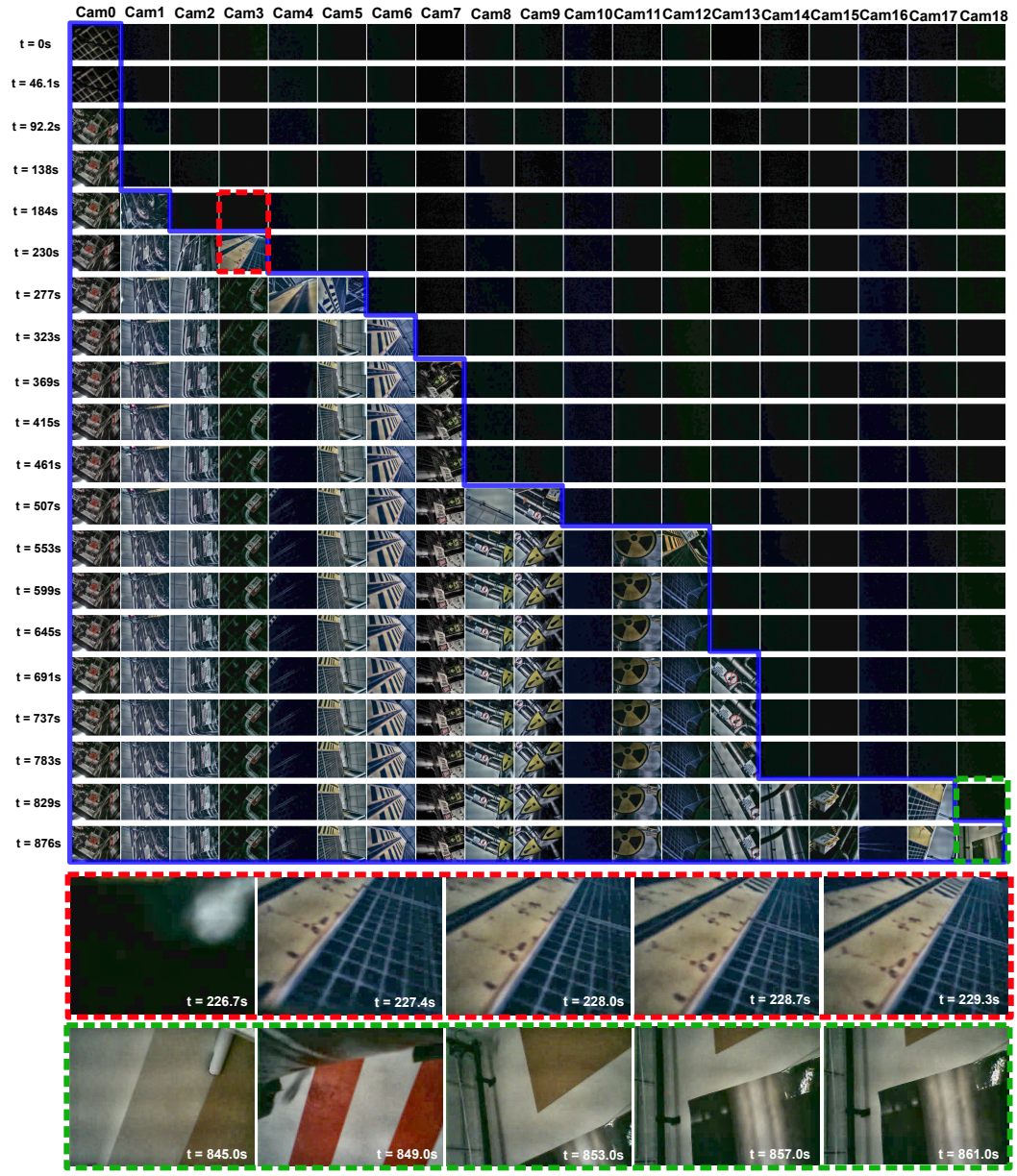

Figure 9: Camera-view visualization time series from 19 cameras across 20 time instances. Two example camera views during eversion are highlighted with red and green dashed outlines.