



Universität Hamburg

DER FORSCHUNG | DER LEHRE | DER BILDUNG

MIN Faculty
Department of Informatics



Data Augmentation in Offline Reinforcement Learning for Robotic Dough Shaping

Master's Thesis Presentation

Fabian Wiczorek



University of Hamburg
Faculty of Mathematics, Informatics and Natural Sciences
Department of Informatics

Technical Aspects of Multimodal Systems

16. January 2024



Table of Contents

Introduction

Related Work

Method

Planned Evaluation

Conclusion

References

Introduction

Related Work

Method

- Goal and Setup

- Data Augmentation

- Training

Planned Evaluation

Conclusion



- ▶ Object Manipulation important for task solving
 - ▶ Rigid objects
 - ▶ Deformable objects



The YCB object dataset[Çal+15].

- ▶ Object Manipulation important for task solving
 - ▶ Rigid objects
 - ▶ Deformable objects



Liquid



Fabric



Elasto-plastic



Mixture

Deformable Objects supported by DaxBench[Che+22].

- ▶ Rope manipulation, articulated objects
- ▶ Focus on elasto-plastic material (dough)

Applications

Introduction

Related Work

Method

Planned Evaluation

Conclusion

References





Table of Contents

Introduction

Related Work

Method

Planned Evaluation

Conclusion

References

Introduction

Related Work

Method

- Goal and Setup

- Data Augmentation

- Training

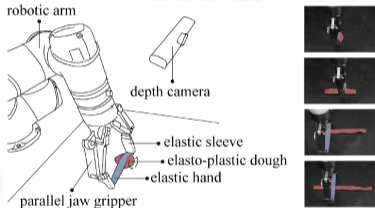
Planned Evaluation

Conclusion



Deformable Elasto-Plastic Object Shaping using an Elastic Hand and Model-Based Reinforcement Learning[MB21]

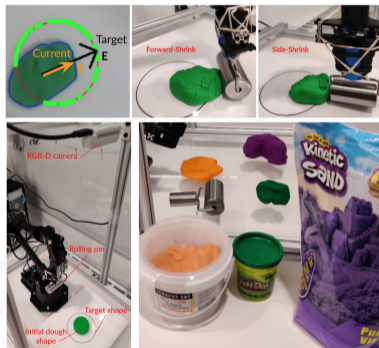
- ▶ Rolling dough with elastic end effector
- ▶ Bounding box from point cloud
- ▶ Dynamics model from random exploration



Results: RL > heuristic sampling > random sampling

Robotic Dough Shaping [Ond+22]

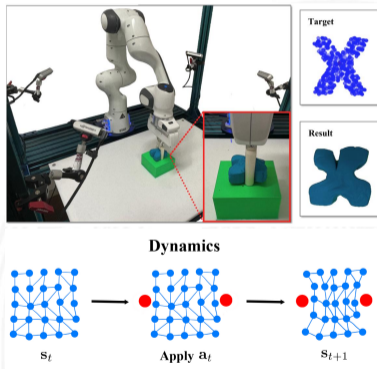
- ▶ Playdoh, Plasticine, Kinetic sand; WidowX-250
- ▶ Flatten into target circle
- ▶ Heuristic methods only
 - ▶ Highest point to edge
 - ▶ Center to missing area
 - ▶ Shrink when dough overlaps
 - ▶ etc.



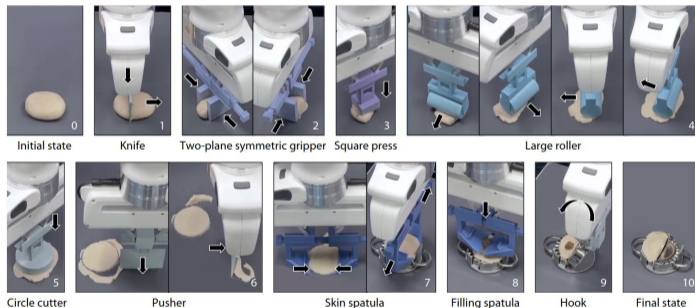
RoboCraft: Learning to See, Simulate, and Shape Elasto-Plastic Objects with Graph Networks[Shi+22]

- ▶ Two finger dough shaping into alphabet letters
- ▶ Perceive dough shape via point cloud
- ▶ Train graph NN as dynamics model (6000 samples)

Gradient Descent > Human > RL

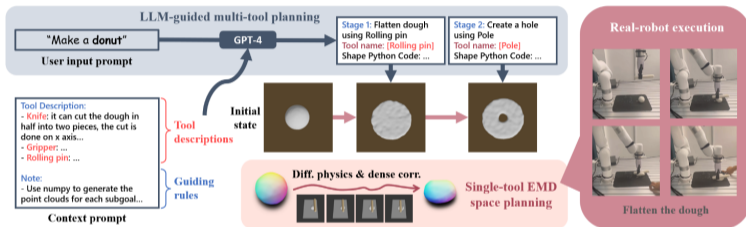


RoboCook: Long-Horizon Elasto-Plastic Object Manipulation with Diverse Tools[Shi+23]



- ▶ 15 tools - 15 GNNs
- ▶ PointNet predicts best tool (self-supervised classifier)
- ▶ Policy uses state + tool
- ▶ Robustly making dumplings and alphabet letters

Make a Donut: Language-Guided Hierarchical EMD-Space Planning for Zero-shot Deformable Object Manipulation[You+23]



- ▶ LLM plans by outputting
 - ▶ Tool
 - ▶ Point cloud code
- ▶ Diff. physics to shape dough using predicted tool
- ▶ Compares against GD, PASTA, BC, SAC
- ▶ 5 attempts at making a donut, baguette, 2 pancakes



Table of Contents

Introduction

Related Work

Method

Planned Evaluation

Conclusion

References

Introduction

Related Work

Method

Goal and Setup

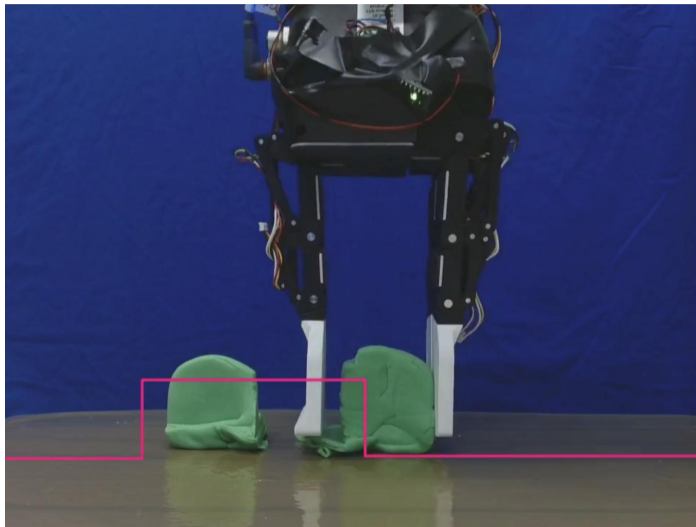
Data Augmentation

Training

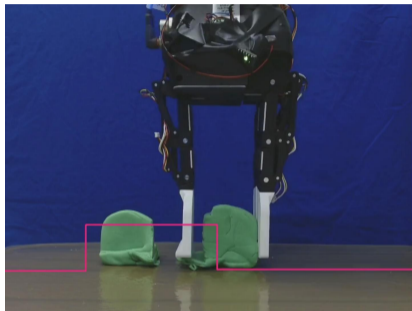
Planned Evaluation

Conclusion



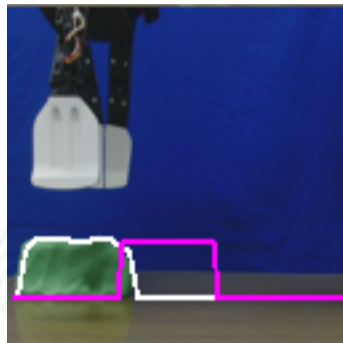


- ▶ Dough shaping on real robot
- ▶ Only pixel inputs
- ▶ Leverage existing Offline RL methods

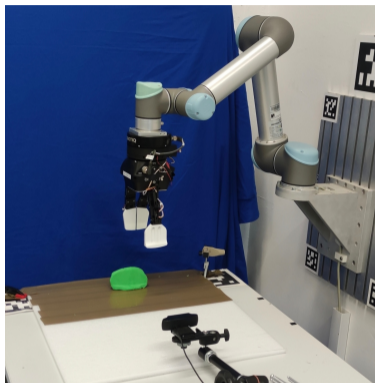
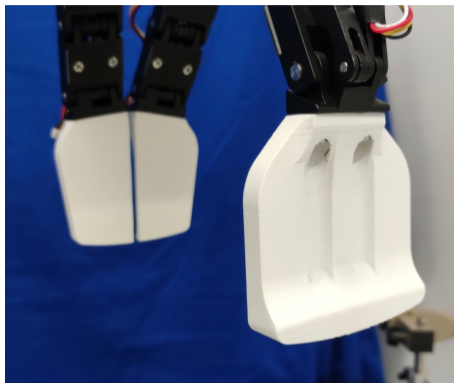


Red line indicates the target shape.

- ▶ UR5, Robotiq 3-finger gripper
- ▶ Input: RGB camera (128x128x3)
- ▶ Action: 4D vector
 - ▶ relative y translation
 - ▶ relative z translation
 - ▶ relative finger distance
 - ▶ relative rotation along z-axis (-90°, 0°, 90°)
- ▶ dough is not fixed



Example input image. The magenta line indicates the target shape. The white line indicates the detected shape.



- ▶ 3D printed fingers, Taped surface
 - ▶ Prevent dough from sticking to robot
- ▶ Unicolored background for less visual disturbances

Method Overview

Introduction

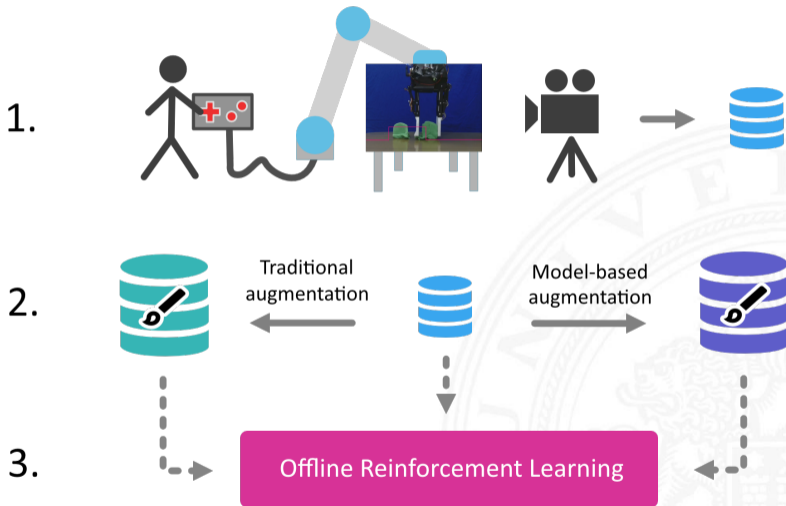
Related Work

Method

Planned Evaluation

Conclusion

References





Data Collection

Introduction

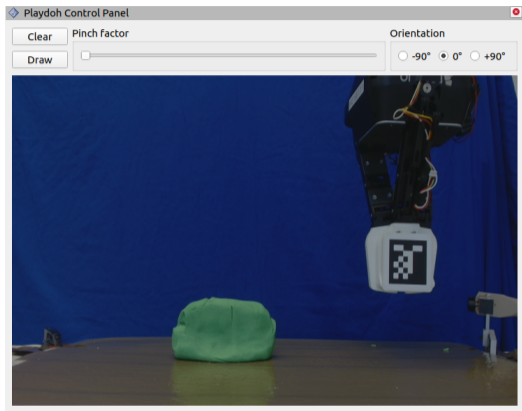
Related Work

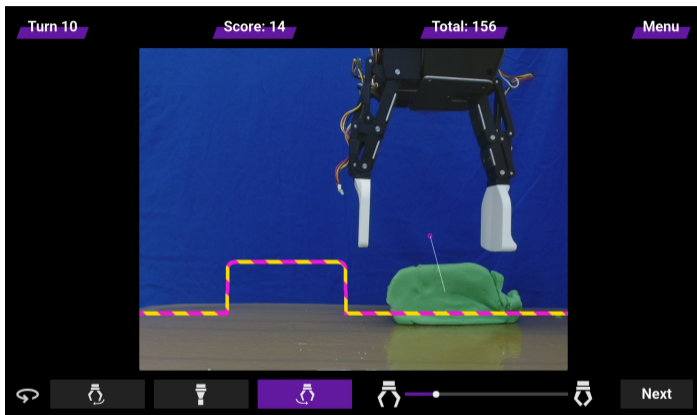
Method

Planned Evaluation

Conclusion

References

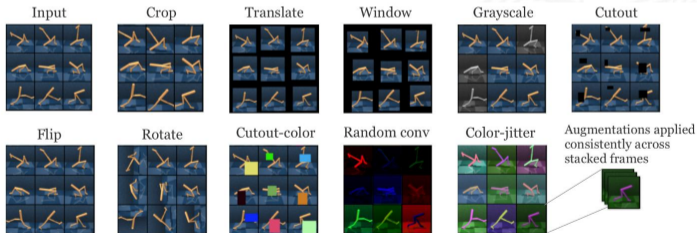




- ▶ Gamification approach
- ▶ Currently 2400 samples - desired 10000 samples

Data Augmentation

- ▶ Data Augmentation in RL shown to be effective[KYF20]
- ▶ Traditional Augmentations
 - ▶ Random Resize Crop, Color Jitter, Noise, Flip, Rotate, ...
- ▶ Model-Based Augmentations
 - ▶ Modify actions of a trajectory



Examples of traditional augmentations[Las+20].

Use augmentations to increase buffer size by factor 10

Data Augmentation - Traditional

Introduction

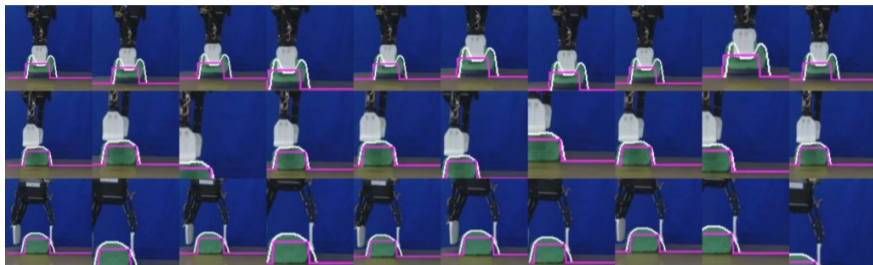
Related Work

Method

Planned Evaluation

Conclusion

References

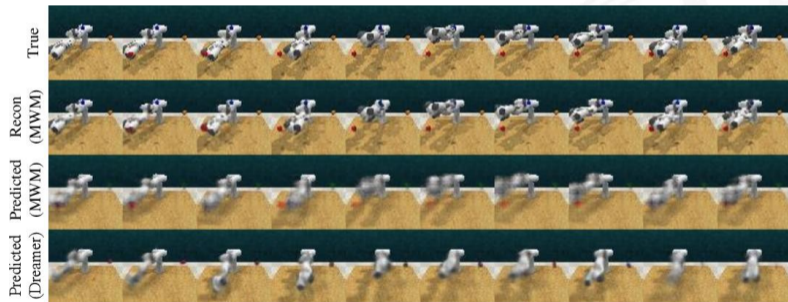


- ▶ Random crop with scaled actions[Las+20]

Model-Based Augmentation

Learned model for complex augmentations

- ▶ Model based RL often online
- ▶ Often employ autoencoder to construct latent space
- ▶ Image quality not primary interest



From 'Masked World Models for Visual Control'[Seo+22].

Model-Based Augmentation - Dynamics Model

Introduction

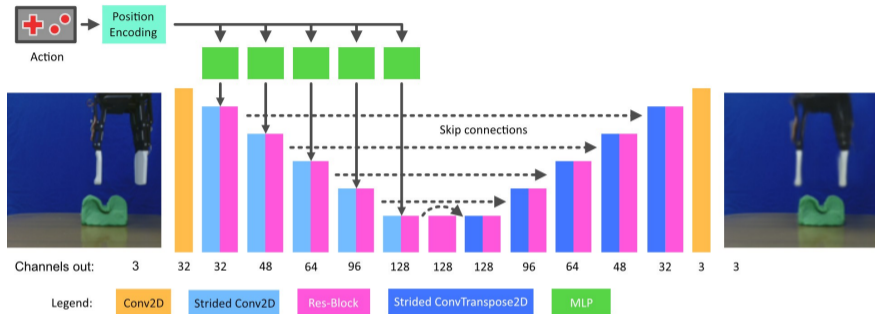
Related Work

Method

Planned Evaluation

Conclusion

References



- ▶ Conditional U-Net (2 Mio parameters)
- ▶ Used in *Model-Based Reinforcement Learning for Atari* [Kai+19]
- ▶ Trained using collected data + traditional augmentations

Model-Based Augmentation - Dynamics Model

Introduction

Related Work

Method

Planned Evaluation

Conclusion

References

Dynamics only



Problem

Image error accumulates over subsequent steps

- ▶ Approach: diffusion

Diffusion Model for Reprojection

Introduction

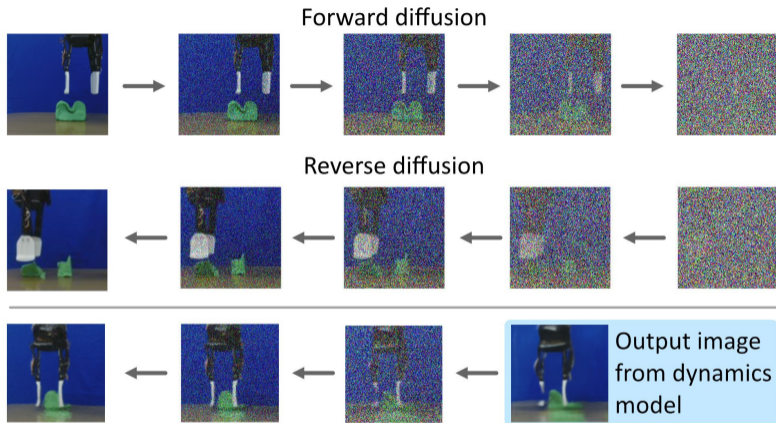
Related Work

Method

Planned Evaluation

Conclusion

References



- ▶ Use unconditioned diffusion model to reproject predicted images
- ▶ Trained using collected data + traditional augmentations

Model-Based Augmentation - Comparison

Introduction

Related Work

Method

Planned Evaluation

Conclusion

References

Dynamics only



Dynamics and diffusion



Model-Based Augmentation - Video

Introduction

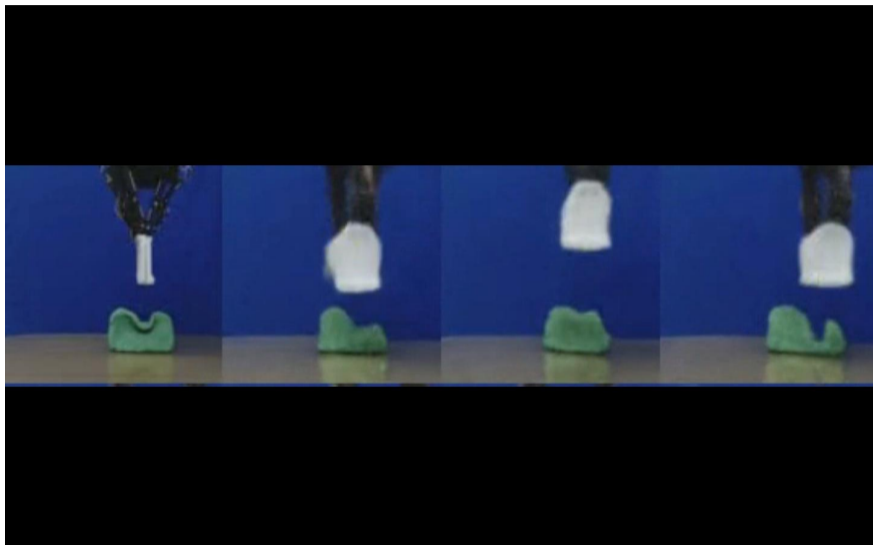
Related Work

Method

Planned Evaluation

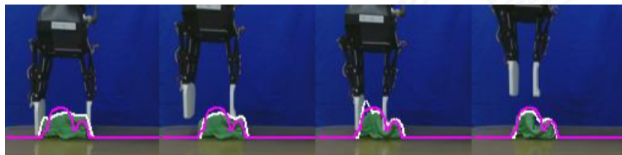
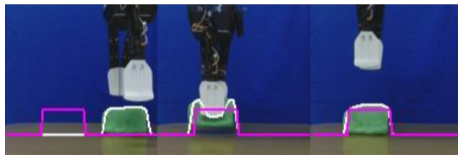
Conclusion

References





- ▶ Currently: Improvement of shape loss over trajectory



Examples that increase the reward.

- ▶ d3rlpy benchmark framework[SI21]
- ▶ Implicit Q-Learning[KNL21]

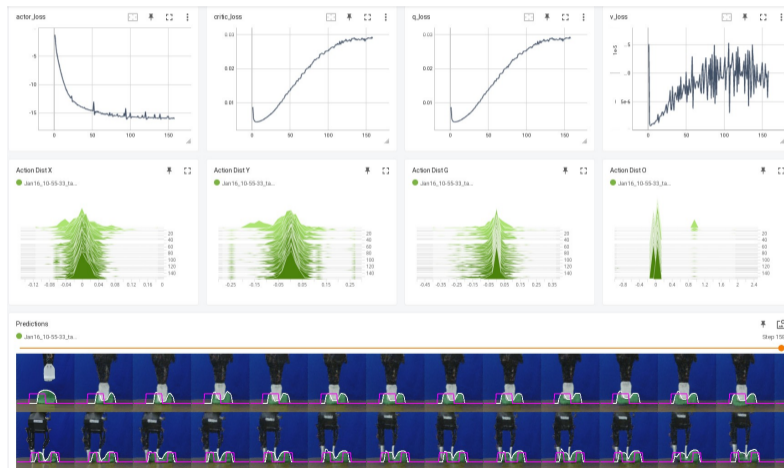




Table of Contents

Introduction

Related Work

Method

Planned Evaluation

Conclusion

References

Introduction

Related Work

Method

Goal and Setup

Data Augmentation

Training

Planned Evaluation

Conclusion



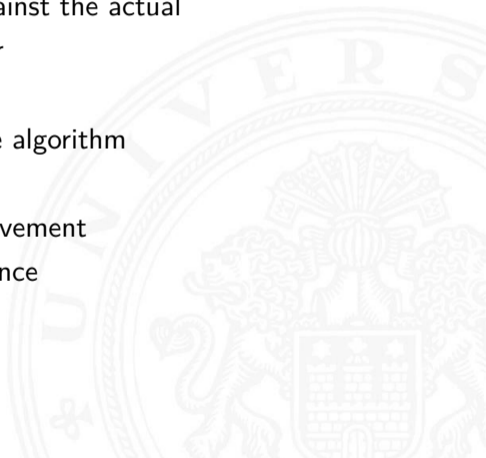


Dynamics model

- ▶ Predict trajectories given 1st state and actions
- ▶ Compare predicted trajectories against the actual
- ▶ image error - detected dough error

Policy

- ▶ Compare different buffers on same algorithm
- ▶ Fixed amount of steps
- ▶ Measure the relative reward improvement
- ▶ Compare against human performance



Current State - Video

Introduction

Related Work

Method

Planned Evaluation

Conclusion

References

The screenshot displays a ROS-based simulation environment with three main panels:

- Displays Panel:** Shows the configuration for the robot simulation. It includes a tree view with 'MotionPlanning' selected. The 'Robot Description' section is expanded, showing parameters like 'robot_description' (robot_description), 'move_group/monitor...' (move_group/monitor...), and 'arm' (arm). A 'Robot Description' text box contains the instruction: 'The name of the ROS parameter where the URDF for the robot is loaded'. Buttons for 'Add', 'Duplicate', 'Remove', and 'Rename' are visible.
- Playdoh Control Panel:** Features a 'Clear' button, a 'Pinch Factor' slider, and an 'Orientation' section with radio buttons for '-90°', '0°', and '+90°'. Below this is a video feed of a real-world robot arm (a Shadow Hand) positioned over a green dough-like object on a table.
- MotionPlanning Panel:** Contains a 'Context' dropdown set to 'Planning'. It has tabs for 'Planning', 'Joints', 'Scene Objects', 'Stored Scenes', 'Stored States', 'Status', and 'Mar'. The 'Commands' section includes buttons for 'Plan', 'Execute', 'Plan & Execute', 'Stop', and 'Clear octomap'. The 'Query' section has dropdowns for 'Plan Group:' (set to 'arm'), 'Start State:' (<current>), and 'Goal State:' (<current>). The 'Options' section includes sliders for 'Planning Time (s):' (5.0), 'Planning Attempts:' (10), 'Velocity Scaling:' (0.20), and 'Accel. Scaling:' (1.00). There are also checkboxes for 'Use Cartesian Path', 'Collision-aware IK', 'Approx IK Solutions', 'External Comm.', 'Replanning', and 'Sensor Positioning'. The 'Path Constraints' dropdown is set to 'None'. A 'Reset' button is located at the bottom left of the interface.



Table of Contents

Introduction

Related Work

Method

Planned Evaluation

Conclusion

References

Introduction

Related Work

Method

- Goal and Setup

- Data Augmentation

- Training

Planned Evaluation

Conclusion





Conclusion and Future Work

Introduction

Related Work

Method

Planned Evaluation

Conclusion

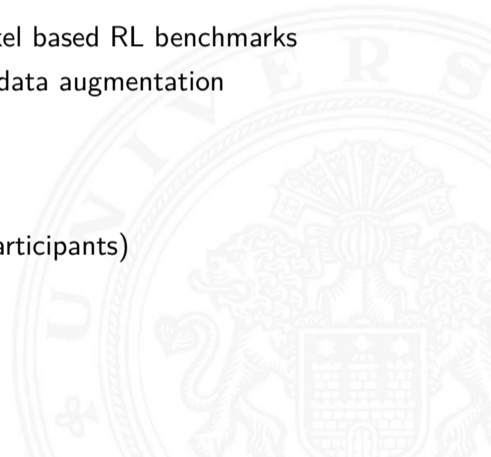
References

Conclusions so far

- ▶ Existing approaches to dough shaping are specialized
 - ▶ Hard to compare
- ▶ Setup being similar to existing pixel based RL benchmarks
- ▶ Dynamics model can be used for data augmentation

What still needs to be done

- ▶ Collect full dataset (at least 12 participants)
- ▶ Evaluate





- [Çal+15] Berk Çalli et al. “The YCB object and Model set: Towards common benchmarks for manipulation research”. In: *2015 International Conference on Advanced Robotics (ICAR)* (2015), pp. 510–517. URL: <https://api.semanticscholar.org/CorpusID:9954873>.
- [Che+22] Siwei Chen et al. “Benchmarking Deformable Object Manipulation with Differentiable Physics”. In: *ArXiv abs/2210.13066* (2022). URL: <https://api.semanticscholar.org/CorpusID:257482484>.
- [Kai+19] Lukasz Kaiser et al. “Model-Based Reinforcement Learning for Atari”. In: *ArXiv abs/1903.00374* (2019). URL: <https://api.semanticscholar.org/CorpusID:67856232>.



References (cont.)

- [KNL21] Ilya Kostrikov, Ashvin Nair, and Sergey Levine. “Offline Reinforcement Learning with Implicit Q-Learning”. In: *ArXiv abs/2110.06169* (2021). URL: <https://api.semanticscholar.org/CorpusID:238634325>.
- [KYF20] Ilya Kostrikov, Denis Yarats, and Rob Fergus. “Image Augmentation Is All You Need: Regularizing Deep Reinforcement Learning from Pixels”. In: *ArXiv abs/2004.13649* (2020). URL: <https://api.semanticscholar.org/CorpusID:216562627>.
- [Las+20] Michael Laskin et al. “Reinforcement Learning with Augmented Data”. In: *ArXiv abs/2004.14990* (2020). URL: <https://api.semanticscholar.org/CorpusID:216868834>.



- [MB21] Carolyn Matl and Ruzena Bajcsy. “Deformable Elasto-Plastic Object Shaping using an Elastic Hand and Model-Based Reinforcement Learning”. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2021), pp. 3955–3962. URL: <https://api.semanticscholar.org/CorpusID:235899408>.
- [Ond+22] Jan Ondras et al. “Robotic Dough Shaping”. In: *2022 22nd International Conference on Control, Automation and Systems (ICCAS)* (2022), pp. 300–307. URL: <https://api.semanticscholar.org/CorpusID:251223445>.



References (cont.)

Introduction

Related Work

Method

Planned Evaluation

Conclusion

References

- [Seo+22] **Younggyo Seo et al.** “Masked World Models for Visual Control”. In: *Conference on Robot Learning*. 2022. URL: <https://api.semanticscholar.org/CorpusID:250113367>.
- [Shi+22] **Haochen Shi et al.** “RoboCraft: Learning to See, Simulate, and Shape Elasto-Plastic Objects with Graph Networks”. In: *ArXiv abs/2205.02909* (2022). URL: <https://api.semanticscholar.org/CorpusID:248562698>.
- [Shi+23] **Haochen Shi et al.** “RoboCook: Long-Horizon Elasto-Plastic Object Manipulation with Diverse Tools”. In: *ArXiv abs/2306.14447* (2023). URL: <https://api.semanticscholar.org/CorpusID:259251806>.



- [SI21] Takuma Seno and Michita Imai. “d3rlpy: An Offline Deep Reinforcement Learning Library”. In: *J. Mach. Learn. Res.* 23 (2021), 315:1–315:20. URL: <https://api.semanticscholar.org/CorpusID:243847533>.
- [You+23] Yang You et al. “Make a Donut: Language-Guided Hierarchical EMD-Space Planning for Zero-shot Deformable Object Manipulation”. In: *ArXiv* abs/2311.02787 (2023). URL: <https://api.semanticscholar.org/CorpusID:265033452>.

Appendix - Reward Distribution

Introduction

Related Work

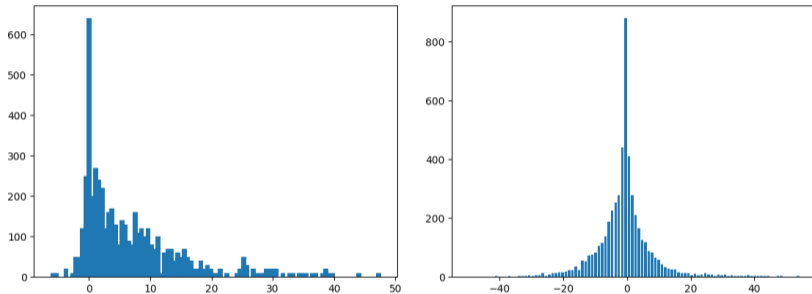
Method

Planned Evaluation

Conclusion

References

More negative rewards in model-based augmentations



Reward distribution over datasets with traditional (left) and model-based (right) augmentations.