Learning to Grasp and Stack Cube using Maniskill2 Environment

Arnold Dsouza
Pallavi Aithal Narayan
Salvin George
Vicky Prince Victor

***Abstract*—This study explores the development of robotic manipulation skills within a simulated environment, focusing on the task of stacking two differently colored cubes. By leveraging the ManiSkill2 framework, we integrate reinforcement learning and supervised learning approaches to teach a robotic arm the precision and adaptability required for stacking tasks. Our work demonstrates the effectiveness of simulation-based training, highlighting its potential for broader applications in robotic manipulation. The successful application of these learning methods provides insights into the challenges of robotic object manipulation and offers a foundation for future research in enhancing robotic interaction with physical objects.**

***Index Terms*—Robotic Manipulation, Simulation Training, Reinforcement Learning.**

I. Introduction

Robotic manipulation, a key domain within robotics, seeks to enhance how robots interact with their environment in precise and meaningful ways. Among various manipulation tasks, object stacking stands out due to its foundational importance for testing robotic precision, planning, and adaptability. This paper presents a focused study on developing a robotic arm's capability to perform a specific stacking task within a controlled simulation environment. Our research harnesses a sophisticated simulation framework, allowing us to train and test our robotic arm in a virtual setting before any real-world application.

The task at hand involves the robot accurately stacking two cubes of distinct colors: placing a red cube atop a green cube. This seemingly simple task encapsulates multiple underlying challenges fundamental to robotic manipulation, including object recognition, spatial reasoning, and the delicate control of force and balance. The specificity of the task allows for a detailed investigation into the manipulation capabilities of the robot, providing insights that are applicable to a broader range of tasks in robotic manipulation.

Utilizing a simulation environment for this purpose offers numerous advantages, including the ability to rapidly prototype and test various approaches without the need for physical trials, which can be time-consuming and resource-intensive. Furthermore, it allows for precise control over experimental conditions, ensuring that the robot's performance can be evaluated consistently across different trials.

Our approach involves training the robot using a combination of reinforcement learning and supervised learning techniques, enabling it to learn from both programmed instructions and feedback from its interactions within the simulation environment. This mixed-methods approach aims to equip the robot with the flexibility and precision required to successfully complete the stacking task, with the ultimate goal of applying these learned skills to real-world scenarios.

In summary, this paper documents our journey and findings in teaching a robotic arm to execute a precise stacking task within a simulated environment. By focusing on this specific task, we aim to contribute valuable insights to the field of robotic manipulation, showcasing the potential of simulation-based training in developing advanced robotic skills.

II. Related Work

Research in robotic manipulation has focused on enabling robots to interact with dynamic and unstructured environments, such as the Maniskill 2 soft-body and rigid-body tasks. Several approaches have been proposed to tackle similar challenges.

Many researchers have explored the application of reinforcement learning (RL) techniques for robotic grasping and manipulation tasks. Approaches including Policy Gradient Methods such as Demonstration-Augmented Policy Gradient(DAPG), Proximal Policy Optimization (PPO), Soft Actor-Critic (SAC), Generative-Adversarial Imitation Learning(GAIL) have been utilized to train robotic agents to grasp and manipulate objects in diverse environments for rigid-body tasks similar to our approach.

SAPIEN ManiSkill [1] has proposed a full-physics simulation benchmark for manipulating a variety of 3D objects. This benchmark allows agents to be trained using a large-scale dataset of demonstrations and assesses their generalization ability in unseen scenarios in testing environments.

In [2] the authors have proposed a novel two-stage fine-tuning strategy that aims to further enhance the generalization capability of model based on the Maniskill2 benchmark. In this work, they utilize PointNet to extract the point cloud features, and then employ Reinforcement Learning or Imitation Learning algorithms [3] to ascertain the agent's direction and traveling distance. Once again the usage of Proximal Policy

*Submitted to the Department of Computer Science at Hochschule Bonn-Rhein-Sieg in partial fulfilment of the requirements for the degree of Master of Science in Autonomous Systems
†Supervised by Dr. Mitrevski, Aleksandar
‡Submitted in March 2024

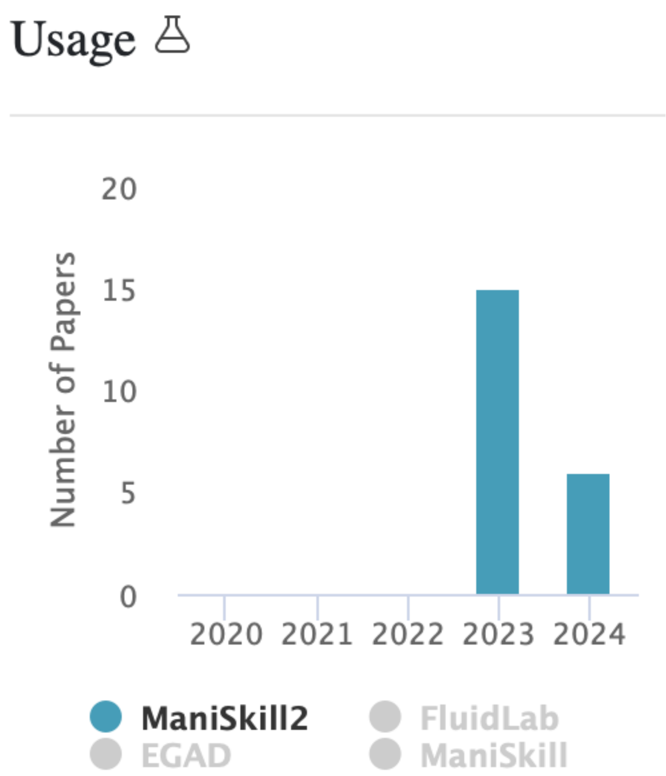


Fig. 1: Maniskill2

Gradient (PPO) [4] for rigid-body tasks and Behavior Cloning (BC) [5] for soft-body tasks can be seen.

In the robotics field, the need for manipulation benchmarks is well acknowledged, and it is still a hot topic of conversation at robotic manipulation workshops. Large datasets of object scans have typically been the focus of earlier works. Previous studies concerning datasets have primarily centered around extensive collections of object scans. While these datasets are valuable for various simulation and planning purposes, as well as for benchmarking in experimental research on grasping and manipulation is limited. One notable limitation in prior research is that most manipulations in these datasets are not easily accessible to other researchers, thus hindering their use in manipulation experiments, except for a few cases.

This current endeavour stands out due to its dual purpose: it not only furnishes a wealth of information about objects essential for many simulation and planning methodologies but also ensures the availability of the actual demonstrations for researchers to employ in experimental settings for learning from demonstrations [6].

III. Methodology

The ManiSkill2 framework, which is intended to support the development of broadly applicable robot manipulation abilities, serves as the foundation for the project's methodology. The fundamental component of this strategy is imitation learning, a machine learning technique that lets robots pick up difficult jobs by mimicking the actions of experts. This is especially helpful in situations where it is impractical to programme every possible course of action and result. The project makes use of an expert demonstration dataset, which is kept in an H5 file and consists of a series of observations and related actions. The details of the project are provided below. The objective is to allow a robot to pick up a red cube and position it steadily on top of a green cube without having to grab it.

- Goal: Take a red cube and set it on top of a green cube.
- Success Metric: The red cube is steadily positioned on top of the green one without being gripped.
- Demonstrations: The format for the demonstration is 1,000 successful trajectories.
- Evaluation Protocol: 100 episodes with varying robot joint initial positions and initial cube poses.

The purpose of these examples is to train a neural network policy that controls the movements of the robot. By customising the baseline policy to the cube stacking environment and optimising the training settings, the project expands upon the cube selection methods offered by ManiSkill2. The neural network uses a Tanh activation layer to make sure that the output actions are within the [-1, 1] range, matching the robot's action space. It is built using three hidden layers, each of size 128. Full connected layers and ReLU activation functions are also used in its construction. Because of this structure, the robot can learn from examples, which greatly reduces the amount of complex manipulation tasks that require manual programming.

The training process iteratively increases the precision of the policy by calculating loss using SmoothL1Loss, optimising weight using the Adam optimizer, and performing forward runs through the policy network. Determining the robot's stacking success rate is essential for tracking improvements in performance. The ManiSkill2 environment and the gymnasium library, which provide a standardised framework for environment interaction, observation, and control, further complement the technique. This all-encompassing approach integrates neural network training, efficient data management, imitation learning, and careful assessment to enable skilled robot manipulation in challenging tasks.

Multiple controllers, including joint position, delta joint position, and delta end-effector pose, are supported by the ManiSkill2 environment. These controllers convert input actions into joint torques that power the robot's motors. The ability to support several controllers is crucial for customising demonstration action spaces to meet the unique needs of the current activity. The project's joint position controller demonstrations have been modified for training in a delta end-effector pose controller setup for rigid-body scenarios. This modification is essential to the project's state-based methodology, which makes use of a dictionary of states that includes ground truth object information (e.g., object positions), task-specific goal information, and robot proprioceptive data. The robot proprioceptive information includes joint positions, joint velocities, the pose of the robot base, and the pose of the gripper's tool center point if the robot uses a two-finger gripper. This rich state information is pivotal for the robot to understand and interact with its environment effectively, enabling it to perform the task of stacking cubes with precision and reliability.

IV. Evaluation

The ManiSkill2 challenging environments are used to evaluate the effectiveness of the Behavior Cloning approach in acquiring robot manipulation abilities [7]. A variety of manipulation tasks with varying difficulty levels are provided in these environments, enabling a thorough evaluation of the learned policy's capabilities.

The success rate is chosen as the primary metric to quantify the policy's performance. The percentage of instances where the agent achieves the specified objective of the environment is reflected in this indicator. The achievement of a block stacking endeavor might be defined as the achievement of all the blocks in the desired arrangement. The episodic success rate is a valuable metric because it directly measures the agent's ability to complete the intended task. This is crucial for real-world robotic applications.

To ensure an unbiased evaluation and prevent overfitting to the training data, we employ a separate evaluation phase after training the policy using BC. This is to ensure an unbiased evaluation and prevent overfitting to the training data. During evaluation, the weights of the policy's weights are frozen

$$(th.no_g rad())$$

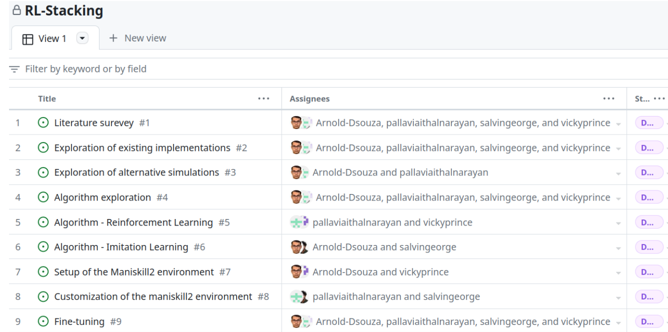
to assess its true generalization capability on unseen episodes.

To account for the inherent variability and stochasticity present in manipulation tasks, the evaluation process involves running the agent on multiple evaluation episodes. This enables a more statistically robust assessment of the policy's accomplishments and its generalizability. The trained policy is used by the agent to interact with the environment in each episode. If the agent achieves the defined goal, the episode is successful.

The success rate for a particular episode is calculated by dividing the number of successful episodes by the total number of evaluation episodes. A high percentage of outcomes indicates that the learned policy is capable of imitating the expert's conduct and achieving the objectives in the majority of evaluation episodes, indicating successful application of the lessons learned from the demonstrations. A low rate of success, on the other hand, suggests weaknesses in the learned policy, possibly resulting from inadequate documentation of expert actions or difficulties with particular job complexities.

The periodic success rate is a useful metric, but it might be useful to take into account additional task-specific metrics, depending on the particular manipulation task. The time it takes to accomplish the task or the quality of the result could be one of the factors.

We get a comprehensive assessment of the policy's capability learned through BC on ManiSkill manipulation tasks by employing a periodic success rate and adhering to these rigorous evaluation procedures. This assessment assesses the efficacy of the BC strategy for robot skill acquisition in this challenging field.


RL-Stacking

View 1 | + New view

Filter by keyword or by field

	Title	Assignees	St...
1	Literature surevey #1	Arnold-Dsouza, pallaviaithalnarayan, salvingeorge, and vickyprince	D...
2	Exploration of existing implementations #2	Arnold-Dsouza, pallaviaithalnarayan, salvingeorge, and vickyprince	D...
3	Exploration of alternative simulations #3	Arnold-Dsouza and pallaviaithalnarayan	D...
4	Algorithm exploration #4	Arnold-Dsouza, pallaviaithalnarayan, salvingeorge, and vickyprince	D...
5	Algorithm - Reinforcement Learning #5	pallaviaithalnarayan and vickyprince	D...
6	Algorithm - Imitation Learning #6	Arnold-Dsouza and salvingeorge	D...
7	Setup of the Maniskill2 environment #7	Arnold-Dsouza and vickyprince	D...
8	Customization of the maniskill2 environment #8	pallaviaithalnarayan and salvingeorge	D...
9	Fine-tuning #9	Arnold-Dsouza, pallaviaithalnarayan, salvingeorge, and vickyprince	D...

Fig. 2: Team Contributions

V. Contributions

This collaborative effort ensured a comprehensive exploration of the research landscape, meticulous implementation of algorithms, and meticulous experimentation within the Maniskill environment.

References

[1] Sapien, "ManiSkill2 Challenge," https://sapien.ucsd.edu/challenges/maniskill/, accessed: March 15, 2024.

[2] F. Gao, X. Li, J. Yu, and F. Shaung, "A two-stage fine-tuning strategy for generalizable manipulation skill of embodied ai," *arXiv preprint arXiv:2307.11343*, 2023.

[3] X. Xu, L. Zuo, and Z. Huang, "Reinforcement learning algorithms with function approximation: Recent advances and applications," *Information sciences*, vol. 261, pp. 1–31, 2014.

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

[5] F. Torabi, G. Warnell, and P. Stone, "Behavioral cloning from observation," *arXiv preprint arXiv:1805.01954*, 2018.

[6] T. Mu, Z. Ling, F. Xiang, D. Yang, X. Li, S. Tao, Z. Huang, Z. Jia, and H. Su, "Maniskill: Generalizable manipulation skill benchmark with large-scale demonstrations," *arXiv preprint arXiv:2107.14483*, 2021.

[7] OpenAI, "ManiSkill," https://openai.com/research/universe, accessed: March 15, 2024.