The Application of Artificial Intelligence and Reinforcement Learning in Robotic Object Manipulation

Ashwin Gurusankar

Artificial intelligence has the potential to transform how robots interact with bodies in diverse environments, allowing them to truly imitate how a human, with innate fine motor control and spatial awareness, handles manipulation based tasks. Developing AI agents to, in turn, enable the creation of intelligent robots allows for tasks to be handled in a much simpler manner; rather than rigidly configuring robots to perform a specific type of task and reconfiguring when new stimuli, such as a new product for production in a factory, are added to the environment. Implementing AI in these scenarios would improve the overall control, decision making, and perception of the robot. However, although the integration of machine learning or deep learning has shown promise in various settings, there are still many significant limitations in the field: the large amounts of data needed for this approach, the experimentation needed to create the components of a model of this scale, and the computational cost of training a model of this scale.

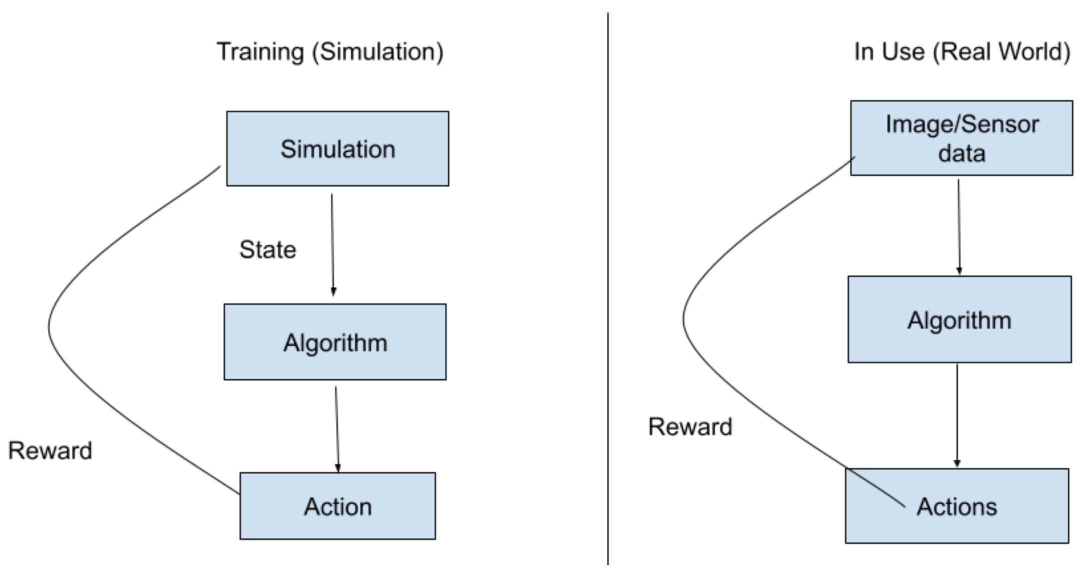
Recent studies in artificial intelligence (AI) have significantly advanced robotic object manipulation, by allowing robots to more confidently interact and perform complex tasks in varying environments. Eshani et al. (2021), collaborated to bring **ManipulaTHOR**, an extension to the **AI2-THOR** 3D simulator for robotics arms, as well as the Arm Point Challenge for the Embodied AI community. Though models have been trained with **AI2-THOR** , these models are of a higher level and **ManipulaTHOR** aims to simplify this process. The ArmPointNav task is dedicated to building towards the goal of generalizable object manipulation. This challenge requires an agent that can control and allow a robot to move an object from an initial to a target location using a dataset consisting of scenes from **AI2-THOR**. Through initial experimentation they found models suitable for navigation were not as effective for object manipulation, displaying room for improvement in this sector of Embodied AI. Another one of these experiments led by researchers from these institutes called **Manipulation Via Visual Object Location Estimation (m-VOLE)**, which allows an agent to explore the environment for the target object and estimate the 3D location even in scenarios where the object is not visible. It uses a technique known as Conditional Segmentation which takes the current environment and a RGB picture of the desired object to generate a mask for the agent to use when finding and manipulating it. This architecture allows for AI agents to eliminate the need to make inferences about the object being in the ideal position.

Along with these advancements in simulation, advances in generalizing object manipulation through other types of learning, researchers at **MIT** helped robots operate with objects using their entire body rather than just using a single mechanism—such as a claw. By using a technique known as *smoothing*, which reinforcement learning is found to do implicitly, these researchers could simplify the decisions to be more cost-efficient, while eliminating choices that take away from efficiency, allowing robots to manipulate objects using their entire body.

Another example of reinforcement learning for object manipulation is the **OpenAI Dactyl project**. Dactyl is attempting to create a robotic hand with the same level of dexterity as the human hand. The success of Dactyl is seen in its use of the reinforcement learning algorithm known as *domain expansion*, a technique involving training the robot in a simulated environment. This allows for the features of the object to be easily adjusted at random for maximum exposure. Though this method is extremely computationally intensive, with OpenAI using a cluster of 64 **NVIDIA V100 GPUs**, the results are undeniably extraordinary. The project was able to achieve a success rate of above 90 percent. The principles applied to Dactyl can be applied to other sectors of AI and robotics, making this achievement a crucial step in AI in robotics as well as object manipulation.

Though these advancements are certainly groundbreaking, they deal with rigid objects; however, deformable objects add another layer of complexity to manipulation. Lin et al. (2021) present their progress on the issue with *SoftGym*, built on the **NVIDIA FleX** simulator. Their simulator includes tasks such as folding cloth or pouring fluids, all for advancing reinforcement learning in more complex environments. Using *SoftGym*, they have found that current reinforcement models have a much greater difficulty predicting the behavior of deformable objects, with success rates leaving much room for improvement. This research allows us to see the current limitations in our approach as well as future challenges.

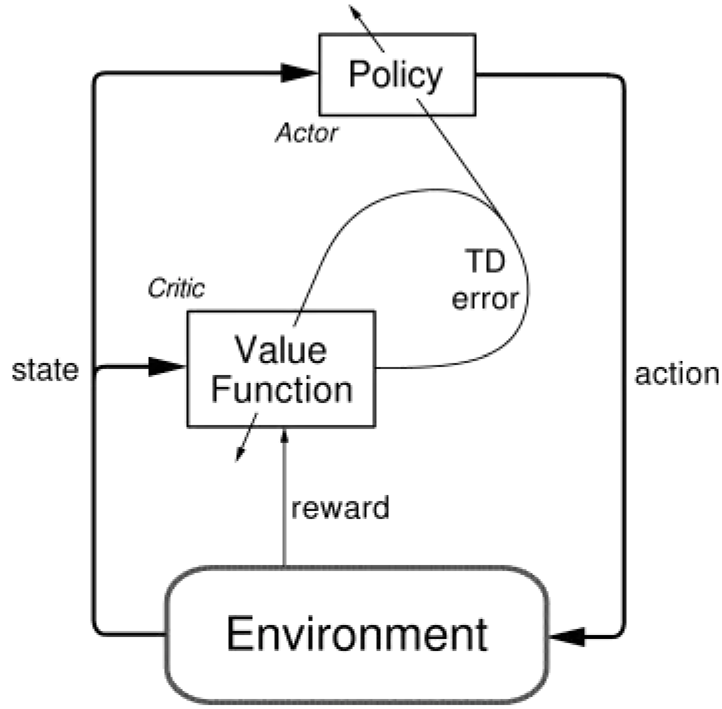
Figure 1
Diagram of model in Simulation and Real World



Note. Figure Created by the author based on proposition.

The proposed approach incorporates reinforcement learning in simulated environments. This framework would allow the agent–and by extension, the robot–to learn by making mistakes and better understand how to maximize its effort when manipulating objects. Furthermore, reinforcement learning would allow the robot to act more efficiently in more complicated or ambiguous situations as well as learn from these situations, contributing to the development of control policies. The simulated environment would allow for the user to change objects and randomize characteristics for maximum accuracy and simulation to real world transfer. The proposition relies on sensors, such as cameras, to provide RGB-D images, joint positions, as well as a reward system based on tactile sensors awarding or punishing based on whether the task is close to being completed or failed. The dataset required would be generated in simulated environments, allowing for safe and scalable training options. The learning would be modeled by Deep Reinforcement Learning algorithms for their ability to be stable and efficient in control tasks. Reinforcement learning would be the central focus for using artificial intelligence in robotics and object manipulation. Reinforcement learning would be most beneficial as it allows the robot to learn from interacting in the environment, optimizing long term rewards rather than relying on hard coded rules, not allowing for any flexibility in situations developers didn't specifically intend for. Along with this Reinforcement Learning can provide a useful framework for users to implement allowing for robots that can easily make decisions, understanding its environment better than any regular program could implement. In order to collect data, a simulation would be the best approach as there are already a number of frameworks and benchmarks to use such as **Nvidia Omniverse** and **SoftGym**. Through simulation we can randomize features of the object through large collections of objects already existing as well as through techniques such as domain randomization allowing for the gap between simulation and reality to be closed as well as allowing for the agent to robustly act in new situations. For assessing models common benchmarks of reinforcement learning for manipulation are needed. These benchmarks should take into account aspects of the robot such as tactile feedback, object orientation and planning. Aligning data acquisition with Reinforcement learning will allow for many scalable and fully autonomous robots.

Figure 2
Diagram of An Actor Critic System



Note. Adapted from *Reinforcement Learning: An Introduction* (Chapter 13, “Policy Gradient Methods”), by R. S. Sutton & A. G. Barto, 2018, MIT Press. http://incompleteideas.net/book/the-book-2nd.html.

Reinforcement learning would remain at the center of any solution for robots to learn. Specifically, a reward system utilizing models such as Soft Actor-Critic, allowing an Actor to learn a policy model utilizing Proximal Policy Optimization for stable training and a critic using a value function to then evaluate the Actor’s abilities. Along with the core reinforcement learning a supervised learning for object detection would allow for a robot to act better in a real world environment, a classification as well as a regression problem that Convolutional Neural Networks can be used to solve. The value function learned by the critic would also be learned via regression and for perception, transformers would be a good solution. For evaluating the model, a supervised learning core would make Reward based metrics the best, for example measuring the cumulative reward as well as making sure that the model is converging on an optimal method along with making generalizations about novel objects through domain randomization can be learned through looking at vast amounts of data. A CNN would be a good choice of model as well.

This solution would be the optimal as it would allow for a robot to have the same level of cognitive function as a human. Like humans, robots with reinforcement learning would achieve a unification of perception and control because it allows a robot to learn from interaction with the environment therefore bridging the gap between perceiving an object and then using motor control to then operate with said object. This eliminates the need for premade controllers that are not only non adaptable, but require many more layers of complexity when dealing with more cerebral tasks. Another way robots can act more like humans is with the additional features of transfer policies and realtime feedback integration. Policies that an agent would learn include grasping an object and moving the robot to a specific point would be crucial and just like a human, a robot can then apply this base knowledge to a variety of environments even with noise naturally presented in new environments. Moreover, realtime feedback is another crucial feature as it allows for the agent to make live adjustments when inferences occur.

Along with being able to closely mimic that of a human operator, compared to other types of ML, Reinforcement Learning is the best for this task as it allows for the model to constantly learn from each interaction all without the expert knowledge and data that other models would require for minimal function. Another reason RL is the best form of modelling for this task is that it allows for more versatility in the the tasks it performs, through processes like domain randomization and processes implicitly integrated like smoothing, allows for robots to make more generalized movements as well as follow patterns that can be adjusted if necessary as it comes across what it believes to be the most optimal policy.

Numerous prior studies have demonstrated that reinforcement learning (RL) provides a powerful framework for enabling robots to behave more like human operators. The following works support the core aspects of this proposal, including perception-control integration, policy generalization, and real-time adaptability. The use of reinforcement learning is already a viable option, proven by Haochen et al (2025), who demonstrated the benefits of RL when they developed Hamlet, a robot that can play badminton using learning based manipulation with model based locomotion, using reinforcement learning policies for the arm to determine factors such as arm velocity based on a prediction of the ball's trajectory. Therefore, requiring minimum control of actuators, letting the model perform much more confidently in a scenario like an actual match where the movement of the game piece would be much more randomized or more difficult for the robot's success.

An example of the simulation to real life gap and transfer learning is OpenAI's **Dactyl** Project, The robot can do many things like move an object such that a particular face is being shown as well as competently move an object and this framework has allowed for Dactyl to continue to improve, being able to solve a rubik's cube showing its capabilities in being multipurpose. Additionally, the use of techniques such as domain randomization aided in the accuracy outside of the simulation. Reinforcement learning also implements processes necessary for object manipulation as it relies on finding the solution that maximizes the reward. Pang et al (2023) have come across one of these mechanics known as smoothing. When robots are manipulating objects the myriad of contact points can cause longer run time and a lack of effectiveness when the model revalues; hence, why physics based models appear to not be as effective. However, reinforcement learning fixes this with a technique implicitly added called Smoothing. This technique averages out insignificant decisions made by the agent and narrows the scope to a set of vital actions. Using reinforcement learning then becomes the obvious decision as it allows for a much more robust model and allows the robot to reduce the time required for the task it is performing.

Though the results of an object manipulation system using Reinforcement Learning like this would yield incredible results, the data required for this endeavor would be a challenge when trying to train this model. Since the model would be trained in a simulation, there would need to be a large set of objects in the simulation for the agent to properly learn and achieve a general approach to object handling. This can be a problem as there is no large scale dataset of objects that can be ported to simulators similar to that of **ImageNet**. This has led to creating datasets being the first set and requiring large amounts of experimentation for accurate models such as ManipulaThor. Along with this issue, the training for a reinforcement model of the sort would be computationally intensive as projects like OpenAI's **Dactyl** and other robots require large clusters of high end GPUs for training which is not a viable option for all who would want to use train models in this fashion.

Other difficulties such as the speed at which learning takes place, the creation of a reward function and the drawbacks of transfer learning still hinder our ability to fully optimize our . First, time consumption can be a massive drawback to reinforcement learning as even in a simulation a robot can take millions of attempts before developing the proper policy for a task, and factoring in techniques implemented for accuracy such as domain randomization can increase the time required tremendously. This makes reinforcement learning not only computationally intensive, but also time consuming. Second, reward functions can be incredibly difficult to implement as it requires consideration of many factors such as sparse rewards where the robot receives no input until it randomly comes upon a solution, that can also be extremely inefficient without intermediate feedback, another is the agent prioritizing other tasks to maximize the rewards because intermediate rewards are constantly given. The fact that the agent can ignore the task at hand and find loopholes to maximize its reward is another big concern when developing a reward function for an agent. All of these facts display how the creation of a stable reward function can be extremely difficult to do and requires testing. Finally though tools such as transfer learning that would be implemented alongside Reinforcement Learning would reduce the need to re-train a model, there are many downsides as well including when transferring the existing model to a train for a new task, it can overwrite the principals the agent learned previously, additionally a transferred model might implement more bias not exploring the environment as effectively because of its past training.

Some immediate next steps for a framework that could make object manipulation viable would be experimentation that would minimize the problems found. For example, a database of simulation objects similar to that of imagenet would make training a model to generally handle objects very effective, this would allow for an agent to train on a wide variety and understand an everyday environment filled with objects that may be mishandled if the model was not trained on objects at least similar in shape and properties. Another is testing the different functions that can be used to reward a model testing different ways of rewarding. Finding the most efficient reward function would allow for a model to then become more efficient and understand the most important actions required when handling an article. Figuring out how to build a Reinforcement Learning model that incorporates the following factors as well as being easy to build upon would be a large, but necessary task. This advancement would allow for others to build upon the model with the basic grasping function and general perception already built in and it can then be specialized for its specific purpose.

In conclusion, Reinforcement Learning paired with the necessary techniques would offer a scalable and adaptable path for general purpose object manipulation prioritizing flexibility and learning from different interactions. While traditional systems rely on manual tuning and many hand crafted rules, RL allows robots to autonomously develop policies for handling objects through interactions. Though problems in data acquisition, reward design, and sim to real life transfer hinder the advance of these models, advancements in simulation technology and other techniques to balance these problems are promising avenues to overcome these limitations. This paper proposed a perspective where RL is the center of an agent with other types of models and techniques such as domain randomization to complement its usage. By investing resources in training data as well as modular reward functions, robots can evolve with more adaptable and resilient systems allowing for larger benefits such as implementation in more dynamic environments and greater autonomy in these systems allowing for faster reaction if new stimuli presents itself. Reinforcement Learning and Artificial Intelligence's integration into robotic object manipulation is not only promising, but necessary for the evolution of general purpose robotic systems.

References

Andrychowicz, M., Baker, B., Chociej, M., Józefowicz, R., Mcgrew, B., Pachocki, J., Petron, A., Plappert, M., Powell, G., Ray, A., Schneider, J., Sidor, S., Tobin, J., Welinder, P., Weng, L., & Zaremba, W. (2019). *Learning Dexterous In-Hand Manipulation*. https://arxiv.org/pdf/1808.00177

Ehsani, K., Farhadi, A., Kembhavi, A., & Mottaghi, R. (2022). *Object Manipulation via Visual Target Localization*. https://arxiv.org/pdf/2203.08141

Ehsani, K., Han, W., Herrasti, A., Vanderbilt, E., Weihs, L., Kolve, E., Kembhavi, A., & Mottaghi, R. (2021). *ManipulaTHOR: A Framework for Visual Object Manipulation*. https://openaccess.thecvf.com/content/CVPR2021/papers/Ehsani_ManipulaTHOR_A_Framework_for_Visual_Object_Manipulation_CVPR_2021_paper.pdf

Karunakaran, D. (2020, December 2). *Soft Actor-Critic Reinforcement Learning algorithm*. Intro to Artificial Intelligence. https://medium.com/intro-to-artificial-intelligence/soft-actor-critic-reinforcement-learning-algorithm-1934a2c3087f

Lin, X., Wang, Y., Olkin, J., & Held, D. (2021). *SoftGym: Benchmarking Deep Reinforcement Learning for Deformable Object Manipulation*. https://proceedings.mlr.press/v155/lin21a/lin21a.pdf

Pang, T., Suh, H. J. T., Yang, L., & Tedrake, R. (2023). Global Planning for Contact-Rich Manipulation via Local Smoothing of Quasi-Dynamic Contact Models. *IEEE Transactions on Robotics*, *39*(6), 4691–4711. https://doi.org/10.1109/tro.2023.3300230

Wang, H., Shi, Z., Zhu, C., Qiao, Y., Zhang, C., Yang, F., Ren, P., Lu, L., & Xuan, D. (2025). *Integrating Learning-Based Manipulation and Physics-Based Locomotion for Whole-Body Badminton Robot Control*. ArXiv.org. https://arxiv.org/abs/2504.17771

Zewe, A. (2023, August 24). *AI helps robots manipulate objects with their whole bodies*. MIT News | Massachusetts Institute of Technology. https://news.mit.edu/2023/ai-technique-robots-manipulate-objects-whole-bodies-0824