Advances in Autonomous Robotic Grasping

An Overview of Reinforcement Learning Approaches

Bálint Károly Farkas
Alba Regia Faculty, Óbuda University, H-8000
Székesfehérvár, Hungary
farkas.balint@amk.uni-obuda.hu

Péter Galambos *Senior Member, IEEE*
Antal Bejczy Center for Intelligent Robotics, Óbuda University
Budapest, Hungary
peter.galambos@irob.uni-obuda.hu

Károly Széll
Antal Bejczy Center for Intelligent Robotics, Óbuda University
Alba Regia Faculty, Óbuda University, H-8000
Székesfehérvár, Hungary
szell.karoly@amk.uni-obuda.hu

***Abstract*—Reinforcement Learning (RL) has become a pivotal tool in robotic manipulation, especially for tasks requiring adaptability, such as object grasping. By allowing robots to learn optimal behaviors through trial and error, RL, when integrated with advanced vision systems, enhances the ability of robots to interact effectively with their environment.**

This paper reviews the fundamental aspects of RL in robotic grasping, including perception, decision-making, and execution, and explores practical applications in frameworks like "Laboratory Automation Plug and Play (LAPP)". While significant advancements have been made, challenges such as sim-to-real transfer, computational efficiency, and robustness remain critical issues.

Additionally, the role of vision systems in object detection and grasp planning is examined, along with the evaluation of performance in simulated and real-world settings. The paper concludes by discussing future directions, emphasizing the need for improved algorithms, better sim-to-real techniques, and multi-agent cooperation.

Overall, RL continues to advance autonomous robotic systems, enhancing their precision and capability in complex tasks.

***Index Terms*—Reinforcement Learning; Robotic Grasping; Autonomous manipulation; System integration;**

I. Introduction

Reinforcement Learning (RL) has emerged as a key technique in the field of autonomous robotic manipulation, addressing complex tasks that require adaptive and intelligent decision-making. RL involves training agents to perform tasks by rewarding desirable actions and penalizing undesirable ones, enabling robots to learn optimal behaviors through trial and error. This approach is particularly effective for tasks such as object grasping and manipulation, where precise control and adaptability are essential.

In robotics, the combination of RL with advanced vision systems has proven to be transformative. Vision systems, powered by deep learning, provide robots with the capability to perceive and understand their environment, identifying and interacting with objects with a high degree of accuracy. By integrating RL, these robots can adjust their actions based on real-time visual feedback, leading to more efficient and accurate manipulation.

The application of RL in robotic grasping involves several key components:

- **Perception:** Utilizing vision systems to detect and recognize objects [1].
- **Decision-Making:** Employing RL algorithms to determine the optimal grasping strategy [2] [3].
- **Execution:** Implementing the learned strategy to manipulate the object.

A notable framework that illustrates the integration of these components in a practical setting is the LAPP framework [4]. This framework aims to enhance automation in pharmaceutical labs by integrating flexible mobile manipulators capable of performing complex tasks. A significant challenge in such automation systems is the integration of devices from various vendors, each with its own interface. The LAPP framework addresses this by using mobile manipulators with vision systems to "learn" the poses of different devices and fetch interface information from a universal cloud database. This approach aims to standardize control and communication protocols, facilitating smoother integration and operation of diverse devices [4]. Developing an autonomous manipulator with a robotic arm that incorporates adaptive grip control, real-time feedback mechanisms could substantially improve precision and reliability in handling delicate or irregularly shaped objects, thereby pushing the boundaries of current robotic systems.

Despite the significant advancements, several challenges remain in the field of RL-based robotic manipulation concentrating around to below-listed topics:

- **Learning from Demonstration**
- **Efficient Deep RL Algorithms**
- **Collision Avoidance**
- **Selecting Suitable Deep RL Methods**
- **Sim-to-Real Transfer:** Closing the gap between simulated training environments and real-world applications

[5].

- **Computational Demands:** Addressing the high computational requirements of RL algorithms.
- **Data Requirements:** Ensuring sufficient and high-quality data for effective training.

Future research is directed towards overcoming these challenges, with a focus on improving algorithmic efficiency, enhancing sim-to-real transfer techniques, and developing robust data collection methodologies. The continued evolution of RL in robotics promises to further advance the capabilities of autonomous systems, enabling them to perform increasingly complex tasks with higher precision and reliability [6].

II. Fundamentals of RL

RL is a type of machine learning where an **agent** (The entity that makes decisions) learns to make decisions by interacting with its **environment** (The external system with which the agent interacts). The agent has a **state** (S - A representation of the current situation the agent is in) and the agent makes an **action** (A - The set of all possible moves or decisions the agent can take) and it will get a feedback: **reward** (R - Feedback from the environment based on the action taken, which can be positive or negative). This process can be seen in figure 1. The agent has a **policy** (π - A strategy that the agent uses to decide which actions to take based on the current state.) Policies can be deterministic or stochastic. The learning process is based on the concept of trial and error, where the agent aims to maximize cumulative rewards over time [6]. This requires a balance between **exploration** (trying out new actions to discover their effects) and **exploitation** (choosing the best-known actions to maximize immediate rewards). Effective RL algorithms are designed to manage this trade-off, ensuring that the agent can both learn about its environment and optimize its actions to achieve the best long-term outcomes.

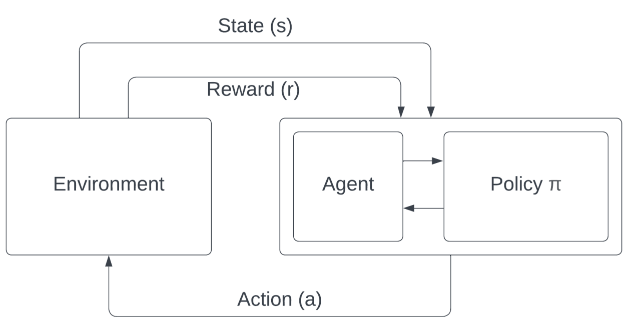

Fig. 1. State-Action-Reward Loop [6]

A. Markov Decision Process (MDP)

An MDP provides a formal framework for decision-making in stochastic environments. It builds upon the concept of Markov chains by incorporating actions and rewards, allowing for the optimization of decisions. The essence of an MDP lies in its Markovian property, which asserts that the future state depends only on the current state and action, and not on the sequence of events that preceded it.

MDPs are characterized by a set of states, actions, transition probabilities, and rewards. The goal is to determine the optimal policy, a mapping from states to actions, that maximizes the cumulative reward over time. In an MDP, the environment is assumed to be stationary, meaning that the transition probabilities and reward functions do not change over time. This property simplifies the decision-making process as it eliminates the need to account for past states or actions.

Applications of MDPs span various fields, including operations research, control theory, statistics, and artificial intelligence. They are used to model and solve problems in game theory, optimal investment strategies, medical testing, logistics, and more [6].

B. Key RL Algorithms in Robotic Grasping

The application of RL in robotic grasping has led to various approaches, each with its strengths and weaknesses. This section introduces the key algorithms then compares different RL-based methods to highlight their effectiveness, efficiency, and adaptability.

- **Deep Deterministic Policy Gradient (DDPG):** DDPG is an algorithm that combines elements of Q-Learning and policy gradient methods within an actor-critic architecture, where actor selects actions and the critic evaluates them. This makes DDPG particularly effective for tasks requiring precise and adaptive motor control, such as grasping objects of varying shapes and sizes [7].
- **Soft Actor-Critic (SAC):** SAC is a state-of-the-art reinforcement learning algorithm designed for continuous action spaces, which combines elements of both value-based and policy-based methods [8]. The successful use of SAC in robotic object grasping, particularly with previously unseen objects, has been demonstrated in recent research [9].
- **Proximal Policy Optimization (PPO):** PPO is an algorithm that optimizes policy by adjusting the parameters to maximize the expected reward. This method is suitable for continuous action spaces and dynamic environment [10]. It balances exploration and exploitation by adjusting the policy within a safe region.
- **Q-Learning and Deep Q-Network (DQN):** Q-Learning is an off-policy algorithm that learns a Q-function to represent the value of taking an action in a given state. It updates the Q-values using the Bellman equation, iteratively refining its policy by selecting actions that maximize the expected future rewards. DQN is an algorithm that extends Q-Learning by using Deep Neural Network to approximate the Q-function [11]. This method is good if there is a simulation environment, and we can transfer it to the real-world environment [12].
- **Advantage Actor-Critic (A2C) and Asynchronous Advantage Actor-Critic (A3C):** A2C and A3C are algorithms that use an actor-critic architecture where the actor

determines actions, and the critic evaluates them. A2C synchronizes updates across agents, while A3C allows asynchronous updates enabling faster and more diverse learning.

- **Imitation Learning (Behavior Cloning and Inverse Reinforcement Learning (IRL)):** Imitation Learning is a machine learning approach where an agent learns to perform tasks by observing and mimicking the actions of an expert, rather than through the trial-and-error process characteristic of RL. Figure 2 shows the main difference between RL and Imitation Learning.

  A common technique within imitation learning is **behavior cloning**, where the agent learns a direct mapping from states to actions using supervised learning. This allows for rapid initial training, as the agent leverages the knowledge embedded in the expert demonstrations without needing to explore the environment independently. Imitation learning can be highly effective when combined with RL. For example, an agent may first learn a baseline policy through imitation and then refine and improve this policy through RL, allowing it to adapt to a wider range of scenarios and perform more effectively in real-world applications. Another advanced approach related to imitation learning is **IRL**. In IRL, instead of directly imitating the actions of the expert, the agent attempts to infer the underlying reward function that the expert is optimizing. Once this reward function is learned, the agent can then use traditional RL methods to discover an optimal policy. IRL is particularly useful when the goal is to understand the motivations behind expert behavior rather than just replicating it [13].

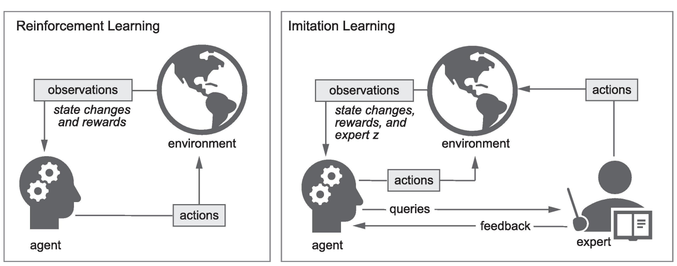

Fig. 2. Reinforcement Learning and Imitation Learning [13]

Each algorithm has its strengths and weaknesses, what are shown in table I.

RL Approach	Strengths	Weaknesses
SAC	Encourages exploration and improves policy robustness.	Higher computational demand.
	Performs well in continuous action spaces.	Complexity in implementation.
PPO	Balances exploration and exploitation effectively, leading to stable training.	Requires careful tuning of hyperparameters.
	Suitable for continuous action spaces.	May need large amounts of data for optimal performance.
Q Learning	Simplicity and ease of understanding.	Inefficient in high-dimensional or continuous action spaces.
	Effective in simpler environments with discrete action spaces.	Slower convergence compared to policy gradient methods.
DQN	Effective in high dimensional state spaces due to deep neural networks.	Struggles with continuous action spaces.
	Proven success in discrete action spaces.	Requires significant computational resources and extensive training time.
Actor-Critic Methods	Combine the benefits of value-based and policy-based methods.	Can be unstable if not properly tuned.
	Efficient learning through asynchronous updates or deterministic policies.	DDPG can suffer from overestimation bias.
Imitation Learning	Rapid learning from expert demonstrations.	Limited generalization to unseen situations.
	Reduces the need for extensive exploration.	Relies heavily on the quantity of expert data.

TABLE I
COMPARATIVE ANALYSIS OF DIFFERENT APPROACHES

III. Core Elements of Robotic Manipulation

This review delves deeply into the applications, techniques, and challenges associated with RL in robotic manipulation and grasping, providing comprehensive insights into this specialized area.

Robotic manipulation involves controlling robotic arms to interact with objects in the environment, including tasks such as picking, placing, and assembling. Grasping, a subset of manipulation, focuses on securely gripping and handling objects of various shapes and sizes. The grasping sequence can be divided into the following components:

Define Task: Specify the gripping task, including the objects to be grasped and the desired outcomes.

Perception: Collect data from sensors such as cameras, Light Detection and Ranging (LiDAR), or depth sensors. Object detection: Use computer vision techniques to identify and locate objects within the environment.

Grasp Planning: Determine potential grasp points and orientations using the data and evaluate the feasibility and stability of proposed grasps.

Execution: Send commands to the robotic arm to execute the selected action. Continuously monitor the grasping process and make real-time adjustments using feedback control strategies.

A. Task Specification

In the context of RL for robotic manipulation and grasping, task specification is a critical first step that lays the foundation for the entire research process. This subsection outlines the essential components and considerations involved in defining the task for an RL-based robotic gripping system. Recent advancements in Large Language Model (LLM) have significantly enhanced the process of translating high-level goals into detailed task specifications [14]. The ability of LLMs to transform natural language descriptions into structured task definitions, including state and action spaces, reward functions, and performance metrics, has been demonstrated in recent studies [15]. One another interesting research is a multi-object rearrangement in [16]. Multi-object rearrangement is a critical skill for service robots, often requiring commonsense reasoning. Achieving this demands knowledge about objects, which is difficult to impart to robots. LLMs can provide this knowledge but lacks an innate understanding of plausible physical arrangements [17].

B. Perception

Vision systems are crucial for enabling robotic grasping as they provide the necessary information for identifying, localizing, and interacting with objects. The integration of advanced vision systems with RL has significantly enhanced the capability of robots to perform autonomous grasping tasks.

Vision Language Model (VLM) is a new class of generative Artificial Intelligence (AI) models. VLM, powers visual AI agents capable of understanding natural language prompts and performing visual question answering [18]. These agents open up new applications across various industries, streamlining app development workflows and offering advanced perception capabilities like image/video summarization, interactive visual Q&A, and visual alerts [19].

These AI agents will be utilized in factories, warehouses, retail stores, airports, traffic intersections, and more, assisting operations teams in making better decisions through enhanced insights from natural interactions [20].

VLMs offer exciting applications in robotics. These models can enhance scene description capabilities, allowing robots to understand and interpret their surroundings more effectively. Here are some notable applications of VLMs in robotics:

- **Scene Description:** Robots equipped with VLMs can generate detailed descriptions of their environment, identifying objects, and their spatial relationships. For example, a robot could describe a cluttered room, noting the positions of furniture and other items.
- **Visual Question Answering:** Robots can respond to natural language questions about their environment. For instance, asking "Where is the blue book?" would prompt the robot to locate and describe the position of the book.
- **Enhanced Navigation:** VLMs allow robots to better navigate dynamic environments by interpreting visual cues and instructions. This can improve the efficiency of robots in settings like warehouses, where they need to move goods and avoid obstacles.

For **Object Detection**, there are Deep Learning techniques, like Convolutional Neural Network (CNN). Models such as You Only Look Once (YOLO) and Faster Region-based Convolutional Neural Network (R-CNN) enable real-time detection and localization of objects in the robot's environment [1] [21] [22] [23].

C. Grasp Planning and Control

Grasp planning is the process by which a robot determines the optimal way to approach, grip, and manipulate objects. It involves selecting the best grasp points and configuring the robot's gripper or hand to achieve a secure and stable hold on the object, then planning the path of the robotic arm to reach the target position. Effective grasp planning is essential for performing precise and reliable manipulation tasks.

1) Optimal Grasp Point Determination: Planning grasp points involves identifying the optimal positions on an object where a robot can apply its gripper or fingers to achieve a secure and stable hold.

Grasp Pose Estimation:

- Feature Extraction: Algorithms analyze the point cloud to extract relevant features, such as edges, surfaces, and contours, which are crucial for identifying potential grasp points.
- Pose Prediction: Using machine learning models, such as CNNs, the system predicts optimal grasp poses by analyzing the object's shape and orientation [24].

2) Path Planning: **Path planning** is a critical component of robotic navigation, enabling robots to determine optimal routes from their current location to a target destination while avoiding obstacles. Path planning involves calculating a feasible route for the robot to follow, considering factors such as the environment, obstacles, and the robot's kinematics. Effective path planning ensures safe, efficient, and reliable navigation in dynamic and unstructured environments.

By leveraging a combination of classical and learning-based path planning methods, robots can achieve highly effective and adaptive navigation capabilities, enabling them to operate autonomously in a wide range of settings. Future research will focus on improving real-time performance, adaptability, and integration with advanced perception systems to further enhance the capabilities of robotic path planning [13].

Hand-Eye Coordination Hand-eye coordination in robotic grasping involves synchronizing visual perception with motor actions to achieve precise manipulation tasks. This process relies on real-time visual data from cameras and sensors, which guide the robot's movements. Advanced computer vision algorithms, such as CNN, process visual inputs to identify optimal grasp points.

Control algorithms then adjust the robot's actions based on this visual feedback, ensuring accurate and stable grasps. Challenges in hand-eye coordination include minimizing latency and maintaining performance in dynamic environments [25].

D. Grasp Evaluation

Evaluation is a critical phase in the development of robotic grasping systems, ensuring that the theoretical models and planning algorithms translate effectively into practical, real-world applications. We can evaluate the grasp with tactile sensors [26].

- **Performance Metrics:** The effectiveness of robotic grasping is evaluated using various metrics, such as grasp success rate, robustness to disturbances, precision, and efficiency. These metrics provide quantitative measures of the system's performance.
- **Benchmarking:** Comparing the robot's performance against standard benchmarks and baselines helps in assessing its capabilities and identifying areas for improvement.
- **Comparative Analysis:** Evaluating different grasping algorithms and techniques under consistent conditions to determine their relative effectiveness and robustness.
- **Environmental Factors:** Testing the system's robustness under varied conditions, such as changes in lighting, object textures, external disturbances, ensures adaptability in diverse settings.

Incorporating these elements into the evaluation process helps refine robotic grasping systems, leading to more robust and effective designs.

IV. Simulated and Real-World Environments

A. Simulated Environments

Simulation plays a crucial role in developing and training RL models for robotic grasping. Simulated environments provide a safe and cost-effective platform to experiment with various RL algorithms and techniques before deploying them in the real world. **Purpose:** Simulations enable rapid prototyping and testing of RL models, allowing researchers to explore different strategies and optimize performance without the risks associated with physical hardware.

Simulation Tools:

- Gazebo
- PyBullet [27]
- MuJoCo
- NVIDIA Isaac Lab [28]

Training Process: In simulated environments, RL models are trained through iterative processes, where virtual robots interact with simulated objects to learn optimal grasping strategies. Synthetic data generated during these interactions is used to refine the models [29].

Benefits and Limitations: Simulation provides a controlled environment for extensive testing, reducing the costs and risks of real-world experiments. However, the reality gap—differences between simulated and real-world conditions—poses a challenge for transferring learned behaviors to physical robots.

B. Real-World Implementation

Transitioning from simulation to real-world implementation involves several techniques and considerations to ensure that RL models trained in virtual environments can perform effectively in real-world scenarios.

Transfer Learning: One of the primary challenges is bridging the gap between simulation and reality. Techniques such as domain adaptation and domain randomization help mitigate the reality gap by training models on a diverse range of simulated conditions, making them more robust to real-world variability.

Fine-Tuning with Real Data: After initial training in simulation, models are fine-tuned with real-world data to adapt to specific nuances and conditions not captured in simulations.

Real World Challenges : Real-world implementation faces challenges such as sensor noise, hardware limitations, and environmental unpredictability. These factors can affect the performance of RL models and require careful calibration and adaptation.

Performance Metrics: evaluating the performance of RL models in real-world scenarios involves metrics like grasp success rate, robustness to disturbances, and operational efficiency. These metrics provide insights into the effectiveness and reliability of the models under real-world conditions.

V. Challenges and Limitations

The application of RL in robotic grasping presents several challenges and limitations that must be addressed to achieve reliable and efficient performance. The section outlines the primary obstacles in this field.

A. Sim-to-Real Transfer

A major challenge is the transferability of models trained in simulation to real-world environments. The "reality gap" refers to the differences between the simulated environment and the real world, which can cause RL models to perform poorly when deployed in physical robots. Bridging this gap requires techniques such as domain randomization, which involves varying simulation parameters during training to make the model robust to real-world variations, and fine-tuning the models using real-world data. [5] [30]

B. Dynamic Environments:

Real-Time Adaptation: Robots must continuously adapt their grasp planning in real-time to account for moving objects, changing positions, and the presence of other dynamic elements like humans or other robots.

Environmental Variability: Dynamic environments can include variations in lighting, temperature, and other environmental factors that affect sensor accuracy and the robot's ability to perceive and interact with objects. The challenge of separating entangled workpieces in random bin picking, which involves object detection, separation planning, and gripper execution, was addressed in recent research [31]. Workpieces can sometimes be obscured by each other. One research focuses on solving this issue by pushing the parts to achieve collision-free object grasping [32].

Handling Object Instabilities: Objects may slip, roll, or exhibit other unexpected movements during grasping. The robot must detect these instabilities through tactile and visual feedback. Upon detecting slippage or rolling, the robot dynamically adjusts its grip strength, position, and orientation to re-establish a secure hold on the object.

C. Computational Demands

Training RL algorithms, especially those incorporating deep learning, requires considerable computational resources. This includes processing large datasets and running extensive simulations, which can be both time-consuming and costly. Advances in algorithm efficiency and the use of specialized hardware like GPUs are critical to manage these computational demands.

D. Data Requirements

Many RL algorithms require a large number of interactions with the environment to learn effective policies, which is impractical for real-world applications where data collection is expensive and time-intensive. Improving sample efficiency is crucial, and methods such as **experience replay** (storing and reusing past experiences) [33] and more **sophisticated exploration strategies** (to better balance exploration and exploitation) are being developed to address this issue. In this work [34], the RL model demonstrated the capability to execute tasks requiring precise movements, such as picking up an electric plug and inserting it into an electrical outlet. **Synthetic data generation techniques**, such as domain randomization, enable the creation of large datasets efficiently. Training these models can be time-consuming, typically requiring extensive computational resources and optimization. However, advancements in hardware and algorithms are continuously reducing the training time, enabling faster iterations and improvements in robotic grasping capabilities [3].

VI. Current Innovations and Research Directions

The field of RL in robotic grasping holds great promise for advancing autonomous systems.

A. Algorithmic Improvements

Continued research into developing more efficient and effective RL algorithms is critical. This includes:

- **Hybrid Approaches:** Combining model-free and model-based methods to leverage the strengths of both.
- **Hierarchical RL:** Breaking down tasks into smaller subtasks to simplify learning processes.
- **Exploration Strategies:** Developing smarter exploration techniques to improve learning efficiency and reduce the time required to train models.

[33] researched a method, which is called Highlight Experience Replay (HiER) and it creates a secondary highlight replay buffer for the most relevant experiences. Horvath et al. [33] validated the HiER and HiER+ on 8 tasks, what are the push, slide and pick-and-place tasks, the Gymnasium-Robotics Fetch benchmarks, and two mazes. HiER could significantly improve the state-of-the-art methods.

B. Enhanced Sim-to-Real Techniques

Improving the transfer of RL models from simulation to real-world applications is essential. Future work can focus on:

- **Domain Randomization:** Varying simulation parameters to make models more robust to real-world variations [35].
- **Fine-Tuning with Real Data:** Using limited real-world data to adjust models initially trained in simulation.
- **Sim-to-Real Transfer Learning:** Techniques that adapt models trained in simulation directly to the real world, reducing the dependency on extensive real-world training data.

C. Innovative Data Collection Methods

Efficient data collection is vital for training robust RL models. Innovative methods can include:

- **Crowdsourced Data:** Leveraging large-scale, diverse datasets collected from various sources.
- **Synthetic Data Generation:** Creating realistic synthetic data using generative models to augment training datasets [36].
- **Active Learning:** Allowing the RL agent to identify and request specific data that would be most beneficial for its learning process.

D. Multi Agent Cooperation

Multi-agent cooperation represents a promising future direction in robotic grasping, enabling multiple robots to work together seamlessly to accomplish complex tasks that would be challenging for a single robot to handle alone. Lan et al. [37] made a recent research about training a multi-robot multi agent system.

E. Human-Robot Collaboration

As robotic systems continue to advance, the potential for seamless collaboration between humans and robots is expanding, particularly in environments where both human dexterity and robotic precision are required. Future research in RL for robotic grasping could significantly enhance human-robot collaboration by enabling robots to better understand, predict, and adapt to human actions in real-time. One of the key areas of focus is developing RL models that can learn from human demonstrations in a dynamic, real-time context. This involves robots observing human actions and incorporating these observations into their own learning process to improve their ability to assist with or take over complex tasks. Moreover, ensuring safety and reliability in shared workspaces is paramount. RL can be employed to optimize robots' behavior to minimize risks and enhance safety when operating alongside humans. This includes learning to recognize and respond to non-verbal cues, such as gestures or body language, and adapting to sudden changes in the environment that may affect human safety [38]. Additionally, RL-driven human-robot collaboration could be applied in various industries, from manufacturing, where robots could assist with intricate assembly tasks, to healthcare, where robotic systems could support surgeons during complex procedures [39]. By refining the ability of robots to collaborate effectively with humans, RL has the potential to greatly expand the roles that robots can play in these critical areas.

F. Integration of Tactile Feedback

The integration of tactile feedback into robotic grasping systems represents a significant advancement in the field, enabling robots to perform more delicate and precise manipulation tasks. Tactile sensors, which mimic the sense of touch, provide critical data about the interaction between the robot and the objects it handles, such as pressure distribution, contact points, and slippage. This real-time feedback allows robots to adjust their grip dynamically, ensuring that objects are securely held without being damaged [40]. The development of more advanced soft grippers equipped with tactile sensors has further enhanced the capability of robotic systems to handle delicate and complex objects. By integrating tactile sensors into these soft grippers, robots can detect subtle signs of slipping or inadequate gripping in real-time. If the sensors identify that an object is beginning to slip, the robot can immediately adjust the grip pressure or reorient the gripper to prevent the object from falling [41]. Visual robotic grasping can be significantly enhanced with the addition of tactile feedback, as the combined sensory input allows robots to not only identify and approach objects accurately but also adjust their grip in real-time to ensure a secure and stable hold, even in unpredictable conditions.

VII. Conclusion

The integration of RL in robotic grasping represents a significant advancement in the field of autonomous systems. Throughout this review, we have explored various RL approaches, their strengths, and limitations, and identified critical challenges such as sim-to-real transfer, computational demands, sample efficiency, robustness, and safety. We also discussed future directions, including algorithmic improvements, enhanced sim-to-real techniques, and innovative data collection methods.

Advancements in these areas are essential for developing robust, efficient, and adaptable robotic systems capable of performing complex manipulation tasks in dynamic environments. Continued research and innovation will drive the next generation of autonomous robotic systems, making them more reliable and effective across diverse applications. The future of RL in robotic grasping is promising, with the potential to transform industries and enhance the capabilities of autonomous technologies.

Acknowledgment

Péter Galambos and Károly Széll thankfully acknowledge the financial support of this work by project no. 2019-1.3.1-KK-2019-00007 implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the 2019-1.3.1-KK funding scheme. Péter Galambos is a Bolyai Fellow of the Hungarian Academy of Sciences. Péter Galambos is supported by the UNKP-22-5 (Bolyai+) New National Excellence Program of the Ministry for Innovation and Technology from the source of the National Research, Development and Innovation Fund.

The authors acknowledge the use of ChatGPT, a language model developed by OpenAI, in an assistive and supervised manner for generating and refining content in this article. ChatGPT was utilized to draft and enhance sections such as the Abstract, Introduction, and Conclusion. The AI-generated content was carefully reviewed and edited by the authors to ensure accuracy and alignment with the research objectives.

References

[1] Muslikhin, J.-R. Horng, S.-Y. Yang, and M.-S. Wang, “Self-Correction for Eye-In-Hand Robotic Grasping Using Action Learning,” *IEEE Access*, vol. 9, pp. 156 422–156 436, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9622215

[2] A. Ghodake, P. Uttam, and B. B. Ahuja, “Accurate 6-DOF grasp pose detection in cluttered environments using Deep Learning,” in *2022 International Conference on Industry 4.0 Technology (I4Tech)*, 2022, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/9952955

[3] D. Horváth, K. Bocsi, G. Erdős, and Z. Istenes, “Sim2Real Grasp Pose Estimation for Adaptive Robotic Applications,” *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 5233–5239, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896323004676

[4] A. Wolf, D. Wolton, J. Trapl, J. Janda, S. Romeder-Finger, P. Galambos, and K. Széll, “Towards Robotic Laboratory Automation Plug & Play: The ”LAPP” Framework,” *SLAS TECHNOLOGY*, vol. 27, no. 1, pp. 18–25, 2022, number: 1 Publisher: Elsevier Inc. [Online]. Available: https://doi.org/10.1016%2Fj.slast.2021.11.003

[5] G. Erdos, I. Paniti, and B. Tipary, “Transformation of robotic workcells to digital twins,” *CIRP Annals*, vol. 69, 2020.

[6] M. Naeem, S. T. H. Rizvi, and A. Coronato, “A Gentle Introduction to Reinforcement Learning and its Application in Different Fields,” *IEEE Access*, vol. 8, pp. 209 320–209 344, 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9261348

[7] G. Qi and Y. Li, “Reinforcement Learning Control for Robot Arm Grasping Based on Improved DDPG,” in *2021 40th Chinese Control Conference (CCC)*, Jul. 2021, pp. 4132–4137, iSSN: 1934-1768. [Online]. Available: https://ieeexplore.ieee.org/document/9550413

[8] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor,” Aug. 2018, arXiv:1801.01290 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1801.01290

[9] Y.-L. Chen, Y.-R. Cai, and M.-Y. Cheng, “Vision-Based Robotic Object Grasping—A Deep Reinforcement Learning Approach,” *Machines*, vol. 11, no. 2, p. 275, 2023. [Online]. Available: https://www.mdpi.com/2075-1702/11/2/275

[10] S. Yang and Q. Wang, “Robotic Arm Motion Planning with Autonomous Obstacle Avoidance Based on Deep Reinforcement Learning,” in *2022 41st Chinese Control Conference (CCC)*, 2022, pp. 3692–3697. [Online]. Available: https://ieeexplore.ieee.org/document/9902722

[11] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, “Deep Reinforcement Learning: A Brief Survey,” *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017. [Online]. Available: https://ieeexplore.ieee.org/document/8103164

[12] W. Liu, H. Niu, M. N. Mahyuddin, G. Herrmann, and J. Carrasco, “A Model-free Deep Reinforcement Learning Approach for Robotic Manipulators Path Planning,” in *2021 21st International Conference on Control, Automation and Systems (ICCAS)*, 2021, pp. 512–517. [Online]. Available: https://ieeexplore.ieee.org/document/9649802

[13] J. Manuel Davila Delgado and L. Oyedele, “Robotics in construction: A critical review of the reinforcement learning and imitation learning paradigms,” *Advanced Engineering Informatics*, vol. 54, p. 101787, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474034622002452

[14] I. Singh, V. Blukis, A. Mousavian, A. Goyal, D. Xu, J. Tremblay, D. Fox, J. Thomason, and A. Garg, “ProgPrompt: Generating Situated Robot Task Plans using Large Language Models,” 2022. [Online]. Available: http://arxiv.org/abs/2209.11302

[15] H. Liu, Y. Zhu, K. Kato, A. Tsukahara, I. Kondo, T. Aoyama, and Y. Hasegawa, "Enhancing the LLM-Based Robot Manipulation Through Human-Robot Collaboration," *IEEE Robotics and Automation Letters*, vol. 9, no. 8, pp. 6904–6911, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10561501

[16] Y. Ding, X. Zhang, C. Paxton, and S. Zhang, "Task and Motion Planning with Large Language Models for Object Rearrangement," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 2086–2092. [Online]. Available: https://ieeexplore.ieee.org/document/10342169

[17] "Sequence-Based Plan Feasibility Prediction for Efficient Task and Motion Planning \textbar Research." [Online]. Available: https://research.nvidia.com/publication/2023-07_sequence-based-plan-feasibility-prediction-efficient-task-and-motion-planning

[18] F. Bordes, R. Y. Pang, A. Ajay, A. C. Li, A. Bardes, S. Petryk, O. Mañas, Z. Lin, A. Mahmoud, B. Jayaraman, M. Ibrahim, M. Hall, Y. Xiong, J. Lebensold, C. Ross, S. Jayakumar, C. Guo, D. Bouchacourt, H. Al-Tahan, K. Padthe, V. Sharma, H. Xu, X. E. Tan, M. Richards, S. Lavoie, P. Astolfi, R. A. Hemmat, J. Chen, K. Tirumala, R. Assouel, M. Moayeri, A. Talattof, K. Chaudhuri, Z. Liu, X. Chen, Q. Garrido, K. Ullrich, A. Agrawal, K. Saenko, A. Celikyilmaz, and V. Chandra, "An Introduction to Vision-Language Modeling," 2024. [Online]. Available: http://arxiv.org/abs/2405.17247

[19] W. Yuan, J. Duan, V. Blukis, W. Pumacay, R. Krishna, A. Murali, A. Mousavian, and D. Fox, "RoboPoint: A Vision-Language Model for Spatial Affordance Prediction for Robotics," Jun. 2024, arXiv:2406.10721 [cs]. [Online]. Available: http://arxiv.org/abs/2406.10721

[20] "Build VLM-Powered Visual AI Agents Using NVIDIA NIM and NVIDIA VIA Microservices," 2024, publication Title: NVIDIA Technical Blog. [Online]. Available: https://developer.nvidia.com/blog/build-vlm-powered-visual-ai-agents-using-nvidia-nim-and-nvidia-via-microservices/

[21] D. Horváth, G. Erdős, Z. Istenes, T. Horváth, and S. Földi, "Object Detection Using Sim2Real Domain Randomization for Robotic Applications," *IEEE Transactions on Robotics*, vol. 39, no. 2, pp. 1225–1243, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/9916581

[22] "Language Guided Robotic Grasping with Fine-Grained Instructions \textbar IEEE Conference Publication \textbar IEEE Xplore." [Online]. Available: https://ieeexplore.ieee.org/document/10342331

[23] V. Tadic, A. Odry, Z. Vizvari, Z. Kiraly, I. Felde, and P. Odry, "Electric Vehicle Charging Socket Detection using YOLOv8s Model," *Acta Polytechnica Hungarica*, vol. 21, no. 10, pp. 121–139, 2024. [Online]. Available: https://acta.uni-obuda.hu/Tadic_Odry_Vizvari_Kiraly_Felde_Odry_150.pdf

[24] S. Kumra and C. Kanan, "Robotic grasp detection using deep convolutional neural networks," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 769–776. [Online]. Available: https://ieeexplore.ieee.org/document/8202237

[25] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 421–436, 2018. [Online]. Available: https://doi.org/10.1177/0278364917710318

[26] E. Y. Puang, Z. Li, C. M. Chew, S. Luo, and Y. Wu, "Learning Stable Robot Grasping with Transformer-based Tactile Control Policies," 2024. [Online]. Available: http://arxiv.org/abs/2407.21172

[27] R. Michalík and A. Janota, "The PyBullet module-based approach to control the collaborative YuMi robot," in *2020 ELEKTRO*, 2020, pp. 1–4. [Online]. Available: https://ieeexplore.ieee.org/document/9130233

[28] "Closing the Sim-to-Real Gap: Training Spot Quadruped Locomotion with NVIDIA Isaac Lab," 2024, publication Title: NVIDIA Technical Blog. [Online]. Available: https://developer.nvidia.com/blog/closing-the-sim-to-real-gap-training-spot-quadruped-locomotion-with-nvidia-isaac-lab/

[29] K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige, S. Levine, and V. Vanhoucke, "Using Simulation and Domain Adaptation to Improve Efficiency of Deep Robotic Grasping," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 4243–4250. [Online]. Available: https://ieeexplore.ieee.org/document/8460875

[30] E. Salvato, G. Fenu, E. Medvet, and F. A. Pellegrino, "Crossing the Reality Gap: A Survey on Sim-to-Real Transferability of Robot Controllers in Reinforcement Learning," *IEEE Access*, vol. 9, pp. 153 171–153 187, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9606868

[31] M. Moosmann, M. Kulig, F. Spenrath, M. Mönnig, S. Roggendorf, O. Petrovic, R. Bormann, and M. F. Huber, "Separating Entangled Workpieces in Random Bin Picking using Deep Reinforcement Learning," *Procedia CIRP*, vol. 104, pp. 881–886, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827121010465

[32] M. Mosbach and S. Behnke, "Grasp Anything: Combining Teacher-Augmented Policy Gradient Learning with Instance Segmentation to Grasp Arbitrary Objects," 2024. [Online]. Available: http://arxiv.org/abs/2403.10187

[33] D. Horváth, J. Bujalance Martín, F. Gàbor Erdos, Z. Istenes, and F. Moutarde, "HiER: Highlight Experience Replay for Boosting Off-Policy Reinforcement Learning Agents," *IEEE Access*, vol. 12, pp. 100 102–100 119, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10595054/references#references

[34] "RVT-2: Learning Precise Manipulation from Few Examples \textbar Research." [Online]. Available: https://research.nvidia.com/index.php/publication/2024-07_rvt-2-learning-precise-manipulation-few-examples

[35] M. Ma, H. Li, G. Hu, S. Liu, and D. Zhao, "Comparison of Different Domain Randomization Methods for Policy Transfer in Reinforcement Learning," in *2023 IEEE 12th Data Driven Control and Learning Systems Conference (DDCLS)*, 2023, pp. 1818–1823, iSSN: 2767-9861. [Online]. Available: https://ieeexplore.ieee.org/document/10166417

[36] D. Lee, W. Chen, and N. Rojas, "Synthetic data enables faster annotation and robust segmentation for multi-object grasping in clutter," in *2024 10th International Conference on Mechatronics and Robotics Engineering (ICMRE)*, 2024, pp. 253–259. [Online]. Available: https://ieeexplore.ieee.org/document/10532168

[37] X. Lan, Y. Qiao, and B. Lee, "Application of Multi Agent Reinforcement Learning to Robotic Pick and Place System," in *2024 10th International Conference on Automation, Robotics and Applications (ICARA)*, 2024, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/10552987

[38] M. Dobeš and N. Sabolová, "Emotion Recognition Using Pretrained Deep Neural Networks," *Acta Polytechnica Hungarica*, vol. 20, no. 4, pp. 195–204, 2023. [Online]. Available: http://acta.uni-obuda.hu/Dobes_Sabolova_133.pdf

[39] E. Lukács, R. Levendovics, and T. Haidegger, "Enhancing Skill Assessment of Autonomous Robot-Assisted Minimally Invasive Surgery: A Comprehensive Analysis of Global and Gesture-Level Techniques applied on the JIGSAWS Dataset," *Acta Polytechnica Hungarica*, vol. 20, no. 8, pp. 133–153, 2023. [Online]. Available: https://acta.uni-obuda.hu/Lukacs_Levendovics_Haidegger_137.pdf

[40] W. Liang, F. Fang, C. Acar, W. Q. Toh, Y. Sun, Q. Xu, and Y. Wu, "Visuo-Tactile Feedback-Based Robot Manipulation for Object Packing," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 1151–1158, 2023, conference Name: IEEE Robotics and Automation Letters. [Online]. Available: https://ieeexplore.ieee.org/document/10016641

[41] Z. Zhou, R. Zuo, B. Ying, J. Zhu, Y. Wang, X. Wang, and X. Liu, "A Sensory Soft Robotic Gripper Capable of Learning-Based Object Recognition and Force-Controlled Grasping," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 1, pp. 844–854, 2024, conference Name: IEEE Transactions on Automation Science and Engineering. [Online]. Available: https://ieeexplore.ieee.org/document/9999176