STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
*Stat., Optim. Inf. Comput.*, Vol. **12**, March 2024, pp 571–601.
Published online in International Academic Press (www.IAPress.org)

# Review of Reinforcement Learning for Robotic Grasping: Analysis and Recommendations

Hiba Sekkat [1,*], Oumaima Moutik [1], Loubna Ourabah [1], Badr ElKari [1], Yassine Chaibi [2], Taha Ait Tchakoucht [1]

[1] *EIDIA, Euromed Research Center, Euro-Med University(UEMF),Fez, Morocco*
[2] *SIAME Laboratory, UPPA, Pau, France*

**Abstract** This review paper provides a comprehensive analysis of over 100 research papers focused on the challenges of robotic grasping and the effectiveness of various machine learning techniques, particularly those utilizing Deep Neural Networks (DNNs) and Reinforcement Learning (RL). The objective of this review is to simplify the research process for others by gathering different forms of Deep Reinforcement Learning (DRL) grasping tasks in one place. Through a thorough analysis of the literature, the study emphasizes the critical nature of grasping for robots and how DRL techniques, particularly the Soft-Actor-Critic (SAC) strategy, have demonstrated high efficiency in handling the task. The results of this study hold significant implications for the development of more advanced and efficient grasping systems for robots. Continued research in this area is crucial to further enhance the capabilities of robots in handling complex and challenging tasks, such as grasping.

**Keywords** Reinforcement Learning (RL), Deep Reinforcement Learning (DRL), Grasp Task, Soft-Actor-Critic (SAC)

**DOI:** 10.19139/soic-2310-5070-1797

## 1. Introduction

Artificial Intelligence (AI) is a source of both excitement and apprehension. In general, it is an overarching concept that refers to computer systems that are able to perceive their environment, reason, learn, and manage data so that they can act according to what they perceive and their goals. With that, AI has recently caused a shift in many industries around the world, from technology to healthcare [1], [2], [3]. This once mysterious field has become a hot topic teasing countless industrial and academic minds. Drastic advances in hardware and data storage, coupled with AI's ability to "self-learn", have put it at the forefront of algorithms for multiple applications such as computer vision [4], [5], [6] and natural language processing [7], [8]. AI uses different forms today whether it is digital assistants, chat-bots [9] or Machine Learning and at the time being, the most prominent topic in AI is machine learning (ML) [10], [11], [12], [13].

There are two primary applications for machine learning techniques, namely classification and regression. Within the realm of these applications, the exploration of machine learning's diverse capabilities is evident in

*Correspondence to: Hiba Sekkat (Email: h.sekkat@ueuromed.org). EIDIA, Euromed Research Center, Euro-Med University(UEMF),Fez, Morocco.

ISSN 2310-5070 (online) ISSN 2311-004X (print)

recent studies. Kyrarini et al. investigated the realm of robot learning for assistive manipulation tasks through a head gesture-based interface [14]. Their novel approach introduced a hands-free robot control system, leveraging optical flow for feature extraction and support vector machines for head gesture recognition. Similarly, Bahrami et al. delved into machine learning for touch localization on ultrasonic wave touchscreens [15]. Employing a robotic finger to simulate touch actions, they captured data for model training. This technique finds applications in classification, clusterization, regression tasks, as well as time series analysis, anomaly detection, and adaptive (robotic) control. Shafiei et al. contributed to the field by developing machine learning classification models that utilize electroencephalogram (EEG) and eye-gaze features [16]. Their objective was to predict the level of surgical expertise in robot-assisted surgery (RAS). In a different context, Kolaghassi et al. conducted a systematic review focusing on intelligent algorithms in gait analysis and prediction for lower limb robotic systems [17]. Notably, $33.3\%$ of the included papers implemented regression models for the estimation and prediction of kinematic and kinetic parameters in gait analysis. Additionally, machine learning algorithms can be categorized into four primary sub-fields: Supervised Learning [18] [19], Semi-Supervised Learning [20] [21], Unsupervised Learning [22], and Reinforcement Learning [23].

The integration of AI and ML is currently a popular and significant subject, with potential benefits when applied in the field of robotics. Many researchers have explored this combination, particularly in the area of deep learning (DL). For instance, Bai et al. have developed an innovative garbage collection robot that implements a deep neural network to recognize and pick up garbage with high precision and autonomy [24].DL was employed by Kase et al. to enable a humanoid robot to perform a Put-In-Box task that consists of several separate tasks [25]. DL techniques were utilized by Gu et al. to introduce a robot designed for collecting tennis balls [26]. To teach a parallel plate gripper how to recognize the grasping configurations of different household items, Caldera et al. suggested the application of a transfer learning approach that involves deep convolutional neural networks[27].Onishi et al. developed a robot for automated fruit harvesting by leveraging DL techniques [28]. Kim et al. utilized a DL approach that involved transferring knowledge between different robots to teach a robot how to perform two different cleaning tasks on a table [29]. In an effort to improve the ability of robots to manipulate objects, Yang et al. investigated a DL approach for grasping objects that are initially invisible, specifically,to enable a robot to grasp the target object, a sequence of pushing and grasping actions is involved in the proposed method [30]. Shang et al. developed a DL technique that employs dexterous hands to grasp new objects [31] .

Reinforcement learning (RL) is a ML technique that has shown great potential in robotics, particularly in object grasping [32]. RL is considered the algorithm of choice for building truly intelligent robots [33]. In this comprehensive review, we delve into the current state-of-the-art RL algorithms, encompassing their methods, types, and potential applications in the domain of robotic grasping. The emphasis of this study lies in exploring the practical applications of RL in robotic grasping scenarios, steering away from intricate mathematical proofs and numerical analyses of RL approaches. Instead, we aim to provide readers with a panoramic view of the evolving landscape, summarizing the history and progression of RL from its early foundations to recent advances. Our motivation is rooted in the recognition of the critical role robotic grasping plays in various applications. To streamline the research process, we meticulously analyzed over 100 research papers, with a particular focus on the effectiveness of machine learning techniques, including Deep Neural Networks (DNNs) and Reinforcement Learning (RL). Our main contribution lies in synthesizing this extensive literature to spotlight the diverse forms of Deep Reinforcement Learning (DRL) grasping tasks and underscore the efficacy of the Soft-Actor-Critic (SAC) strategy within DRL techniques. As we progress through this review, we bridge the explored concepts with practical

applications in robotic grasping, adding an intuitive layer to enhance comprehension in Section 3. In the same section, we provide detailed explanations and address open problems, particularly focusing on the most prominent algorithms in robotic grasping — DDPG, TD3, and SAC. Their comparative analysis in diverse state-of-the-art applications unfolds in Section 4. To conclude, Section 5 offers a comprehensive summary, shedding light on both the benefits and drawbacks of RL in the specific context of robotic grasping.

## 2. Reinforcement Learning

### *2.1. Brief history*

The history of reinforcement learning is founded on two important areas of research that were independently pursued before intertwining into modern reinforcement learning: animal psychology and optimal control. The psychology of animal learning was the impetus for the idea of trial-and-error learning. The trial and error theory of learning was first introduced by the famous psychologist Edward L. Thorndike [34], this procedure was implemented in some of the early works in artificial intelligence and resulted in the renaissance of reinforcement learning in the early 1980s. The optimal control problem was composed originally to design a controller to minimize the loss function of a dynamic system over time [35]. In the mid-1950s, more exactly in 1957, An innovative perspective on Hamilton-Jacobi theory was introduced by Richard Bellman, who also devised an approach to address the optimal control problem known as dynamic programming [36]. Other methods are emerging and are combined with the two previously mentioned areas in the late 1980s. These methods are the temporal difference approaches and this union between the three domains gave rise to the modern field of RL. [37] give a lot more details about Reinforcement Learning history. To sum it all up, RL is a type of ML that allows an agent to learn how to reach a goal based on trial and error. This concept also named the *Law of Effect* aims to enable the agent to test actions and receive feedback (reinforcement). RL involves adjusting the behavior of the agent to maximize the cumulative reward it receives, and it has broad applications in solving control and optimization problems that entail sequential decision-making. Consequently, it is a subject of great interest. The bar chart in **Figure 1** illustrates this concern in the percentage of published papers on this hot topic from 2014 to date.

### *2.2. Methods of Reinforcement Learning*

To estimate value functions and action-value functions, as illustrated in **Figure 2**, there are three main families of algorithms used in RL: Dynamic Programming (DP), Monte Carlo (MC) and Temporal difference (TD). **Dynamic programming (DP)** methods aim to find the optimal policy, but they require an ideal system model and computational constraints for non-tangible tasks are unfeasible [38]. Policy iteration and value iteration are commonly used DP methods, with policy iteration seeking the optimal policy through iterative policy improvement and evaluation [39]. However, it is rarely used due to its large computational cost. In contrast, value iteration determines the optimal policy by identifying the optimal value functions, which is more efficient since it does not evaluate many policies [40]. However, a perfect model of the system is required to extract the optimal policy using the optimal value function. **Monte Carlo (MC)** methods are model-free [41] and rely on sampling to estimate mean returns for various policies by taking samples of state sequences, actions, and rewards under the policy. Since the agent doesn't have a model of the system, it determines the value functions for each action through exploration and

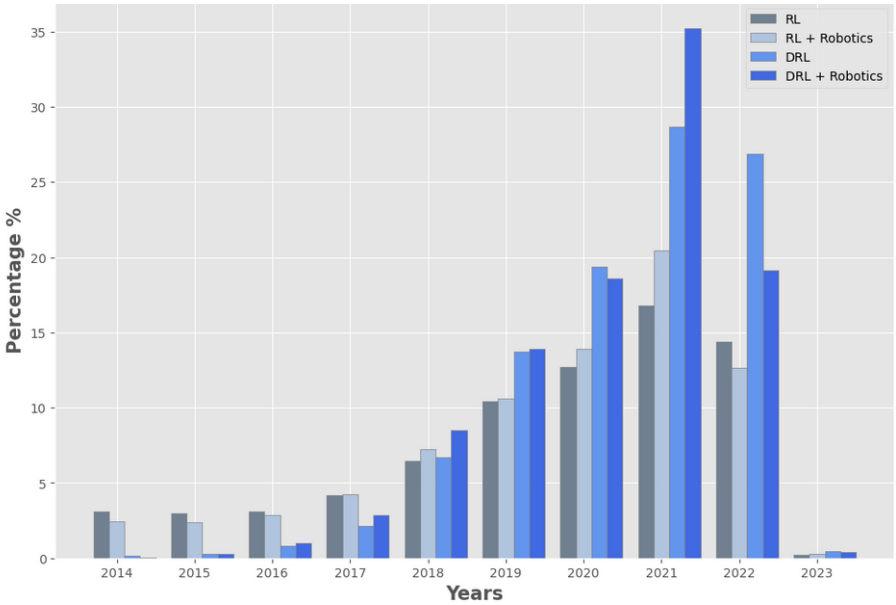


Figure 1. Percentage of publications in RL, RL+Robotics, DRL, DRL+Robotics in the last 10 years based on WOS database

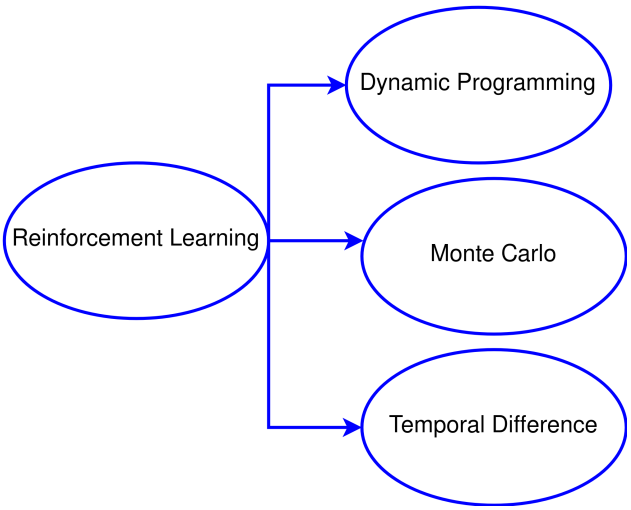


Figure 2. RL algorithm families

deduces the optimal policy. MC methods define action-value functions since value functions alone are insufficient without a model to switch to great value states. However, MC methods require waiting until the end of the episode before learning can begin, which is penalizing in long or continuous systems. an episode before updating the value functions, unlike MC methods [42] **Temporal Difference (TD)** methods combine both DP and MC by using the ideas of both methods to update the value functions incrementally. TD methods do not require waiting until the end of the episode before learning can begin, which is penalizing in long or continuous systems. Since the main goal of AI is to replicate human behavior, neither DP nor MC alone is sufficient, and TD methods are often used instead [42].

### *2.3. Types of Reinforcement Learning*

It is tricky to present an exhaustive and detailed list of all the RL algorithms applied to robot manipulation.The focus of this discussion will be limited to the major branches of algorithms, which include model-based and model-free algorithms, as well as policy-based and value-based algorithms. **Figure 3** represents a complete list of algorithms.

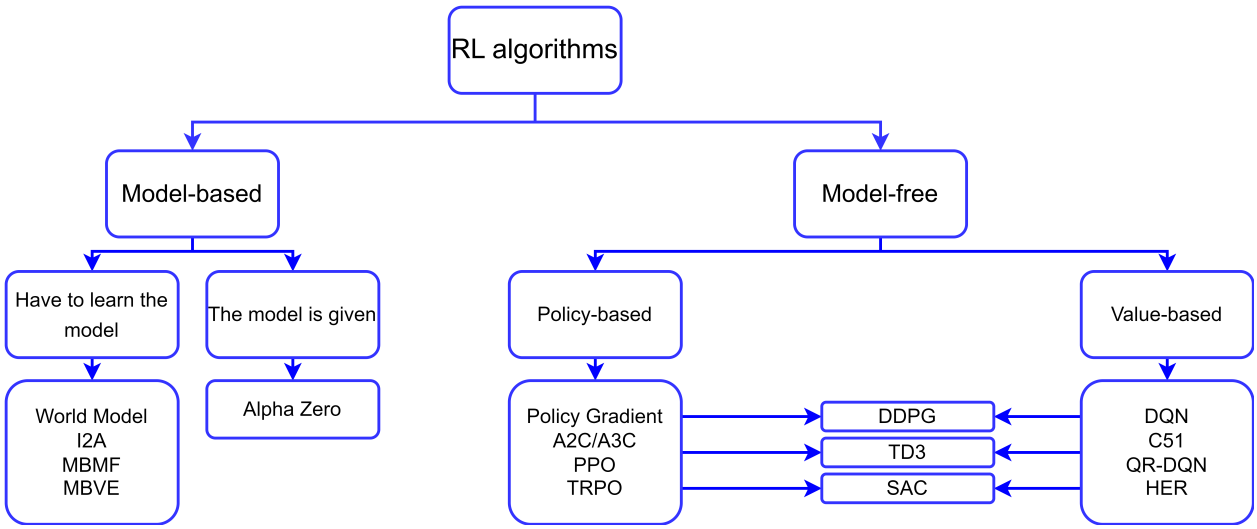


Figure 3. Reinforcement Learning algorithms

*2.3.1. Model-Based and Model-Free* RL algorithms can be categorized into two groups depending on whether the agent has knowledge of the environment's model or is learning it [43]. If the agent has the model, it can predict its performance when taking a particular action, resulting in improved sample efficiency compared to model-free methods. However, learning the model can introduce bias, leading to sub-optimal behavior in the real environment.In contrast, model-free techniques rely on reward signals and learn value functions solely from the agent's interactions with the environment. They are easier to implement and adjust for hyper-parameters, making them more popular than model-based methods.

*2.3.2. Policy-Based and Value-Based* In *value-based* algorithms, the estimation of the action-value function is carried out in reference to the optimal value $Q^*(s, a)$. This is usually achieved through off-policy learning, as detailed in the preceding chapter. Conversely, *policy-based* approaches identify the optimal action to take at a given state (s) to maximize the reward. This is often done on-policy.Nguyen et al. [33] contend that policy-based techniques are more dependable and consistent than Q-learning methods, which estimate Q indirectly based on an objective function. However, policy-based methods may fail due to various factors. Despite this, policy-based algorithms exhibit higher sample efficiency by effectively reusing data, a crucial factor for their successful implementation on real robots.

In the realm of reinforcement learning, the dichotomy between policy-based and value-based algorithms has long been a focal point of research. While policy-based methods focus on defining the optimal behavior directly, and value-based methods estimate the value of different actions in a given state, recent advancements have extended these paradigms. A noteworthy addition to this landscape is Fuzzy Reinforcement Learning, which

introduces a nuanced approach to decision-making in uncertain environments [44]. Unlike traditional methods that rely on precise values and policies, fuzzy reinforcement learning leverages fuzzy logic to navigate the inherent uncertainties of real-world scenarios [45]. This integration of fuzziness into the learning process not only offers a more adaptive and flexible approach but also aligns well with the challenges posed by complex and dynamic robotic grasping tasks [46]. Moreover, the evolution of reinforcement learning has seen the emergence of Reverse Reinforcement Learning (Reverse RL), where the focus shifts from learning optimal behavior to inferring the underlying reward structure from observed behavior [47]. This approach holds promise in scenarios where defining a reward function is challenging or impractical, contributing a unique perspective to the exploration. Additionally, Adversarial Deep Reinforcement Learning (Adversarial DRL) has garnered attention for its ability to address complex tasks, such as robotic cloth manipulation, without explicit reward function design [48]. By introducing adversarial elements into the learning process, this technique allows agents to learn near-optimal behaviors through expert demonstration and self-exploration. The interplay between agent and environment takes on a dynamic and adversarial nature, further enhancing the adaptability of reinforcement learning methodologies [49]. As we navigate the landscape of these advanced reinforcement learning paradigms, offering a comprehensive understanding of their applications and impact on robotic grasping tasks, our focus turns to delve specifically into Deep Reinforcement Learning (DRL). In the subsequent sections, we will explore the nuances of DRL techniques, their implementation in robotic grasping scenarios, and the state-of-the-art advancements in this exciting intersection of machine learning and robotics.

## 3. Deep Reinforcement Learning for robotic grasping

DL is an excellent tool for processing unstructured environments due to its ability to learn from vast amounts of data and identify patterns. However, while this aspect is crucial for recognition, it is not equivalent to decision-making. RL, on the other hand, facilitates decision-making, making it an indispensable feature. And since robotic tasks, more precisely robotic grasping tasks, require an interaction between the agent and the environment so merging between DL and RL (DRL) is very crucial to the improvement of robotic tasks. As cited by haarnoja et al. empirical evidence suggests that model-free DRL is highly effective in various domains, including video games, as well as simulated robotic manipulation and locomotion [50]. Ibarz et al. discussed the successful application of DRL techniques in various tasks, including quadrupedal walking, grasping unfamiliar objects, and acquiring a diverse set of intricate manipulation skills. [51], these case studies demonstrate that DRL is a feasible approach for learning directly in the real world, using raw sensory inputs, and tackling physically challenging tasks like dexterous manipulation and walking. The aforementioned research highlights that policies learned through DRL exhibit effective generalization, as seen in the case of robotic grasping. But before talking about robotic grasping. Section 3.1 will discuss, generally, different strategies used to learn robotic grasping while section 3.2, will focus more on the three main algorithms of DRL used nowadays for robotic manipulation, specifically, for the robotic grasping.

### *3.1. Robotic Grasping*

As mentioned before, robotic grasping is a very challenging and interesting task, that's why many reviews were conducted in that context. Here, we're going to cite the latest reviews on the application of DRL in the grasp

task. In 2017,provided a concise overview of DRL, with a particular emphasis on the primary algorithms used in this field [52] . In 2018, Khanzhahi et al. conducted a review and classification of DRL algorithms, highlighting their advantages and limitations, as well as discussing the challenges that DRL has successfully overcome [53]. Mousavi et al. conducted a review of fundamental DRL algorithms, with a focus on research methodology [54]. In 2019, Chatzilygeroudis et al. described a method for robots to acquire learning using micro-data reinforcement learning [55]. Bhagat et al reviewed DRL based intelligent soft robotics .[56]. In 2021, Du et al. conducted a comprehensive thorough study on vision-based robotic grasping, which identified the primary tasks required for successful vision-based robotic grasping: object localization, grasp estimation, and object pose estimation [57]. Connolly et al. examined the accuracy and realism of models generated by two simulation platforms for simple robotic grasping tasks [58]. The goal of this review was to investigate the extent to which the resulting models could accurately represent reality. Marwan et al. conducted an extensive review of various research approaches, where within the past five years, various techniques such as sensing, learning, and gripping have been utilized. The review covered a range of topics in these areas [59]. In 2022, Wang et al. classified DRL algorithms and their applications, and conducted a thorough evaluation of the current DRL methods [60].

*3.1.1. Grasping in cluttered environments* Robotic grasping is an important aspect of robotics and automation. It involves the ability of robots to manipulate objects using different mechanisms, such as suction grasping, only grasping, synergies prehensile and non-prehensile, or multi-functional grippers. In recent years, learning robotic grasping policies using DRL has gained significant attention as a promising approach. A comprehensive overview of current research in the field, with a focus on studies related to grasping objects using different mechanisms relying on DRL methods, will be provided in this subtopic. In 2017, Mahler et al. presented a robot bin-picking system that uses grasping only, by fine-tuning a Convolutional Neural Network (CNN) for grasp quality using Dex-Net. Using this approach, the robot was able to achieve high success rates in picking and placing objects from a bin [61]. In 2018, Morrison et al. introduced Generative Grasping CNN (GG-CNN), a real-time generative grasp synthesis method that uses a CNN to generate grasp candidates for an object [62]. It was demonstrated that this method achieved better results in terms of both success rates and execution time compared to other state-of-the-art methods. Zeng et al. showed that model-free deep reinforcement learning is capable of learning these synergies from the ground up [63]. Integrated grippers that combine different types of gripping mechanisms have been developed to enable grasping diverse objects in various operational settings.

Silver et al. introduced a pushing and pick-and-place method using Deep Deterministic Policy Gradient (DDPG) and Actor-Critic method [64]. Their approach was able to learn to push objects into graspable configurations and pick them up with a gripper. In 2019,Kang et al. introduced an integrated gripper that merges a suction gripping system with a linkage-driven underactuated gripper [65]. It may be necessary to perform pre-grasping manipulation such as shifting or pushing an object, and algorithms have been developed to learn these additional manipulation tasks. Berscheid et al. developed an algorithm that can learn how to shift objects to increase their grasp probability[66]. Semantic grasping methods have also been developed to estimate the 6DOF pose for grasping by robotic manipulators. Zhu et al. introduced a method for robotic semantic grasping that enables estimation of the 6DOF grasping pose for a robotic manipulator, thereby allowing for a perpendicular grip to be made on the object's surface [67]. Murali et al. introduced, using partial point cloud observations, a technique that generates a strategy for grasping in 6-DOF for any target object in a cluttered environment [68]. Shao et al. proposed a suction grasping method using Q-Learning and ResNet with U-net (CNN) [69]. Their approach was able to learn to grasp

objects with suction cups, achieving high success rates in cluttered environments. In 2020, Sarantopoulos et al. presented a pushing and grasping method using Deep Q-Learning (DQN) [70]. Their approach was able to learn to push objects into graspable configurations and grasp them with a gripper.Wu et al. introduced a generative attention learning framework that utilizes a single depth image and circumvents continuous motor control to achieve high-performance multi-fingered grasping in clutter [71]. The approach developed by Wu et al. successfully enabled learning of multi-fingered grasping in cluttered settings, allowing for the grasping of objects with several fingers. A method based on DRL and visio-motor feedback was introduced by Joshi et al. [72] to address the issue of robotic grasping. Their approach was able to learn to grasp objects by taking into account both visual and motor information. Kim et al. developed a deep learning-based approach for grasping diverse unseen target objects in a cluttered environment [73]. Pose estimation of textureless and textured objects is an important aspect of robotic grasping. A push-grasping policy was learned for grasping a particular object in clutter by Xu et al. in 2021 [74], utilizing a hierarchical RL framework based on goal-directed conditioning that exhibits efficient utilization of samples. Their approach was able to learn to push objects into graspable configurations and grasp them with a gripper. Tang et al. developed a self-supervised approach to train a robot in joint planar pushing and 6-DoF grasping policies [75]. They used two distinct deep neural networks that were trained to map from 3D visual observations to actions, with the aid of a Q-learning framework. Their approach was able to learn to push objects into graspable configurations and grasp them with a 6-DoF. Dong et al. proposed an innovative method for estimating the position/orientation of objects with and without textures by leveraging the objects colors as a crucial feature for object recognition, particularly for grasping tasks [76].Teaching a robot to identify a desired object by utilizing its color as a distinctive feature and then locate it and picking it up in an unsupervised manner is another approach that has been investigated. Mohammed et al. developed a method for training a robot to locate and pick up objects based on their color [77]. Finally, Sundermeyer et al. proposed an end-to-end network that generates a probability distribution of parallel-jaw grasps with 6-DoF efficiently, using only depth recordings of a scene, enabling efficient grasping of objects in cluttered environments [78].

In order to offer a thorough overview of the cutting-edge state-of-the-art accomplishments and upcoming challenges in this area, presented in **Table 1** are the latest research papers on grasping in cluttered environments.

*3.1.2. Simulation-to-real-world transfer* The field of robotic grasping has a high demand for transfer learning from simulation to reality. It is important to first conduct simulations in order to fully comprehend the training environment. One of the major challenges for robots is to learn the skills necessary to adjust to the properties of grasped objects. Numerous studies have explored this area. James et al. presented a method called Randomized-to-Canonical Adaptation Networks (RCANs) which addresses the issue of the visual reality gap without relying on real-world data [83]. Wu et al. proposed an attention mechanism that improves the success rate of grasping objects in cluttered environments by mapping pixel space to Cartesian space [84]. Fang et al. suggested a framework that combines planning and learning for efficient exploration in complex environments [85], while Irpan et al. examined the problem of model selection for DRL in real-world settings [86]. Wu et al. presented a tactile closed-loop method called MAT, enabling the robot to seize the object even when the hand's initial location is coarse [87]. Shao et al. introduced UniGrasp, a method for generating grasping motions that takes into account the geometry of the object and the attributes of the gripper [88]. RL has also been used to acquire skillful in-hand manipulation policies for reorienting objects on a Shadow Dexterous Hand in the physical world, as shown by Andrychowicz et al. [89], and to enable a robot to perform robust object pushing through training, as explored by Clavera et al. [90]. Rao

Table 1. Latest Research Papers on Grasping in Cluttered Environments: A Summary of State-of-the-Art Achievements and Future Challenges

| Year | Application | Robot | Gripper | Learning algorithm | Highlight | Achievement | Future challenges | Reference |
|---|---|---|---|---|---|---|---|---|
| 2019 | Push and Pick | UR5 | A combination of the suction cup and two finger gripper | Deep Q-Network (DQN) | A new system for robotics that can automatically pick up objects in scenes that are cluttered. | • The suction cup together with the two-finger gripper for grasping is more efficient.<br>• The active exploration strategy shows superior performance compared to methods with only a static affordance map. | • Improve the system's robustness and adaptability to a wider range of object shapes and sizes.<br>• Optimize the system's speed and accuracy especially in real-world scenarios. | [79] |
| 2022 | Push to grasp | UR5 | Parallel jaw gripper | Self-supervised deep RL | A DRL approach for teaching robots how to manipulate objects in cluttered environments. | • Effective performance in both packed and pile object scenarios<br>• Outperforms the selected SOA in terms of task completion rate and grasp success in both scenarios. | • The limitation of the pushing strategy when dealing with objects that are hard to push due to friction.<br>• The possibility of grasp removing non-goal objects. | [80] |
| 2022 | Push to grasp | UR5 | Parallel jaw gripper | Truncated Quantile Critics (TQC) | Push objects to designated goal locations while avoiding collisions with other items in the workspace. | Outperforms an existing control-based method in terms of various metrics, including constant object contact and smooth trajectories while avoiding obstacles. | • Include testing it in real-world environments and addressing potential limitations, such as scalability to more complex scenarios or robustness to changes in object dynamics.<br>• Explore ways to improve the learning efficiency of the system, such as by incorporating human demonstrations or leveraging transfer learning. | [81] |
| 2023 | Pick and place with and without grip | UR5 | RG2 gripper | DQN | A framework for self-supervised and intelligent robotic pick-and-place operations in environments with clutter | • Achieve the optimal policy by going through a process of self-supervised trial and error.<br>• Promising results in comparison to different variants and baseline approaches for varying clutter densities and different test cases. | • The feature map concatenation factor of DenseNet-121 results in efficient management requirements.<br>• Incorporating these extensions may result in a system that is too large, leading to overestimation of future rewards and potential issues. To improve efficiency and throughput, it may be beneficial to explore Double Q-learning and Dueling Q-learning variants in the future. | [82] |

et al. introduced a loss function named RL-scene consistency loss is utilized to make sure that image translation is invariant with respect to the Q-values associated with it [91]. Ho et al. proposed RetinaGAN, a GAN-based approach to achieve consistency in object detection when adapting simulated images to realistic ones [92]. Ding et al. investigated a sim-to-real approach for incorporating tactile sensing into RL for tasks involving contact-rich interactions [93], while Lee et al. studyed the problem of robotic stacking with complex objects and propose a set of challenging objects intended to necessitate sophisticated techniques beyond basic pick-and-place methods [94]. Pedersen et al. proposed a method to transfer a grasping agent trained with DRL from a simulated environment to a physical robot [95]. They employed a reverse real-to-sim approach, utilizing a CycleGAN to bridge the reality gap between the simulated and real environments. These studies demonstrate the importance and effectiveness of simulation and transfer learning for robotic grasping in real-world applications. **Table 2** presents the latest research papers on grasping with a Sim-to-Real transfer, with the aim of offering a thorough summary of the current state-of-the-art achievements and future challenges in this field.

*3.1.3. Robots learning from demonstration* Learning from demonstration (LfD) is a significant concept in robotics, where a robot can acquire new skills by reproducing those of an expert. This model is highly significant in terms of developing robotics to realise complex tasks such as the grasp task. To this end, many studies and reviews were conducted, amongst them Zhu et al. [101] which reviewed Recent advancements and progress in the domain of LfD. In the other hand, Hussein et al. onducted a review of imitation learning methods and outlined various design options at different stages of the learning process [102]. Imitation learning approaches seek to replicate human behavior in a specific task, wherein an agent acquires the capability to carry out the task by mapping observations to actions based on demonstrations. Finn et al. investigated the potential use of inverse optimal

Table 2. Summarizing State-of-the-Art Achievements and Future Challenges in Grasping with Sim-to-Real Transfer

| Year | Application | Robot | Gripper | Learning algorithm | Highlight | Achievement | Future challenges | Reference |
|---|---|---|---|---|---|---|---|---|
| 2022 | Synergistic pushing and grasping | UR5 | RG2 gripper | Double DQN | A bifunctional push-grasping synergistic strategy is proposed for goal-agnostic and goal-oriented grasping tasks. | • Coordination of goal-agnostic and goal-oriented grasping tasks.<br>• System performance evaluated in simulation and real world.<br>• Synergy between pushing and grasping learned for accurate and efficient object pickup.<br>• Pre-trained model in simulation achieved high success rate in real world without fine-tuning, indicating practical feasibility. | N/A | [96] |
| 2022 | Non-prehensile push to grasp | Franka Emika Panda robotic arm | N/A | DVAE-SAC (Variational Autoencoder) | A sim-to-real technique for robotics applications that enables the transfer of a trained agent from simulation to reality without retraining or fine-tuning the control policy in the real domain | • Efficient visual manipulation learning.<br>• Effective sim-to-real transfer.<br>• Effective domain adaptation achieved. | • Performance gap finetuning potential.<br>• Handling complex real-world observations. | [97] |
| 2023 | Haptics-based object insertion | Franka Emika Panda robot | Soft Robotics Inc. mGrip gripper | SAC | Robot learning system trained in simulation for contact-rich object insertion with end-effector wrench and proprioception feedback, transferring directly to the real robot. | • Robotic learning for object insertion.<br>• Adaptability in object insertion.<br>• Ablation study and comparisons. | • Improve performance in specific scenarios.<br>• Improve inertial parameter identification. | [98] |
| 2023 | Food scooping with a spatula | Franka Panda arm | Parallel-jaw gripper | The NAF (Normalized Advantage Functions) | AdaptSim: a framework that focuses on maximizing task performance in target environments, rather than aligning simulation and reality dynamics. | • AdaptSim for sim-real adaptation.<br>• Parameter meta-learning for simulation adaptation.<br>• Improved real-world training efficiency. | • Design choices relaxation: practicality considerations.<br>• Adaptive updates: step size, covariance.<br>• Parameterization effect examination.<br>• Differentiable simulation for speedup. | [99] |
| 2023 | Robotic origami folding | Two robot manipulators: one for folding, the other for creasing | • An elongated gripper: end manipulation gripper.<br>• A roller: form the crease. | Path planning algorithm | Tackling a challenging robotic origami step: achieving a predetermined fold with one manipulator | • Robust paper folding strategy.<br>• Proving framework effectiveness experimentally.<br>• Real-time folding feedback algorithm.<br>• Accurate cardboard folding achieved. | • Paper crease asymmetry problem.<br>• Non-symmetric paper modeling.<br>• Robotic origami through pre-existing creases.<br>• Data-driven solutions with reinforcement learning. | [100] |

control (IOC) for learning behaviors from demonstrations, particularly for controlling high-dimensional robotic systems with torque [103]. Schoettler et al. examined challenging industrial insertion tasks that involve visual input and different types of natural rewards, including sparse rewards and goal images [104]. They demonstrated that combining reinforcement learning (RL) with prior knowledge, these tasks can be effectively solved with a moderate amount of interaction in the real world. Zhu et al. presented a model-free approach to DRL that utilizes a limited amount of demonstration data to support an RL agent. Their methodology was applied to robotic manipulation tasks and resulted in the training of policies that involve both visual perception and motor control, which utilize RGB camera inputs to determine joint velocities in an end-to-end manner [105]. Ragaglia et al. suggested a resolution to the Robot Learning from Demonstration (RLfD) challenge in dynamic environments. To demonstrate its efficiency, a set of pick-and-place experiments were performed using an ABB YuMi robot and the system's performance was evaluated accordingly. [106]. Recent studies have shown the possibility of training multi-task deep visuomotor policies for robotic manipulation through various forms of LfD and RL. The end-to-end LfD architectures' capabilities have been enhanced by Abolghasemi et al. to encompass object manipulation in environments with clutter [107]. A low-cost hardware interface has been proposed by Song et al. which can collect grasping demonstrations from individuals in diverse environments [108]. A dataset of human-robot demonstrations suitable for training robots for various tasks was presented by Sharma et al. [109]. Kim et al. provided an overview of robotic cleaning tasks utilizing different control methods [110], while Yang et al. suggested a DL model to learn robotic manipulation actions from videos of human demonstrations [111]. In contrast, Kilinc et al. suggested a RL based approach that does not rely on human demonstrations [112]. Smith et al. conducted research on how automated robotic learning frameworks can help overcome challenges related to defining and scaffolding

the learning process for multi-stage tasks [113]. Shahid et al. proposed a learning-based approach that utilizes simulation data to train robots for object manipulation tasks using RL [114]. Sena et al. presented a learning from demonstration model that takes into account the teacher's understanding of and influence on the learner [115]. An effective LfD policy for the secure grasping of compliant food objects by robots was proposed by Misimi et al. [116]. The approach used a blend of RGB-D images and tactile data to estimate the appropriate gripper pose, gripper finger configuration, and object forces. Ravichandar et al. provided a review of machine-learning methods used for robot learning from and imitation of a teacher, and discussed the mature and emerging application areas for LfD, highlighting the significant challenges that remain in both theory and practice [117]. Liang et al. investigated the feasibility of using LfD for teaching construction tasks to co-robots [118]. Solak et al. proposed n approach for acquiring in-hand robotic manipulation skills from human demonstrations using Dynamical Movement Primitives (DMPs). Subsequently, they replicated these tasks using a sturdy compliant controller based on the Virtual Springs Framework (VSF). The framework utilized real-time feedback from the contact forces recorded on the robot's fingertips [119]. Marzari et al. proposed a multi-subtask reinforcement learning (RL) methodology to overcome the limitations of learning from demonstration [120]. Meanwhile, James et al. discussed a voxel prediction approach for translation prediction in robotic manipulation and proposed a coarse-to-fine resolution increase [121]. Cai et al. presented a deep imitative reinforcement learning approach for agile autonomous racing using visual inputs, highlighting the potential of Learning from Demonstration (LfD) for enabling robots to perform complex tasks by imitating expert behavior [122]. **Table 3** here provides a comprehensive summary of the present-day accomplishments and future hurdles in the domain of robots learning from demonstrations to grasp, by showcasing the most recent research papers.

Table 3. Summarizing State-of-the-Art Achievements and Future Challenges in Grasping with Robots Learning from Demonstrations

| Year | Application | Robot | Gripper | Learning algorithm | Highlight | Achievement | Future challenges | Reference |
|---|---|---|---|---|---|---|---|---|
| 2022 | Grasp and release | Franka-Emika Panda 7-DOF robotic arm | Parallel jaw gripper | STL-based Bayesian optimization of LfD skills | A novel approach that takes into account precise task requirements in the context of Learning from Demonstration abilities. | • LfD algorithm development.<br>• STL specifies task constraints.<br>• Robot experiments successful. | • Curse of dimensionality.<br>• Complicated parameter space geometry.<br>• Nested STL exploration. | [123] |
| 2022 | Pick and place | WidowX 250 robot arm | Parallel jaw gripper | DeL-TaCo (Joint Demo-Language Task Conditioning) | A multi-task policy is trained on challenging robotic tasks using a combination of visual demonstration and language instruction through a method called DeL-TaCo (Joint Demo-Language Task Conditioning). | • DeL-TaCo framework developed.<br>• Generalization improvement.<br>• Human effort reduced. | • Interpretable modular encoders.<br>• Leveraging pretrained models. | [124] |
| 2023 | Close the box | Franka Emika Panda arm | Parallel jaw gripper | semi black box BO-$PI^2$ | Bayesian Optimized Policy Search methods and the Dynamic Bayesian Network (DBN) are utilized to improve the learned robotic abilities via demonstrations of keyframes. | • BO-PI2 development achievement.<br>• Dynamic Bayesian Network use.<br>• Improved reinforcement learning.<br>• Outperforms state-of-the-art. | • Branching policy exploration.<br>• Handling dynamic components.<br>• Extending to work with trajectories. | [125] |
| 2023 | Motion planning and grasp | Franka Emika Panda arm | Parallel jaw gripper | Learning human preferences from kinesthetic demonstration | Integrating human preferences into trajectory planning for robotic manipulators. | • Efficient planning demonstration.<br>• Low interaction effort requirement. | • Extend to shared object settings.<br>• Explore complex reward formulations. | [126] |

*3.1.4. Vision-based robotic grasp* Various methods have been explored in several studies aimed at developing vision-based robotic grasping techniques as a means of enabling intelligent robots to perceive and interact with their surroundings. For instance, Sehgal et al. proposed a Genetic Algorithm (GA) that accelerates the learning agent [127]. Similarly, Haarnoja et al. employed Soft Q-Learning (SQL), a maximum entropy reinforcement learning algorithm, to manipulate the robot's gripper and move it to a specific target position in Cartesian space [128].A scalable reinforcement learning approach for learning vision-based dynamic manipulation skills has been developed by Kalashnikov et al. [129]. In a separate study, Du et al. conducted a thorough investigation on the topic

of vision-guided robotic grasping [57]. Wu et al. proposed a method to mitigate the issue of poor performance in a stochastic environment by using an Actor-duelling-Critic (ADC) algorithm [130]. Lin et al. introduced a UAV vision-based aerial grasping system to capture target objects [131]. Nonetheless, these discrete settings have not yet been investigated in practical applications involving state-action spaces that are continuous and of high dimensions. To address this issue, Bodnar et al. developed Quantile QT-Opt (Q2-Opt), a distributional variant of distributed Q-learning algorithm, for continuous domains and evaluated its performance in both simulated and real vision-based robotic grasping tasks [132]. Additionally, Kobayashi et al. proposed a Reward-Punishment Actor-Critic (RP-AC) algorithm to optimize robot trajectory by acquiring suitable rewards [133], while Demura et al. used the You Only Look Once (YOLO) object detection approach to identify the optimal grasp point for stable manipulation in their Q-Learning grasping motion acquisition method [134]. This technique enabled the robot to pick up the uppermost folded towel from a stack and place it on a table. Kim et al. demonstrated that deep learning-based techniques with direct visual input can achieve state-of-the-art results for robotic grasping in a cluttered environment with diverse unseen target objects [73]. Julian et al. introduced a robot learning framework that allows for continuous adaptation [135], whine an approach that utilizes a reproducible sensor for precise and haptic grasping was proposed by Song et al. [136]. Muthusamy et al. proposed a novel dynamic finger system that utilizes vision to detect and suppress object slippage, and presented a baseline and feature-based method to detect slippage in the presence of illumination and vibration uncertainty [137]. Chen et al. suggested a framework for robotic visual grasping based on DRL, which has demonstrated effectiveness in learning complex control policies independently by training visual perception and control policy separately instead of end-to-end [138]. A simulated standard for evaluating robotic grasping that prioritizes off-policy learning and the aptitude for generalizing to unfamiliar objects was introduced by Quillen et al., highlighting the significance of diversity in facilitating the adaptation of the approach to novel objects that were not encountered in the training phase, as off-policy learning facilitates the usage of grasping data across a broad spectrum of objects [139]. Danielsen et al. explored diverse robotic manipulation and grasping techniques, and demonstrated through two PyBullet experiments the possibility of using DRL techniques to teach a robotic arm, which possesses seven degrees of freedom, how to grasp objects [140]. In another comprehensive survey, Kleeberger et al. presented a summary of ML techniques utilized for vision-based robotic manipulation and grasping [141]. Liu et al. noted that manipulators still face the challenge of only being able to grasp specific objects, unlike human beings that can use brain decision-making to pick up unfamiliar objects [142]. Reinforcement learning is often used in academia to train grasping algorithms, but it encounters issues such as insufficient algorithm stability, poor sample utilization, and limited exploration. To solve these problems, Liu et al. proposed using LfD, BC, and DDPG [142]. Grimm et al. presented a comprehensive system that encompasses stone segmentation, the creation of grasping hypotheses, and the implementation of pushing actions to achieve sturdy stone grasping [143]. Although reinforcement learning techniques have been effective, they are yet to achieve widespread success in various robotic manipulation tasks. To address this issue, James et al. presented an Attention-driven Robotic Manipulation (ARM) algorithm, which has the potential to tackle a variety of tasks with sparse rewards, requiring only a few demonstrations [144]. To overcome the challenge of actor-critic deep reinforcement learning methods struggling with the grasping of varied objects, especially in cases where learning is based on raw images and rewards that are sparse, Kim et al. utilized state representation learning (SRL) to capture crucial information for future use in RL [145]. In the field of robotic grasp, Cao et al. introduced a neuromorphic vision sensor named dynamic and active-pixel vision sensor (DAVIS) [146]. On the other hand, Wang et al. developed A learning system referred to as the Remote-Local Distributed (ReLoD)

system, which operates in real-time. It distributes calculations of two DRL algorithms between a local and a remote computer [147]. **Table 4** presents a comprehensive summary of the current achievements and future challenges in the field of vision-based robotic grasping. The table includes the most recent research papers.

Table 4. Summarizing State-of-the-Art Achievements and Future Challenges in Vision-based robotic grasping

| Year | Application | Robot | Gripper | Learning algorithm | Highlight | Achievement | Future challenges | Reference |
|---|---|---|---|---|---|---|---|---|
| 2021 | Grasp a whole set of objects | Franka Emika Panda arm | Two parallel fingers | GloCAL: Glocalized Curriculum-Aided Learning | GloCAL algorithm clusters discrete tasks based on their evaluation scores to generate a learning curriculum for agents. | • Automatic curriculum learning algorithm.<br>• Superiority of the algorithm. | • Parallel processing extension.<br>• Adaptive policy approach.<br>• Real-world efficiency testing. | [148] |
| 2022 | Turning on a light, pulling cloth from shelf, pulling a toy car, taking a lid off a saucepan, and folding a towel | Franka Emika Panda arm | Parallel jaw gripper | C2F-ARM (Coarse-to-Fine Attention-driven Robot Manipulation) | A coarse-to-fine discretization strategy, which replaces actor-critic methods that are prone to instability and require large amounts of data. | • C2F-ARM algorithm development.<br>• Sample-efficient learning algorithm.<br>• Simplification of ARM system.<br>• Support for multiple cameras.<br>• Investigating voxelization improvements. | • Voxel value expressiveness: Enhance small-scale detail.<br>• Dynamic path planning: Navigate complex environments.<br>• Continuous residual refinement: Improve output pose precision.<br>• Multitask/few-shot evaluation: Assess system versatility. | [121] |
| 2022 | Grasp on the Moon | A Robotnik Summit XL-GEN mobile manipulator that features a Kinova Gen2 robotic arm with 7DOF | Three-finger mechanical gripper | Truncated Quantile Critics (TQC) | One potential use of deep reinforcement learning is for visually-guided robotic grasping of objects situated on the Moon. | • Robotic grasping on Moon.<br>• 3D vs 2D observations.<br>• Domain randomization investigation.<br>• Sim-to-real transfer demonstration. | • Enhancing stability in diverse environments.<br>• Ensuring robustness for space robotics. | [149] |
| 2023 | • Press Button<br>• Pick Shed<br>• Open Drawer | WidowX 250 five-axes robot arm | Parallel jaw gripper | SAC-CQL and Synaptic Intelligence | Evaluate the efficacy of regularization-based techniques for offline RL of robotic manipulation tasks that rely on visual input performed in a sequential manner. | • Investigated catastrophic forgetting.<br>• Used SAC-CQL and SI.<br>• Tested different task scenarios.<br>• Task order affects learning.<br>• Prior knowledge importance. | • Improve knowledge transfer.<br>• Integrating prior knowledge.<br>• Experiment extension for patterns. | [150] |

### *3.2. DDPG, TD3, SAC*

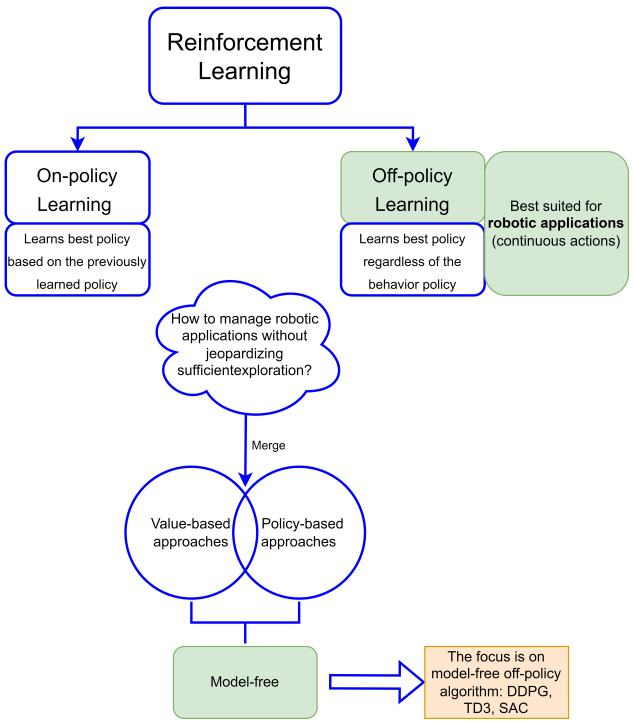


Figure 4. The reasoning behind model-free off-policy algorithms

The balance between exploring new options and exploiting existing knowledge is a well-known occurrence in RL. The agents must experiment with various choices in order to choose better options, but as they approach closer to the ideal course of action, they must make use of what they already know. Behavior guidelines are employed as the policy to interact with the environment and as a tool for exploration during training. On the other hand, target policy refers to the policy that the agent tries to learn. This reciprocity between behavior policy and target policy is conventional for on-policy and off-policy learning. While on-policy methods need the agent to act in accordance with the learned policy, off-policy methods can learn the best policy regardless of the behavior policy which is best suited to robotics applications [151], [139]. Furthermore, in the context of robotics, most actions, and state spaces are continuous. To handle continuous action spaces efficiently without losing adequate exploration, it's better to merge between value-based and policy-based approaches [152].Value-based and policy-based techniques, commonly known as model-free methods, do not utilize any environment model, thereby reducing their sample efficiency [152], [153]. **Figure 4** resumes the reasoning mentioned above. That's why in this section we'll do a thorough study of three of the main model-free off-policy DRL algorithms: *Deep Deterministic Policy Gradient (DDPG), Twin Delayed DDPG (TD3), Soft Actor-Critic (SAC).*

*3.2.1. Deep Deterministic Policy Gradient* As mentioned earlier, QL was a real breakthrough in RL, it is an off-policy TD algorithm that aims to learn the optimal action-value function $Q^(s, a)$. Once $Q^(s, a)$ has been learned, the policy can be derived from it. However, this algorithm is not suitable for large state-action spaces because there may be many unvisited regions, and it cannot generalize to state-action pairs that have not been visited. In other words, the algorithm's effectiveness is limited to small state-action spaces. The utilization of Deep Q-Learning is preferred due to its effectiveness when dealing with more complex state and action spaces. In such cases, Deep Learning is used as a function approximator to achieve optimal results. The process of function approximation involves creating an approximation of the Q-function based on examples of an agent's interactions with the environment. This technique enables the algorithm to generalize from states that have been visited by the agent to states that have not been visited, resulting in a substantial decrease in the quantity of states of states that need to be visited to reach an approximate solution. Besides being a DRL algorithm, Deep Q-Learning (DQL) is the act of combining Q-Learning with a deep neural network, and a deep neural network that approximates a Q function is called a deep Q-Network (DQN). It is important to note that in such a thriving field like AI, many terms are not fully established. For instance, DQL can also be referred to as DQN which can lead to confusion. Thus, before getting into the explanation of DQL, here is **Table 5** that attempts to enlighten the differences between Q-learning (QL), deep Q-learning (DQL), and deep Q-network (DQN) so that no skepticism occurs.

If a neural network is used, the Q-function is represented by a function that has parameters defined by the weights w. This means that, with each iteration, in lieu of modifying the Q values, the parameter vector w, which specifies the function, is updated instead:

$$W \leftarrow W + \alpha[r + \gamma max_{a_{t+1}} Q(s_{t+1}, a_{t+1}, w) - Q(s, a, w)] \bigtriangledown_w Q(s, a, w) \tag{1}$$

where $\bigtriangledown_w Q(s, a, w)$ is the gradient. To select optimal actions with DQN, The Neural Network (NN) takes an input state s and its outputs are going to be the q values corresponding to different actions within the action space if the actions are discrete. If they're not, then it couldn't enumerate all the actions in this manner. In the discrete set, to use the DQN values to select actions to convert it into a policy and select the optimal actions in the environment, all

Table 5. Dissimilarity between QL, DQL and DQN

| **Questions** | **QL** | **DQL** | **DQN** |
|---|---|---|---|
| Is it an RL algorithm? | Yes | Yes | No |
| Does it use neural networks? | No | Yes | No |
| Is it a model? | No | No | Yes |
| Can it deal with continuous state spaces? | No | Yes | Yes |
| Can it deal with continuous action spaces? | Yes | Yes | Yes |
| Does it converge? | Yes | Maybe | Maybe |
| Is it an online learning algorithm? | Yes | No | No |

that should be done is take the argmax over a of Q(s,a) by taking the maximum of all of those discrete set of values and get a* and that's also our policy $\pi(s)$. The described policy is employed to choose actions once the Q-Network has been trained. However, it is worth noting that a similar process occurs even during the training phase. During training, the Bellman targets are set as the Q targets.

$$y = R_{t+1} + \gamma max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) \tag{2}$$

So overall, all the agent has to do is take the maximum of a discrete set of values. On the other hand, in the continuous set there is no meaningful way to enumerate the actions, so some modifications should occur on the Q-Network. The NN can take the state s and action a as input and output Q(s,a) but there is still going to be a problem here which is that the policy can't be simply set based on the argmax over a. So this looks a bit like an optimization problem where for each state the agent has to determine the best action given the action input and this will be too expensive. There is one potential solution to overcome this problem. Let's train a NN to produce the outputs of this optimization problem by mapping the input state to this output action a* which is the solution of this optimization problem. So the network takes in s as input and produces the best action a* as output which remembers us of what the optimal policy should be doing, so let's call this network the policy network. To train this to maximize the q function, the Q-function is parameterized in a Q-Network and the training will occur by using the standard squared bellman error loss $L = \sum(Q_{target} - Q)^2$ and it's very common to call this kind of setup an actor critic setup where the policy is called the actor obviously because it produces the actions and the Q-Network is called the critic because you can think of it as evaluating a state action tuple and saying how good it is which is exactly what q is. So this is the actor-critic algorithm and this algorithm where you can take DQN and modify it in this way to work well with continuous actions is called Deep Deterministic Policy Gradient (DDPG) and this method is quite often used in robotics.

- Some related work: Kerzel et al. put forward a novel method to tackle the challenge of collecting a vast number of training samples within a reasonable time frame, and demonstrated their method on a reach-for-grasp task that employs the Deep Deterministic Policy Gradients (DDPG) algorithm [154]. The goal-auxiliary DDPG algorithm, introduced by Wang et al., facilitates the effective acquisition of policies

for controlling grasping in 6 dimensions (6D) from point cloud data. This approach entails utilizing demonstrations from a specialist grasp planner and motion, moreover, it incorporates anticipation of grasping objectives as an additional task to enhance the performance of both the critic and the actor. [155]. Wang et al. also proposed the experience-based policy gradient method (EBDDPG), which promotes smooth robot movements. Results demonstrated that this method improves the success rate of grasping tasks and encourages smoother manipulation [156]. Controlling the gripping of a robot arm can be improved by using the enhanced DDPG reinforcement learning algorithm introduced by Qi and Li [157]. In addition, Beik Mohammadi et al. presented an online continuous deep reinforcement learning approach for a reach-to-grasp task in a mixed-reality environment [158].

- Open problems: When using reinforcement learning with discrete action spaces, sub-optimal policies can arise due to a problem called overestimation bias. In continuous control settings, deterministic policy gradients can also suffer from overestimation bias [159]. Overestimation bias is a familiar issue within algorithms for reinforcement learning that are based on value estimation, such as DDPG and deep Q-networks, that arise from function approximation and can lead to sub-optimal policies [160]. To overcome this issue, a modified version of the DDPG algorithm, called Twin Delayed Deep Deterministic Policy Gradient (TD3), has been proposed.

*3.2.2. Twin Delayed Deep Deterministic Policy Gradient* Twin-Delayed DDPG (TD3) is a highly intelligent deep reinforcement learning model that combines the latest methods in AI. These include continuous Double Deep Q-Learning, actor-critics, and policy gradient [161]. As outlined in the previous section, TD3 comes in to improve the approximation error [162] [161] [163] [163]. TD3 is a modified version of DDPG that incorporates several techniques to address the overestimation of the value function. These techniques include Target Policy Smoothing, Delayed update of Target and Policy Networks, and Clipped Double Q-learning. Let's go into details, TD3 uses 2 critics, from which the word twin comes so each critic has different values of the Q-value. The TD3 algorithm can be seen as two parts: the QL part of the training process and the policy learning part. In the part QL, first, the replay memory is initialized, then for the actors, two NN are built, one NN for the actor model and one NN of the actor target. For the critics, two NN are built for for the critic model and two NN for the critic targets. So in total, there are 2 actor NN and 4 critic NN. Here's an overview of the training process of these neural networks:
$Actortarget \rightarrow Critictarget \rightarrow Critictarget \rightarrow Criticmodel \rightarrow Criticmodel \rightarrow Actormodel$
After building these NN, a batch of transitions (s, $s_{t+1}$, a, r) is sampled from the memory. Then for each element of the batch, The actor target plays the next action $a_{t+1}$ form the next state $s_{t+1}$ then a Gaussian noise is added to this next action $a_{t+1}$ and clamped within the scope of values that the environment accommodates. Afterwards, the two critic targets take $(s_{t+1}, a_{t+1})$ as input and output two Q-values $Q_1(s_{t+1}, at+1)$ and $Q_2(s_{t+1}, at+1)$. Only the smaller of the two Q-values is kept, representing the estimated value of the following state. This minimum allow us to get the final target which is:

$$Q_T = r + \gamma min(Q_1, Q_2) \tag{3}$$

Each couple (s,a) is inputted into both critic models and they output two Q-Values $Q_1(s,a)$ and $Q_2(s,a)$ which are compared to the minimum critic target. Then the loss between the two critic models is computed through:

$$CriticLoss = MSELoss(Q_1(s,a), Q_T) + MSELoss(Q_2(s,a), Q_T) \tag{4}$$

In order to reduce the critic loss, the parameters of the two Critic models over the iterations are updated with back propagation and the weights are updated through stochastic gradient descent.

Moving to the policy learning part, the Q-values of the critic models are used to perform gradient ascent to maximize the returns. Once the actor model is updated, the agent returns better actions which maximizes the Q-values and the agent moves nearer to the optimal return. In other words, every d iterations, the actor model is updated through gradient ascent on the output of the first critic model. Then every d iterations, the actor target's weights is updated through Polyak averaging:

$$\theta_i' \leftarrow \tau\theta_i + (1-\tau)\theta_i' \tag{5}$$

The equation mentioned consists of four components, the first component $\theta_i'$ represents the actor target parameters, the second component $\tau$ denotes a small number, the third component $\theta_i$ represents the actor model parameters, and the last component $\theta_i'$ represents the actor target parameters before updating. This equation can be interpreted as a gradual transfer of weights from the actor model to the actor target, which results in bringing the actor target closer to the actor model with each iteration. Thus, the actor model learns from the actor target, which stabilizes the learning process. Similarly, after every d iterations, the weights of the critic targets are updated in a similar manner through polyak averaging.

$$\phi' \leftarrow \tau\phi + (1-\tau)\phi' \tag{6}$$

In this TD3 algorithm, $\phi$ denotes the parameters of the critic target. The delayed aspect of this approach is due to the fact that the actor and critic are updated only every d iterations, which is intended to enhance performance compared to the standard DDPG technique.

- Related work: Hou et al. introduced RTD3, a modified version of the TD3 algorithm, to tackle the problem of overestimation bias in multi-degree of freedom manipulator learning through deep reinforcement learning [164]. Overestimation of Q-values by the learned Q-function is a common problem with DDPG, which may cause the policy to break, as it exploits the Q-function's errors. To address this issue, M et al. combined TD3 with Hindsight Experience Replay (HER) [165]. Khoi et al. utilized the TD3 algorithm along with a novel reward model to simulate the gait of a 6-DOF biped robot in a Gazebo/ROS environment [166]. Yang and Xu aimed to design a robot that can aid in warehouse object grasping using various DRL algorithms, including TD3 [167].
- Open problems: According to Nguyen and La [33] as well as Nian et al. [168], some of the recent successful RL algorithms, including Trust Region Policy Optimization (TRPO), Asynchronous Actor-Critic Agents (A3C), and Proximal Policy Optimization (PPO) are prone to sample inefficiency. In contrast, off-policy methods based on Q-learning, like the Deep Deterministic Policy Gradient (DDPG), and Twin Delayed DDPG (TD3) are less susceptible to this issue. They utilize replay buffers to efficiently learn from past samples. However, these off-policy methods based on Q-learning are highly sensitive to hyper-parameters and need a significant amount of tuning to achieve convergence. To address the issue of convergence fragility, Soft Actor-Critic (SAC) adopts a similar approach as the aforementioned methods and integrates techniques to combat this challenge.

*3.2.3. Soft Actor-Critic* Soft Actor-Critic (SAC) is also a DRL algorithm defined for continuous actions. The three main components of SAC are: an actor-critical architecture with distinct networks of policies and value functions,

a formulation that is not limited by the policy used to collect the data and enables the utilization of previously gathered data to enhance effectiveness. Additionally, it includes the maximization of entropy to guarantee stability and promote the exploration of alternative options.

SAC uses a modified RL objective function and its main goals are to optimize both the policy's rewards and entropy. The concept of entropy refers to the level of unpredictability associated with a random variable. The reasons behind wanting the policy to have high entropy are: to encourage exploration, to induce equal probabilities for actions that have either equal or almost identical Q values, to make sure that the policy does not break down by repeatedly selecting a specific action that could potentially take advantage of any inconsistencies within the estimated Q function. With all the mentioned above, SAC algorithms can overcome the brittleness problem. And its objective function to maximize the expected return and the entropy at the same time is:

$$J(\pi) = \sum_{t=0}^{T} \mathbb{E}_{(s_t,a_t)\sim\rho_\pi}[r(s_t,a_t) + \alpha H(\pi(. \mid s_t))] \tag{7}$$

In order to achieve this optimization, SAC uses 3 networks: a state-value function V parametrized by $\psi$, a soft Q-function Q parametrized by $\theta$ and a policy function $\pi$ parametrized by $\phi$.

The Value network can be trained by minimizing:

$$J_V(\psi) = \mathbb{E}_{s_t\sim D}\left[\frac{1}{2}\left(V_\psi(s_t) - \mathbb{E}_{a_t\sim\pi_\phi}\left[Q_\theta(s_t,a_t) - \log\pi_\phi(a_t \mid s_t)\right]\right)^2\right] \tag{8}$$

This equation implies that across all states sampled from the replay buffer of the experiment D, it is necessary to reduce the squared difference between the value network prediction and the anticipated prediction of the function Q added to the political function $\pi$ entropy (here, the negative log of the policy function measures it).

To train the Q-Network, the following error should be minimized:

$$\hat{Q}(s_t,a_t) = r(s_t,a_t) + \gamma\mathbb{E}_{s_{t+1}\sim p}[V_\psi(s_{t+1})] \tag{9}$$

This means that for all (s,a) pairs within the experiment's replay buffer, one aims to reduce the squared difference between the Q-Function's prediction and the immediate reward plus the updated awaited value of the following state. V is the target value function here. And to train the policy network, the following error should be minimized:

$$J_\pi(\phi) = \mathbb{E}_{s_t\sim D}[D_{KL}(\pi_\phi(. \mid s_t) \parallel \frac{\exp(Q_\theta(s_t,.))}{Z_\theta(s_t)})] \tag{10}$$

Essentially, this objective function is intended to cause the policy function distribution to more closely resemble the distribution of the exponentiation of the Q-function standardized by a different Z-function.

- Some related work: Chen and Lu proposed a system for object grasping that combines object detection techniques and the Soft-Actor-Critic (SAC) algorithm, using an approaching-tracking-grasping scheme [169]. Feldman et al. introduced an approach to self-supervised reinforcement learning using a hybrid discrete-continuous adaptation of SAC [170]. Shahid et al. suggested a learning-based approach for object

manipulation using simulated data and two model-free reinforcement learning algorithms: SAC and Proximal Policy Optimization (PPO) [171].

- Open problems : Haarnoja et al. proposed the SAC algorithm to combat convergence brittleness observed in the other off-policy model free DRL algorithms (DDPG, TD3) [172]. But it turned out that the SAC algorithm also suffers from brittleness due, this time, to the alpha temperature that regulates exploration. To overcome this problem, the authors suggested automatic temperature tuning. Haarnoja et al. adapted this solution, however, another situation occurred which is the high variance problem [173]. These limitations are still configuring as open issues to this day.

## 4. Discussion and quantitative analysis

Determining the best algorithm to realize the grasping task is still debated by many researchers. On-policy or off-policy? Policy-based or value-based? Model-based or Model-free? The review determined, based on the current state-of-the-art, which algorithm is the best fit for continuous control applications such as the grasping task. The paper stated that all-in-one off-policy, model-free algorithms, including DDPG, TD3, and SAC, have been the most effective ones thus far. To support this statement, **Table 6** was synthesized in the third section that included most of the algorithms used for learning the grasp task. All the papers reviewed in the study showed that the SAC algorithm outperforms the other off-policy algorithms. It is worth noting, however, that the papers also evaluated the performance of both off-policy algorithms and on-policy algorithms. Haarnoja et al. proposed the soft actor-critic, an off-policy actor-critic DRL method built on the framework of maximum entropy [172]. In this approach, the actor strived to maximize both entropy and expected reward. Their strategy delivered state-of-the-art performance on a variety of continuous control benchmark problems by combining off-policy updates with a stable stochastic actor-critic formulation. SAC, as an algorithm based on the maximum entropy principle, showed superior performance compared to baseline methods in terms of both learning time and final performance, particularly in challenging tasks. It also demonstrated better sample efficiency and ultimate performance than state-of-the-art techniques in previous studies. In contrast to PPO, which struggled with complicated and high-dimensional tasks, SAC was able to learn quickly due to its ability to handle large batch sizes. These results suggest that maximum entropy principle-based algorithms may be more effective in challenging tasks. The researchers developed a soft actor-critic algorithm based on these findings, which was shown to outperform state-of-the-art model-free deep reinforcement learning techniques such as DDPG and PPO. As a following work, Haarnoja et al. described SAC and thoroughly assessed SAC on a number of benchmark tasks as well as difficult real-world tasks including quadrupedal robot mobility and manipulating robots with a dexterous hand [173]. By making these adjustments, SAC surpassed the performance of earlier on-policy and off-policy approaches in terms of sample efficiency and asymptotic performance, achieving state-of-the-art performance. Additionally, they showed that, in contrast to other algorithms that are off-policy, their method exhibits considerable stability and achieves similar performance across different arbitrary seeds.. These findings imply that SAC is a strong contender for learning in practical robotics challenges. Their empirical research demonstrated that SAC, which can be used to train deep neural network policies and does not require any environment-specific hyperparameter tuning, can perform on par with or better than state-of-the-art model-free deep RL methods like the off-policy TD3 algorithm and the on-policy PPO algorithm. Chen and Lu demonstrated that their developed system, which separates object detection

from DRL control, enables autonomous grasping of a moving object with varying trajectories [169]. Even though gripping a moving object in an unstructured environment is a challenging task, the actual experiment showed that the recommended intelligent system can produce encouraging outcomes with the SAC algorithm. Better outcomes than with DDPG or TD3 algorithms. Ünal [174] examined the controller strategies for a pick-and-place operation using a bi-rotor aerial manipulator. In addition, they studied how the change in the goal location of the object that the aerial manipulator must transport affects the training of the learning approaches and looked at the implications of manipulator degrees of freedom for DRL approaches. In their experiments, they analyzed the on-policy algorithms first. No matter how little their final mean reward differences were, TRPO outperformed PPO in terms of overall performance. PPO learned more quickly than the TRPO algorithm. Their results indicated that all approaches, with TRPO being the most stable, had similar mean episode lengths at the conclusion of training. PPO obtained high success rates more quickly than any other algorithm. Afterward, They analyzed the off-policy algorithms. When compared to the others, DDPG converged to a somewhat worse mean reward. They all arrived at a similar mean episode duration throughout training, with the SAC algorithm being the finest and the DDPG algorithm being the least efficient as expected, given SAC and TD3 build upon DDPG and attempt to increase its convergence and stability. Additionally, they displayed the mean success rates of the three off-policy algorithms during training, and once more, the results are the same: SAC and TD3 achieve very similar success rates, while DDPG achieves the worst. All of them achieved a respectable success rate almost simultaneously, with DDPG being a little bit slower. Subsequently, they compared the best on-policy and off-policy algorithms (SAC and TRPO). Compared to the SAC algorithm, TRPO was superior. This may be the case since the SAC algorithm's hyperparameters were not specifically tuned for the task at hand given the small difference between them. TRPO was superior in terms of time duration, but in terms of the number of time steps the results show that this is not the case. So the overall result stated that off-policy algorithms are demonstrably considerably more sample-efficient. Here, [171], PPO and SAC were studied, the fine-tuning approach, which displayed the continual adaptation of on-policy RL to changing contexts and enabled the acquired policy to adjust and execut the revised task, was offered to quicken the learning process. It was shown that the learned control strategy may be applied to a variety of object geometries and initial robot/part configurations. In fact, SAC should have acquired the task at a faster rate in terms of the number of episodes required because this is an off-policy algorithm that utilizes previously recorded transition data stored in a replay buffer. The training performance of the SAC algorithm and the PPO algorithm for the considered gripping task were compared in order to validate this notion. During the initial 2 million time steps of both algorithms, the mean reward and the number of successful episode steps were plotted, and a comparison was provided. The SAC algorithm learned to amass substantially greater average rewards than PPO and to complete tasks in just 2M time steps, confirming the premise, while PPO failed to complete tasks for the same amount of episodes. The results for both methods, however, were considerably different when the average reward and the count of successful episode steps were studied against the wall time. As seen in the results, with the SAC's off-policy updates, the rewards obtained during each update iteration are lower compared to PPO's on-policy updates. When obtaining new experience incurs significant costs and computational resources are not a concern, off-policy approaches such as SAC may be more favorable.

Table 6. Summary of the key findings of the SOA robotic grasping that considered closely related works

| Year | Application | Robot | Gripper and features | Simulation env | Learning algorithm | Study methodology | Key findings | Reference |
|---|---|---|---|---|---|---|---|---|
| 2018 | Evaluation of SAC against both prior off-policy and on-policy DRL algorithms across a variety of continuous control tasks, including the grasp task. | N/A | N/A | • OpenAI gym<br>• rllab | • SAC<br>• DDPG<br>• TD3<br>• SQL<br>• PPO | SAC's performance is compared to that of earlier methods on a variety of difficult continuous control tasks from the OpenAI gym benchmark suite as well as on the Humanoid task's rllab implementation | • In terms of learning time and final performance, SAC surpasses baseline approaches by a significant margin on the harder tasks while performing comparable to them on the easier tasks<br>• Although SQL is likewise capable of learning all tasks, its asymptotic performance is inferior and it is slower than SAC | [172] |
| 2018 | • Mobility<br>• Robotic manipulation | • Minitaur robot<br>• Dynamixel Claw | • Quadruped with eight direct-drive actuators<br>• Dexterous hand with 3 fingers | • OpenAI gym<br>• rllab | • SAC (learned temperature)<br>• SAC (fixed temperature)<br>• DDPG<br>• TD3<br>• PPO | Five separate iterations of each algorithm were trained using various random seeds, and each carried out an evaluation rollout every 1000 environmental steps. For SAC, there are two versions included: one where the temperature parameter is fixed, treated as a hyperparameter, and tuned for each environment separately. The other version uses an automatic temperature adjustment | In The most difficult tasks, the soft actor-critic method consistently outperforms both on-policy and off-policy techniques | [173] |
| 2018 | Grasp and lift a set of simple objects | ABB Yumi | • Parallel-jaw gripper<br>• Depth camera | • Pybullet<br>• 3DNet | TRPO | • Training the model in simulation<br>• Evaluating the model in both virtual and real-world experiments | • Training on large workspace: success rate = 0 even after 200 policy iterations due to failure of exploration<br>• Training on a small workspace achieved a perfect score on the same task | [175] |
| 2020 | Several robotic arms working together to learn a common strategy to reach an object | Kuka arm | N/A | Bullet physics engine | PPO | Variability in the agents' capacity to see and act with precision in depend on the environment's perturbance | Variable performance has the biggest impact on the network's ability to converge, even while interruptions in robots' ability to correctly actuate have had a much lesser impact than those in their ability to consistently sense the position of the item | [176] |
| 2020 | Push and grasp to manipulate objects in clutter | Baxter | Parallel grippers | MuJoCo | TD3 | • Policy: decides where to start pushing and pushing direction based on the current image<br>• Policy training: TD3<br>• Grip detection: rule-based method | The algorithm is capable of removing many objects with high effectiveness and success rates | [177] |
| 2020 | Reach to grasp in cluttered environment | Pepper | Realsense D435 camera | Qibullet | PPO | Two robots are employed in the simulated environment in this study; one is operated by the PPO algorithm while the other is placed in random locations | The PPO method assisted in applying a simulation-trained action module policy. When tested on an actual robot and in simulation, the developed strategy was able to offer motion gestures that successfully contacted a moving object | [178] |
| 2021 | Reach to grasp | Dual arm robot | 7-DOF | Gazebo | SAC | • SAC-based motion planning is used to demonstrate how to dynamically prevent self-collision, joint limitations, and singularities as well as how to direct the arm to the desired position<br>• Testing the model in simulation<br>• Testing the model in the real world | • The correct choice of network inputs and reward functions has a significant impact on the outcomes of network training<br>• The suggested approach does an excellent job of preventing the issues of self-collision, joint limit, and singularity | [179] |
| 2021 | Sim-to-real approaching-tracking-grasping moving objects | Baxter | parallel grippers | CoppeliaSim | • SAC<br>• TD3<br>• DDPG | • Training process with SAC in simulation<br>• Comparison between SAC, TD3 and DDPG<br>• Testing of the trained model and policy on a real robot to evaluate the grasping system proposed | • With the SAC algorithm, the training converges with a fair reward and a success rate after 100,000 episodes<br>• After the same number of episodes with DDPG and TD3 ,training converges with a less just reward and success rate<br>• A simulation-trained SAC-based policy can be successfully used in reality. This is because the background environment, robot dynamics, and object recognition in the simulation and the real world are different. | [169] |
| 2021 | Aerial manipulation pick and place tasks | UAV with a robot arm | N/A | • Box2D physics engine<br>• OpenAI's gym | • A2C<br>• TRPO<br>• PPO<br>• DDPG<br>• TD3<br>• SAC | • On-policy Algorithm Analysis<br>• Off-policy Algorithm Analysis<br>• On-policy vs. Off-policy Algorithms analysis | • No matter how little their final mean reward differences were, TRPO outperformed PPO in terms of overall performance. PPO learnt more quickly than TRPO.<br>• SAC and TD3 get relatively similar success rates to those of the three off-policy algorithms during training, but DDPG earns the worst outcomes.<br>• TRPO is superior than the SAC algorithm in terms of time duration, however when the graphs are displayed in terms of the number of time steps, TRPO is not superior.<br>• Off-policy algorithms are demonstrably considerably more sample-efficient. | [174] |
| 2022 | Grasping in cluttered environment (bin-picking) | Franka Emika Panda arm | • Realsense D415 camera on the robot gripper<br>• Parallel gripper | Pybullet | DDPG | This approach improves the performance of both the actor and the critic through the utilization of demonstrations from an expert motion and grasp planner, as well as employing grasping goal prediction as an auxiliary task | Optimal performance is achieved when both the performer and the critic are assigned the goal-auxiliary task | [155] |

## 5. Conclusion

The research paper presents an extensive examination of different Reinforcement Learning (RL) algorithms intended for robotic grasping tasks, concentrating particularly on Deep Reinforcement Learning (DRL) algorithms. The study highlights the most effective DRL algorithms for handling complex and challenging tasks, such as grasping. Additionally, to simplify the research process for others, the paper provides a collection of different forms of DRL grasping tasks. The analysis indicates that model-free off-policy approaches, such as DDPG, TD3, and SAC, are more suitable for robotic applications, especially for continuous actions. The study concludes with a summary of the benefits and drawbacks of RL and a deep analysis of the most prominent algorithms in robotics grasping, along with open problems for further research. The insights presented in this paper emphasize the importance of continued research and development of DRL algorithms to enhance the capabilities of robots in handling complex and challenging tasks. Overall, this study contributes to advancing the field of robotic manipulation and provides a useful resource for researchers seeking to explore the potential of RL in robotics grasping.

## A. Nomenclature

- I2A : Imagination-Augmented Agents
- MBMF : Model-Based Priors for Model-Free Reinforcement Learning
- MBVE : Model-Based Value Expansion
- A2C : Advantage Actor Critic
- A3C : Asynchronous Advantage Actor Critic
- PPO : Proximal Policy Optimization
- TRPO : Trust Region Policy Optimization
- DQN : Deep Q-Learning
- C51 : Categorical DQN
- QR-DQN : Quantile Regression DQN
- HER : Hindsight Experience Replay

## Acknowledgement

This work was supported by Euro-Mediteranean University of Fez.

## REFERENCES

1. Jianxing He, Sally L Baxter, Jie Xu, Jiming Xu, Xingtao Zhou, and Kang Zhang. The practical implementation of artificial intelligence technologies in medicine. *Nature medicine*, 25(1):30–36, 2019.
2. Rosemary Luckin and Mutlu Cukurova. Designing educational technologies in the age of ai: A learning sciences-driven approach. *British Journal of Educational Technology*, 50(6):2824–2838, 2019.
3. Kwonsang Sohn and Ohbyung Kwon. Technology acceptance theories and factors influencing artificial intelligence-based intelligent products. *Telematics and Informatics*, 47:101324, 2020.
4. Vijay Kakani, Van Huan Nguyen, Basivi Praveen Kumar, Hakil Kim, and Visweswara Rao Pasupuleti. A critical review on computer vision and artificial intelligence in food industry. *Journal of Agriculture and Food Research*, 2: 100033, 2020.

5. Andre Esteva, Katherine Chou, Serena Yeung, Nikhil Naik, Ali Madani, Ali Mottaghi, Yun Liu, Eric Topol, Jeff Dean, and Richard Socher. Deep learning-enabled medical computer vision. *NPJ digital medicine*, 4(1):1–9, 2021.
6. Mohamed R Ibrahim, James Haworth, and Tao Cheng. Understanding cities with machine eyes: A review of deep computer vision in urban analytics. *Cities*, 96:102481, 2020.
7. Andrew Wen, Sunyang Fu, Sungrim Moon, Mohamed El Wazir, Andrew Rosenbaum, Vinod C Kaggal, Sijia Liu, Sunghwan Sohn, Hongfang Liu, and Jungwei Fan. Desiderata for delivering nlp to accelerate healthcare ai advancement and a mayo clinic nlp-as-a-service implementation. *NPJ digital medicine*, 2(1):1–7, 2019.
8. Sherin Mary Mathews. Explainable artificial intelligence applications in nlp, biomedical, and malware classification: a literature review. In *Intelligent computing-proceedings of the computing conference*, pages 1269–1292. Springer, 2019.
9. Mark Anthony Camilleri and Ciro Troise. Live support by chatbots with artificial intelligence: A future research agenda. *Service Business*, pages 1–20, 2022.
10. Bhabendu Kumar Mohanta, Debasish Jena, Utkalika Satapathy, and Srikanta Patnaik. Survey on iot security: Challenges and solution using machine learning, artificial intelligence and blockchain technology. *Internet of Things*, 11:100227, 2020.
11. Isabella Castiglioni, Leonardo Rundo, Marina Codari, Giovanni Di Leo, Christian Salvatore, Matteo Interlenghi, Francesca Gallivanone, Andrea Cozzi, Natascha Claudia D'Amico, and Francesco Sardanelli. Ai applications to medical images: From machine learning to deep learning. *Physica Medica*, 83:9–24, 2021.
12. Batta Mahesh. Machine learning algorithms-a review. *International Journal of Science and Research (IJSR).[Internet]*, 9:381–386, 2020.
13. Xian-Da Zhang. A matrix algebra approach to artificial intelligence. 2020.
14. Maria Kyrarini, Quan Zheng, Muhammad Abdul Haseeb, and Axel Gräser. Robot learning of assistive manipulation tasks by demonstration via head gesture-based interface. In *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, pages 1139–1146. IEEE, 2019.
15. Sahar Bahrami, Jérémy Moriot, Patrice Masson, and François Grondin. Machine learning for touch localization on ultrasonic wave touchscreen. *arXiv preprint arXiv:2202.08947*, 2022.
16. Somayeh B Shafiei, Saeed Shadpour, James L Mohler, Farzan Sasangohar, Camille Gutierrez, Mehdi Seilanian Toussi, and Ambreen Shafqat. Surgical skill level classification model development using eeg and eye-gaze data and machine learning algorithms. *Journal of Robotic Surgery*, pages 1–9, 2023.
17. Rania Kolaghassi, Mohamad Kenan Al-Hares, and Konstantinos Sirlantzis. Systematic review of intelligent algorithms in gait analysis and prediction for lower limb robotic systems. *IEEE Access*, 9:113788–113812, 2021.
18. Amir Masoud Rahmani, Efat Yousefpoor, Mohammad Sadegh Yousefpoor, Zahid Mehmood, Amir Haider, Mehdi Hosseinzadeh, and Rizwan Ali Naqvi. Machine learning (ml) in medicine: Review, applications, and challenges. *Mathematics*, 9(22):2970, 2021.
19. Ramchandra Rimal. Identifying the neurocognitive difference between two groups using supervised learning. *Statistics, Optimization & Information Computing*, 12(1):15–33, 2024.
20. Kushal Rashmikant Dalal. Analysing the role of supervised and unsupervised machine learning in iot. In *2020 international conference on electronics and sustainable communication systems (ICESC)*, pages 75–79. IEEE, 2020.
21. Deepti Lamba, William H Hsu, and Majed Alsadhan. Predictive analytics and machine learning for medical informatics: A survey of tasks and techniques. In *Machine Learning, Big Data, and IoT for Medical Informatics*, pages 1–35. Elsevier, 2021.
22. Miguel Vasco. Multimodal representation learning for robotic cross-modality policy transfer. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2225–2227, 2020.
23. Khushi Kumari Jha, Roshan Jha, Ankita Kumari Jha, Md Al Mahedi Hassan, Saurav Kumar Yadav, and Tr Mahesh. A brief comparison on machine learning algorithms based on various applications: a comprehensive survey. In *2021 IEEE International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)*, pages 1–5. IEEE, 2021.
24. Jinqiang Bai, Shiguo Lian, Zhaoxiang Liu, Kai Wang, and Dijun Liu. Deep learning based robot for automatically picking up garbage on the grass. *IEEE Transactions on Consumer Electronics*, 64(3):382–389, 2018.

25. Kei Kase, Kanata Suzuki, Pin-Chu Yang, Hiroki Mori, and Tetsuya Ogata. Put-in-box task generated from multiple discrete tasks by ahumanoid robot using deep learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6447–6452. IEEE, 2018.
26. Shenshen Gu, Xinyi Chen, Wei Zeng, and Xin Wang. A deep learning tennis ball collection robot and the implementation on nvidia jetson tx1 board. In *2018 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 170–175. IEEE, 2018.
27. Shehan Caldera, Alexander Rassau, and Douglas Chai. Robotic grasp pose detection using deep learning. In *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1966–1972. IEEE, 2018.
28. Yuki Onishi, Takeshi Yoshida, Hiroki Kurita, Takanori Fukao, Hiromu Arihara, and Ayako Iwai. An automated fruit harvesting robot by using deep learning. *Robomech Journal*, 6(1):1–8, 2019.
29. Jaeseok Kim, Nino Cauli, Pedro Vicente, Bruno Damas, Alexandre Bernardino, José Santos-Victor, and Filippo Cavallo. Cleaning tasks knowledge transfer between heterogeneous robots: a deep learning approach. *Journal of Intelligent & Robotic Systems*, 98(1):191–205, 2020.
30. Yang Yang, Hengyue Liang, and Changhyun Choi. A deep learning approach to grasping the invisible. *IEEE Robotics and Automation Letters*, 5(2):2232–2239, 2020.
31. Weiwei Shang, Fangjing Song, Zengzhi Zhao, Hongbo Gao, Shuang Cong, and Zhijun Li. Deep learning method for grasping novel objects using dexterous hands. *IEEE Transactions on Cybernetics*, 2020.
32. Marwan Qaid Mohammed, Kwek Lee Chung, and Chua Shing Chyi. Review of deep reinforcement learning-based object grasping: Techniques, open challenges, and recommendations. *IEEE Access*, 8:178450–178481, 2020.
33. Hai Nguyen and Hung La. Review of deep reinforcement learning for robot manipulation. In *2019 Third IEEE International Conference on Robotic Computing (IRC)*, pages 590–595. IEEE, 2019.
34. Thos Beacall. A tooth of hybodus grossicornis from the inferior oolite. *Nature*, 58(1504):390–390, 1898.
35. James B Rawlings and Michael J Risbeck. Model predictive control with discrete actuators: Theory and application. *Automatica*, 78:258–265, 2017.
36. Arthur E Bryson. Optimal control-1950 to 1985. *IEEE Control Systems Magazine*, 16(3):26–33, 1996.
37. Alex M Andrew. Reinforcement learning: An introduction by richard s. sutton and andrew g. barto, adaptive computation and machine learning series, mit press (bradford book), cambridge, mass., 1998, xviii+ 322 pp, isbn 0-262-19398-1,(hardback,£ 31.95). *Robotica*, 17(2):229–235, 1999.
38. Rui Nian, Jinfeng Liu, and Biao Huang. A review on reinforcement learning: Introduction and applications in industrial process control. *Computers & Chemical Engineering*, 139:106886, 2020.
39. Yuhu Wu, Xi-Ming Sun, Xudong Zhao, and Tielong Shen. Optimal control of boolean control networks with average cost: A policy iteration approach. *Automatica*, 100:378–387, 2019.
40. Biao Luo, Yin Yang, and Derong Liu. Policy iteration q-learning for data-based two-player zero-sum game of linear discrete-time systems. *IEEE Transactions on Cybernetics*, 51(7):3630–3640, 2020.
41. Elton Pan, Panagiotis Petsagkourakis, Max Mowbray, Dongda Zhang, and Ehecatl Antonio del Rio-Chanona. Constrained model-free reinforcement learning for process optimization. *Computers & Chemical Engineering*, 154: 107462, 2021.
42. Mohsen Paniri, Mohammad Bagher Dowlatshahi, and Hossein Nezamabadi-pour. Ant-td: Ant colony optimization plus temporal difference reinforcement learning for multi-label feature selection. *Swarm and Evolutionary Computation*, 64:100892, 2021.
43. Jacob Buckman, Danijar Hafner, George Tucker, Eugene Brevdo, and Honglak Lee. Sample-efficient reinforcement learning with stochastic ensemble value expansion. *Advances in neural information processing systems*, 31, 2018.
44. Tayyab Khan, Karan Singh, Mohd Shariq, Khaleel Ahmad, KS Savita, Ali Ahmadian, Soheil Salahshour, and Mauro Conti. An efficient trust-based decision-making approach for wsns: Machine learning oriented approach. *Computer Communications*, 209:217–229, 2023.
45. Boutaina EL Kinany, Mohamed Alfidi, and Zakaria Chalh. Fuzzy logic control for balancing a two-armed inverted pendulum. *Statistics, Optimization & Information Computing*, 11(1):136–142, 2023.
46. Armando de Jesús Plasencia-Salgueiro. Deep reinforcement learning for autonomous mobile robot navigation. In *Artificial Intelligence for Robotics and Autonomous Systems Applications*, pages 195–237. Springer, 2023.

47. Saurabh Arora and Prashant Doshi. A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297:103500, 2021.
48. Rongrong Liu, Florent Nageotte, Philippe Zanne, Michel de Mathelin, and Birgitta Dresp-Langley. Deep reinforcement learning for the control of robotic manipulation: a focussed mini-review. *Robotics*, 10(1):22, 2021.
49. Dong Han, Beni Mulyana, Vladimir Stankovic, and Samuel Cheng. A survey on deep reinforcement learning algorithms for robotic manipulation. *Sensors*, 23(7):3762, 2023.
50. Tuomas Haarnoja, Vitchyr Pong, Aurick Zhou, Murtaza Dalal, Pieter Abbeel, and Sergey Levine. Composable deep reinforcement learning for robotic manipulation. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 6244–6251. IEEE, 2018.
51. Julian Ibarz, Jie Tan, Chelsea Finn, Mrinal Kalakrishnan, Peter Pastor, and Sergey Levine. How to train your robot with deep reinforcement learning: lessons we have learned. *The International Journal of Robotics Research*, 40(4-5): 698–721, 2021.
52. Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38, 2017.
53. Niloofar Khanzhahi, Behrouz Masoumi, and Babak Karasfi. Deep reinforcement learning issues and approaches for the multi-agent centric problems. In *2018 9th Conference on Artificial Intelligence and Robotics and 2nd Asia-Pacific International Symposium*, pages 87–95. IEEE, 2018.
54. SS Mousavi, M Schukat, and E Howley. Deep reinforcement learning: an overview. lect notes netw syst 16: 426–440, 2018.
55. Konstantinos Chatzilygeroudis, Vassilis Vassiliades, Freek Stulp, Sylvain Calinon, and Jean-Baptiste Mouret. A survey on policy search algorithms for learning robot controllers in a handful of trials. *IEEE Transactions on Robotics*, 36(2): 328–347, 2019.
56. Sarthak Bhagat, Hritwick Banerjee, Zion Tsz Ho Tse, and Hongliang Ren. Deep reinforcement learning for soft, flexible robots: Brief review with impending challenges. *Robotics*, 8(1):4, 2019.
57. Guoguang Du, Kai Wang, Shiguo Lian, and Kaiyong Zhao. Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review. *Artificial Intelligence Review*, 54(3):1677–1734, 2021.
58. Matthew Connolly, Aswin K Ramasubramanian, Matthew Kelly, Jack McEvoy, and Nikolaos Papakostas. Realistic simulation of robotic grasping tasks: review and application. *Procedia CIRP*, 104:1704–1709, 2021.
59. Qaid Mohammed Marwan, Shing Chyi Chua, and Lee Chung Kwek. Comprehensive review on reaching and grasping of objects in robotics. *Robotica*, 39(10):1849–1882, 2021.
60. Xu Wang, Sen Wang, Xingxing Liang, Dawei Zhao, Jincai Huang, Xin Xu, Bin Dai, and Qiguang Miao. Deep reinforcement learning: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
61. Jeffrey Mahler and Ken Goldberg. Learning deep policies for robot bin picking by simulating robust grasping sequences. In *Conference on robot learning*, pages 515–524. PMLR, 2017.
62. Douglas Morrison, Peter Corke, and Jürgen Leitner. Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach. *arXiv preprint arXiv:1804.05172*, 2018.
63. Andy Zeng, Shuran Song, Stefan Welker, Johnny Lee, Alberto Rodriguez, and Thomas Funkhouser. Learning synergies between pushing and grasping with self-supervised deep reinforcement learning. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4238–4245. IEEE, 2018.
64. Tom Silver, Kelsey Allen, Josh Tenenbaum, and Leslie Kaelbling. Residual policy learning. *arXiv preprint arXiv:1812.06298*, 2018.
65. Long Kang, Jong-Tae Seo, Sang-Hwa Kim, Wan-Ju Kim, and Byung-Ju Yi. Design and implementation of a multi-function gripper for grasping general objects. *Applied Sciences*, 9(24):5266, 2019.
66. Lars Berscheid, Pascal Meißner, and Torsten Kröger. Robot learning of shifting objects for grasping in cluttered environments. In *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 612–618. IEEE, 2019.
67. Shanshan Zhu, Xiaoxiang Zheng, Ming Xu, Zhiwen Zeng, and Hui Zhang. A robotic semantic grasping method for pick-and-place tasks. In *2019 Chinese Automation Congress (CAC)*, pages 4130–4136. IEEE, 2019.

68. Adithyavairavan Murali, Arsalan Mousavian, Clemens Eppner, Chris Paxton, and Dieter Fox. 6-dof grasping for target-driven object manipulation in clutter. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6232–6238. IEEE, 2020.
69. Quanquan Shao, Jie Hu, Weiming Wang, Yi Fang, Wenhai Liu, Jin Qi, and Jin Ma. Suction grasp region prediction using self-supervised learning for object picking in dense clutter. In *2019 IEEE 5th International Conference on Mechatronics System and Robots (ICMSR)*, pages 7–12. IEEE, 2019.
70. Iason Sarantopoulos, Marios Kiatos, Zoe Doulgeri, and Sotiris Malassiotis. Split deep q-learning for robust object singulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6225–6231. IEEE, 2020.
71. Bohan Wu, Iretiayo Akinola, Abhi Gupta, Feng Xu, Jacob Varley, David Watkins-Valls, and Peter K Allen. Generative attention learning: a "general" framework for high-performance multi-fingered grasping in clutter. *Autonomous Robots*, 44(6):971–990, 2020.
72. Shirin Joshi, Sulabh Kumra, and Ferat Sahin. Robotic grasping using deep reinforcement learning. In *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*, pages 1461–1466. IEEE, 2020.
73. Taewon Kim, Yeseong Park, Youngbin Park, and Il Hong Suh. Acceleration of actor-critic deep reinforcement learning for visual grasping in clutter by state representation learning based on disentanglement of a raw input image. *arXiv preprint arXiv:2002.11903*, 2020.
74. Kechun Xu, Hongxiang Yu, Qianen Lai, Yue Wang, and Rong Xiong. Efficient learning of goal-oriented push-grasping synergy in clutter. *IEEE Robotics and Automation Letters*, 6(4):6337–6344, 2021.
75. Bingjie Tang, Matthew Corsaro, George Konidaris, Stefanos Nikolaidis, and Stefanie Tellex. Learning collaborative pushing and grasping policies in dense clutter. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6177–6184. IEEE, 2021.
76. Huixu Dong, Dilip K Prasad, and I-Ming Chen. Object pose estimation via pruned hough forest with combined split schemes for robotic grasp. *IEEE Transactions on Automation Science and Engineering*, 18(4):1814–1821, 2020.
77. Marwan Qaid Mohammed, Lee Chung Kwek, Shing Chyi Chua, and Esmail Ai Alandoli. Color matching based approach for robotic grasping. In *2021 International Congress of Advanced Technology and Engineering (ICOTEN)*, pages 1–8. IEEE, 2021.
78. Martin Sundermeyer, Arsalan Mousavian, Rudolph Triebel, and Dieter Fox. Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13438–13444. IEEE, 2021.
79. Yuhong Deng, Xiaofeng Guo, Yixuan Wei, Kai Lu, Bin Fang, Di Guo, Huaping Liu, and Fuchun Sun. Deep reinforcement learning for robotic pushing and picking in cluttered environment. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 619–626. IEEE, 2019.
80. Kamal Mokhtar, Cock Heemskerk, and Hamidreza Kasaei. Self-supervised learning for joint pushing and grasping policies in highly cluttered environments. *arXiv preprint arXiv:2203.02511*, 2022.
81. Nils Dengler, David Großklaus, and Maren Bennewitz. Learning goal-oriented non-prehensile pushing in cluttered scenes. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1116–1122. IEEE, 2022.
82. Muhammad Babar Imtiaz, Yuansong Qiao, and Brian Lee. Prehensile and non-prehensile robotic pick-and-place of objects in clutter using deep reinforcement learning. *Sensors*, 23(3):1513, 2023.
83. Stephen James, Paul Wohlhart, Mrinal Kalakrishnan, Dmitry Kalashnikov, Alex Irpan, Julian Ibarz, Sergey Levine, Raia Hadsell, and Konstantinos Bousmalis. Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12627–12637, 2019.
84. Bohan Wu, Iretiayo Akinola, and Peter K Allen. Pixel-attentive policy gradient for multi-fingered grasping in cluttered scenes. In *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 1789–1796. IEEE, 2019.
85. Meng Fang, Cheng Zhou, Bei Shi, Boqing Gong, Jia Xu, and Tong Zhang. Dher: Hindsight experience replay for dynamic goals. In *International Conference on Learning Representations*, 2019.

86. Alexander Irpan, Kanishka Rao, Konstantinos Bousmalis, Chris Harris, Julian Ibarz, and Sergey Levine. Off-policy evaluation via off-policy classification. *Advances in Neural Information Processing Systems*, 32, 2019.
87. Bohan Wu, Iretiayo Akinola, Jacob Varley, and Peter Allen. Mat: Multi-fingered adaptive tactile grasping via deep reinforcement learning. *arXiv preprint arXiv:1909.04787*, 2019.
88. Lin Shao, Fabio Ferreira, Mikael Jorda, Varun Nambiar, Jianlan Luo, Eugen Solowjow, Juan Aparicio Ojea, Oussama Khatib, and Jeannette Bohg. Unigrasp: Learning a unified model to grasp with multifingered robotic hands. *IEEE Robotics and Automation Letters*, 5(2):2286–2293, 2020.
89. OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.
90. Ignasi Clavera, David Held, and Pieter Abbeel. Policy transfer via modularity and reward guiding. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1537–1544. IEEE, 2017.
91. Kanishka Rao, Chris Harris, Alex Irpan, Sergey Levine, Julian Ibarz, and Mohi Khansari. Rl-cyclegan: Reinforcement learning aware simulation-to-real. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11157–11166, 2020.
92. Daniel Ho, Kanishka Rao, Zhuo Xu, Eric Jang, Mohi Khansari, and Yunfei Bai. Retinagan: An object-aware approach to sim-to-real transfer. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10920–10926. IEEE, 2021.
93. Zihan Ding, Ya-Yen Tsai, Wang Wei Lee, and Bidan Huang. Sim-to-real transfer for robotic manipulation with tactile sensory. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6778–6785. IEEE, 2021.
94. Alex X Lee, Coline Manon Devin, Yuxiang Zhou, Thomas Lampe, Konstantinos Bousmalis, Jost Tobias Springenberg, Arunkumar Byravan, Abbas Abdolmaleki, Nimrod Gileadi, David Khosid, et al. Beyond pick-and-place: Tackling robotic stacking of diverse shapes. In *5th Annual Conference on Robot Learning*, 2021.
95. Ole-Magnus Pedersen, Ekrem Misimi, and François Chaumette. Grasping unknown objects by coupling deep reinforcement learning, generative adversarial networks, and visual servoing. In *2020 IEEE international conference on robotics and automation (ICRA)*, pages 5655–5662. IEEE, 2020.
96. Dafa Ren, Shuang Wu, Xiaofan Wang, Yan Peng, and Xiaoqiang Ren. Learning bifunctional push-grasping synergistic strategy for goal-agnostic and goal-oriented tasks. *arXiv preprint arXiv:2212.01763*, 2022.
97. Carlo Rizzardo, Fei Chen, and Darwin Caldwell. Sim-to-real via latent prediction: Transferring visual non-prehensile manipulation policies. *Frontiers in Robotics and AI*, 9, 2022.
98. Samarth Brahmbhatt, Ankur Deka, Andrew Spielberg, and Matthias Müller. Zero-shot transfer of haptics-based object insertion policies. *arXiv preprint arXiv:2301.12587*, 2023.
99. Allen Z Ren, Hongkai Dai, Benjamin Burchfiel, and Anirudha Majumdar. Adaptsim: Task-driven simulation adaptation for sim-to-real transfer. *arXiv preprint arXiv:2302.04903*, 2023.
100. Dezhong Tong, Andrew Choi, Demetri Terzopoulos, Jungseock Joo, and M Khalid Jawed. Deep learning of force manifolds from the simulated physics of robotic paper folding. *arXiv preprint arXiv:2301.01968*, 2023.
101. Zuyuan Zhu and Huosheng Hu. Robot learning from demonstration in robotic assembly: A survey. *Robotics*, 7(2):17, 2018.
102. Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, 50(2):1–35, 2017.
103. Chelsea Finn, Sergey Levine, and Pieter Abbeel. Guided cost learning: Deep inverse optimal control via policy optimization. In *International conference on machine learning*, pages 49–58. PMLR, 2016.
104. Gerrit Schoettler, Ashvin Nair, Jianlan Luo, Shikhar Bahl, Juan Aparicio Ojea, Eugen Solowjow, and Sergey Levine. Deep reinforcement learning for industrial insertion tasks with visual inputs and natural rewards. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5548–5555. IEEE, 2020.
105. Yuke Zhu, Ziyu Wang, Josh Merel, Andrei Rusu, Tom Erez, Serkan Cabi, Saran Tunyasuvunakool, János Kramár, Raia Hadsell, Nando de Freitas, et al. Reinforcement and imitation learning for diverse visuomotor skills. *arXiv preprint arXiv:1802.09564*, 2018.

106. Matteo Ragaglia et al. Robot learning from demonstrations: Emulation learning in environments with moving obstacles. *Robotics and autonomous systems*, 101:45–56, 2018.
107. Pooya Abolghasemi and Ladislau Bölöni. Accept synthetic objects as real: End-to-end training of attentive deep visuomotor policies for manipulation in clutter. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6506–6512. IEEE, 2020.
108. Shuran Song, Andy Zeng, Johnny Lee, and Thomas Funkhouser. Grasping in the wild: Learning 6dof closed-loop grasping from low-cost demonstrations. *IEEE Robotics and Automation Letters*, 5(3):4978–4985, 2020.
109. Pratyusha Sharma, Lekha Mohan, Lerrel Pinto, and Abhinav Gupta. Multiple interactions made easy (mime): Large scale demonstrations data for imitation. In *Conference on robot learning*, pages 906–915. PMLR, 2018.
110. Jaeseok Kim, Anand Kumar Mishra, Raffaele Limosani, Marco Scafuro, Nino Cauli, Jose Santos-Victor, Barbara Mazzolai, and Filippo Cavallo. Control strategies for cleaning robots in domestic applications: A comprehensive review. *International Journal of Advanced Robotic Systems*, 16(4):1729881419857432, 2019.
111. Shuo Yang, Wei Zhang, Weizhi Lu, Hesheng Wang, and Yibin Li. Learning actions from human demonstration video for robotic manipulation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1805–1811. IEEE, 2019.
112. Ozsel Kilinc and Giovanni Montana. Reinforcement learning for robotic manipulation using simulated locomotion demonstrations. *Machine Learning*, pages 1–22, 2022.
113. Laura Smith, Nikita Dhawan, Marvin Zhang, Pieter Abbeel, and Sergey Levine. Avid: Learning multi-stage tasks via pixel-level translation of human videos. *arXiv preprint arXiv:1912.04443*, 2019.
114. Asad Ali Shahid, Loris Roveda, Dario Piga, and Francesco Braghin. Learning continuous control actions for robotic grasping with reinforcement learning. In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 4066–4072. IEEE, 2020.
115. Aran Sena and Matthew Howard. Quantifying teaching behavior in robot learning from demonstration. *The International Journal of Robotics Research*, 39(1):54–72, 2020.
116. Ekrem Misimi, Alexander Olofsson, Aleksander Eilertsen, Elling Ruud Øye, and John Reidar Mathiassen. Robotic handling of compliant food objects by robust learning from demonstration. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6972–6979. IEEE, 2018.
117. Harish Ravichandar, Athanasios S Polydoros, Sonia Chernova, and Aude Billard. Recent advances in robot learning from demonstration. *Annual review of control, robotics, and autonomous systems*, 3:297–330, 2020.
118. Ci-Jyun Liang, VR Kamat, and CC Menassa. Teaching robots to perform construction tasks via learning from demonstration. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, volume 36, pages 1305–1311. IAARC Publications, 2019.
119. Gokhan Solak and Lorenzo Jamone. Learning by demonstration and robust control of dexterous in-hand robotic manipulation skills. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8246–8251. IEEE, 2019.
120. Luca Marzari, Ameya Pore, Diego Dall'Alba, Gerardo Aragon-Camarasa, Alessandro Farinelli, and Paolo Fiorini. Towards hierarchical task decomposition using deep reinforcement learning for pick and place subtasks. In *2021 20th International Conference on Advanced Robotics (ICAR)*, pages 640–645. IEEE, 2021.
121. Stephen James, Kentaro Wada, Tristan Laidlow, and Andrew J Davison. Coarse-to-fine q-attention: Efficient learning for visual robotic manipulation via discretisation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13739–13748, 2022.
122. Peide Cai, Hengli Wang, Huaiyang Huang, Yuxuan Liu, and Ming Liu. Vision-based autonomous car racing using deep imitative reinforcement learning. *IEEE Robotics and Automation Letters*, 6(4):7262–7269, 2021.
123. Akshay Dhonthi, Philipp Schillinger, Leonel Rozo, and Daniele Nardi. Optimizing demonstrated robot manipulation skills for temporal logic constraints. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1255–1262. IEEE, 2022.
124. Albert Yu and Raymond J Mooney. Using both demonstrations and language instructions to efficiently learn robotic tasks. *arXiv preprint arXiv:2210.04476*, 2022.

125. Onur Berk Tore, Farzin Negahbani, and Baris Akgun. Keyframe demonstration seeded and bayesian optimized policy search. *arXiv preprint arXiv:2301.08184*, 2023.
126. Armin Avaei, Linda van der Spaa, Luka Peternel, and Jens Kober. An incremental inverse reinforcement learning approach for motion planning with human preferences. *arXiv preprint arXiv:2301.10528*, 2023.
127. Adarsh Sehgal, Hung La, Sushil Louis, and Hai Nguyen. Deep reinforcement learning using genetic algorithm for parameter optimization. In *2019 Third IEEE International Conference on Robotic Computing (IRC)*, pages 596–601. IEEE, 2019.
128. Tuomas Haarnoja, Vitchyr Pong, Aurick Zhou, Murtaza Dalal, Pieter Abbeel, and Sergey Levine. Composable deep reinforcement learning for robotic manipulation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6244–6251, 2018. doi: 10.1109/ICRA.2018.8460756.
129. Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. *arXiv preprint arXiv:1806.10293*, 2018.
130. Menghao Wu, Yanbin Gao, Alexander Jung, Qiang Zhang, and Shitong Du. The actor-dueling-critic method for reinforcement learning. *Sensors*, 19(7), 2019. ISSN 1424-8220. doi: 10.3390/s19071547. URL https://www.mdpi.com/1424-8220/19/7/1547.
131. Lishan Lin, Yuji Yang, Hui Cheng, and Xuechen Chen. Autonomous vision-based aerial grasping for rotorcraft unmanned aerial vehicles. *Sensors*, 19(15):3410, 2019.
132. Cristian Bodnar, Adrian Li, Karol Hausman, Peter Pastor, and Mrinal Kalakrishnan. Quantile qt-opt for risk-aware vision-based robotic grasping. *arXiv preprint arXiv:1910.02787*, 2019.
133. Taisuke Kobayashi, Takumi Aotani, Julio Rogelio Guadarrama-Olvera, Emmanuel Dean-Leon, and Gordon Cheng. Reward-punishment actor-critic algorithm applying to robotic non-grasping manipulation. In *2019 Joint IEEE 9th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 37–42, 2019. doi: 10.1109/DEVLRN.2019.8850699.
134. Satonori Demura, Kazuki Sano, Wataru Nakajima, Kotaro Nagahama, Keisuke Takeshita, and Kimitoshi Yamazaki. Picking up one of the folded and stacked towels by a single arm robot. In *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1551–1556, 2018. doi: 10.1109/ROBIO.2018.8665040.
135. Ryan Julian, Benjamin Swanson, Gaurav S Sukhatme, Sergey Levine, Chelsea Finn, and Karol Hausman. Never stop learning: The effectiveness of fine-tuning in robotic reinforcement learning. *arXiv preprint arXiv:2004.10190*, 2020.
136. Yaoxian Song, Yun Luo, and Changbin Yu. Tactile–visual fusion based robotic grasp detection method with a reproducible sensor. *International Journal of Computational Intelligence Systems*, 14(1):1753–1762, 2021.
137. Rajkumar Muthusamy, Xiaoqian Huang, Yahya Zweiri, Lakmal Seneviratne, and Dongming Gan. Neuromorphic event-based slip detection and suppression in robotic grasping and manipulation. *IEEE Access*, 8:153364–153384, 2020. doi: 10.1109/ACCESS.2020.3017738.
138. Zhixin Chen, Mengxiang Lin, Zhixin Jia, and Shibo Jian. Towards generalization and data efficient learning of deep robotic grasping. *CoRR*, abs/2007.00982, 2020. URL https://arxiv.org/abs/2007.00982.
139. Deirdre Quillen, Eric Jang, Ofir Nachum, Chelsea Finn, Julian Ibarz, and Sergey Levine. Deep reinforcement learning for vision-based robotic grasping: A simulated comparative evaluation of off-policy methods. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6284–6291. IEEE, 2018.
140. Jostein René Danielsen. Vision-based robotic grasping in simulation using deep reinforcement learning. Master's thesis, UiT Norges arktiske universitet, 2021.
141. Kilian Kleeberger, Richard Bormann, Werner Kraus, and Marco F Huber. A survey on learning-based robotic grasping. *Current Robotics Reports*, 1(4):239–249, 2020.
142. Weiwei Liu, Linpeng Peng, Junjie Cao, Xiaokuan Fu, Yong Liu, Zaisheng Pan, and Jian Yang. Ensemble bootstrapped deep deterministic policy gradient for vision-based robotic grasping. *IEEE Access*, 9:19916–19925, 2021.
143. Raphael Grimm, Markus Grotz, Simon Ottenhaus, and Tamim Asfour. Vision-based robotic pushing and grasping for stone sample collection under computing resource constraints. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6498–6504, 2021. doi: 10.1109/ICRA48506.2021.9560889.

144. Stephen James and Andrew J Davison. Q-attention: Enabling efficient learning for vision-based robotic manipulation. *IEEE Robotics and Automation Letters*, 7(2):1612–1619, 2022.
145. Taewon Kim, Yeseong Park, Youngbin Park, Sang Hyoung Lee, and Il Hong Suh. Acceleration of actor-critic deep reinforcement learning for visual grasping by state representation learning based on a preprocessed input image. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 198–205. IEEE, 2021.
146. Hu Cao, Guang Chen, Zhijun Li, Yingbai Hu, and Alois Knoll. Neurograsp: multimodal neural network with euler region regression for neuromorphic vision-based grasp pose estimation. *IEEE Transactions on Instrumentation and Measurement*, 71:1–11, 2022.
147. Yan Wang, Gautham Vasan, and A Rupam Mahmood. Real-time reinforcement learning for vision-based robotics utilizing local and remote computers. *arXiv preprint arXiv:2210.02317*, 2022.
148. Anil Kurkcu, Cihan Acar, Domenico Campolo, and Keng Peng Tee. Glocal: Glocalized curriculum-aided learning of multiple tasks with application to robotic grasping. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7089–7096. IEEE, 2021.
149. Andrej Orsula, Simon Bøgh, Miguel Olivares-Mendez, and Carol Martinez. Learning to grasp on the moon from 3d octree observations with deep reinforcement learning. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4112–4119. IEEE, 2022.
150. Sudhir Pratap Yadav, Rajendra Nagar, and Suril V Shah. Learning vision-based robotic manipulation tasks sequentially in offline reinforcement learning settings. *arXiv e-prints*, pages arXiv–2301, 2023.
151. Shixiang Gu, Ethan Holly, Timothy Lillicrap, and Sergey Levine. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3389–3396, 2017. doi: 10.1109/ICRA.2017.7989385.
152. Rongrong Liu, Florent Nageotte, Philippe Zanne, Michel de Mathelin, and Birgitta Dresp-Langley. Deep reinforcement learning for the control of robotic manipulation: A focussed mini-review. *Robotics*, 10(1), 2021. ISSN 2218-6581. doi: 10.3390/robotics10010022. URL https://www.mdpi.com/2218-6581/10/1/22.
153. Xiaoqian Huang, Mohamad Halwani, Rajkumar Muthusamy, Abdulla Ayyad, Dewald Swart, Lakmal Seneviratne, Dongming Gan, and Yahya Zweiri. Real-time grasping strategies using event camera. *J. Intell. Manuf.*, 33(2):593–615, February 2022.
154. Matthias Kerzel, Hadi Beik Mohammadi, Mohammad Ali Zamani, and Stefan Wermter. Accelerating deep continuous reinforcement learning through task simplification. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6, 2018. doi: 10.1109/IJCNN.2018.8489712.
155. Lirui Wang, Yu Xiang, Wei Yang, Arsalan Mousavian, and Dieter Fox. Goal-auxiliary actor-critic for 6d robotic grasping with point clouds. In Aleksandra Faust, David Hsu, and Gerhard Neumann, editors, *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pages 70–80. PMLR, 08–11 Nov 2022. URL https://proceedings.mlr.press/v164/wang22a.html.
156. Yongchao Wang, Xuguang Lan, Chuzhen Feng, Lipeng Wan, Jin Li, Yuwang Liu, and Decai Li. An experience-based policy gradient method for smooth manipulation. In *2019 IEEE 9th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 93–97, 2019. doi: 10.1109/CYBER46603.2019.9066580.
157. Guangjun Qi and Yuan Li. Reinforcement learning control for robot arm grasping based on improved ddpg. In *2021 40th Chinese Control Conference (CCC)*, pages 4132–4137, 2021. doi: 10.23919/CCC52363.2021.9550413.
158. Hadi Beik Mohammadi, Mohammad Ali Zamani, Matthias Kerzel, and Stefan Wermter. Mixed-reality deep reinforcement learning for a reach-to-grasp task. In Igor V. Tetko, Věra Kůrková, Pavel Karpov, and Fabian Theis, editors, *Artificial Neural Networks and Machine Learning – ICANN 2019: Theoretical Neural Computation*, pages 611–623, Cham, 2019. Springer International Publishing. ISBN 978-3-030-30487-4.
159. David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 387–395, Bejing, China, 22–24 Jun 2014. PMLR. URL https://proceedings.mlr.press/v32/silver14.html.

160. Qiang He and Xinwen Hou. Wd3: Taming the estimation bias in deep reinforcement learning. In *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 391–398, 2020. doi: 10.1109/ICTAI50040.2020.00068.
161. Stephen Dankwa and Wenfeng Zheng. Twin-delayed ddpg: A deep reinforcement learning technique to model a continuous movement of an intelligent robot agent. In *Proceedings of the 3rd International Conference on Vision, Image and Signal Processing*, ICVISP 2019, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450376259. doi: 10.1145/3387168.3387199. URL https://doi.org/10.1145/3387168.3387199.
162. A.T.D. Perera and Parameswaran Kamalaruban. Applications of reinforcement learning in energy systems. *Renewable and Sustainable Energy Reviews*, 137:110618, 2021. ISSN 1364-0321. doi: https://doi.org/10.1016/j.rser.2020.110618. URL https://www.sciencedirect.com/science/article/pii/S1364032120309023.
163. Árpád Fehér, Szilárd Aradi, and Tamás Bécsi. Hierarchical evasive path planning using reinforcement learning and model predictive control. *IEEE Access*, 8:187470–187482, 2020. doi: 10.1109/ACCESS.2020.3031037.
164. Yangyang Hou, Huajie Hong, Zhaomei Sun, Dasheng Xu, and Zhe Zeng. The control method of twin delayed deep deterministic policy gradient with rebirth mechanism to multi-dof manipulator. *Electronics*, 10(7), 2021. ISSN 2079-9292. doi: 10.3390/electronics10070870. URL https://www.mdpi.com/2079-9292/10/7/870.
165. Dhuruva Priyan G. M, Abhik Singla, and Shalabh Bhatnagar. Hindsight experience replay with kronecker product approximate curvature. *CoRR*, abs/2010.06142, 2020. URL https://arxiv.org/abs/2010.06142.
166. Phan Bui Khoi, Hanoi University of Science and Technology, 01 Dai Co Viet, Hai Ba Trung, Hanoi, 100000, Vietnam, Nguyen Truong Giang, and Hoang Van Tan. Control and simulation of a 6-DOF biped robot based on twin delayed deep deterministic policy gradient algorithm. *Indian J. Sci. Technol.*, 14(31):2460–2471, July 2021.
167. Yichu Yang and Wei Xu. *Development of Warehouse Robot Arms for Grasping Objects*.
168. Rui Nian, Jinfeng Liu, and Biao Huang. A review on reinforcement learning: Introduction and applications in industrial process control. *Comput. Chem. Eng.*, 139(106886):106886, August 2020.
169. Pengzhan Chen and Weiqing Lu. Deep reinforcement learning based moving object grasping. *Inf. Sci. (Ny)*, 565: 62–76, July 2021.
170. Zohar Feldman, Hanna Ziesche, Ngo Anh Vien, and Dotan Di Castro. A hybrid approach for learning to shift and grasp with elaborate motion primitives. In *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, May 2022.
171. Asad Ali Shahid, Dario Piga, Francesco Braghin, and Loris Roveda. Continuous control actions learning and adaptation for robotic manipulation through reinforcement learning. *Auton. Robots*, 46(3):483–498, March 2022.
172. Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. January 2018.
173. Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, and Sergey Levine. Soft actor-critic algorithms and applications. December 2018.
174. Kazım Ünal. *A comparative study of deep reinforcement learning methods and conventional controllers for aerial manipulation*. 2021.
175. Michel Breyer. *Flexible robotic grasping with sim-to-real transfer based reinforcement learning*. 2018.
176. Wenshuai Zhao, Jorge Peña Queralta, Li Qingqing, and Tomi Westerlund. Towards closing the sim-to-real gap in collaborative multi-robot deep reinforcement learning. In *2020 5th International Conference on Robotics and Automation Engineering (ICRAE)*, pages 7–12, 2020. doi: 10.1109/ICRAE50850.2020.9310796.
177. Yiwen Chen, Zhaojie Ju, and Chenguang Yang. Combining reinforcement learning and rule-based method to manipulate objects in clutter. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6, 2020. doi: 10.1109/IJCNN48605.2020.9207153.
178. Arturo Cruz-Maya. Target reaching behaviour for unfreezing the robot in a semi-static and crowded environment. December 2020.
179. Ching-Chang Wong, Shao-Yu Chien, Hsuan-Ming Feng, and Hisasuki Aoyama. Motion planning for dual-arm robot based on soft actor-critic. *IEEE Access*, 9:26871–26885, 2021. doi: 10.1109/ACCESS.2021.3056903.