IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 29, No. 6, December, 2024, pp. 2219-2229
Journal homepage: http://iieta.org/journals/isi

# Waste Objects Segregation Using Deep Reinforcement Learning with Deep Q Networks

Nida Khan, Kunal Kulkarni, Yashashree Mahale, Shrikrishna Kolhar*, Smita Mahajan

Symbiosis Institute of Technology, Symbiosis International (Deemed University), Pune 412115, India

Corresponding Author Email: shrikrishna.kolhar@sitpune.edu.in



https://doi.org/10.18280/isi.290612

**Received:** 31 August 2024
**Revised:** 30 October 2024
**Accepted:** 19 November 2024
**Available online:** 25 December 2024

***Keywords:***
*Deep Q networks, image classification, reinforcement learning, waste classification, waste management*

**ABSTRACT**

Effective waste classification is critical in addressing the rising environmental pollution and waste volume. Conventional sorting methods are labor-intensive and error-prone, particularly with the increasing diversity of waste materials. This study presents an innovative approach using deep reinforcement learning for waste object detection and classification to automate waste management processes. The proposed system aims to boost operational efficiency, enhance resource recovery, and reduce waste going to landfills by leveraging deep reinforcement learning. The Deep Q Network model proposed achieved an accuracy of approximately 73%. By employing DQN, an advanced reinforcement learning algorithm, the system ensures improved waste object image classification to handle complex tasks with distributional characteristics. This study can be further extended to design and develop an autonomous waste sorting system.

## 1. INTRODUCTION

Increasing pollution concerning biodiversity, environmental degradation, and waste management is a significant area of research. The global waste issue requires innovative approaches to reduce environmental pressures, and more resources are needed. Automatic and proper waste classification is at the heart of the problem, as traditionally, waste classification is done manually and is a dirty and complex job. In addition, different waste quantities require precise recycling and disposal sorting [1]. Here is an automatic garbage classification method that adopts a new image classification method based on deep reinforcement learning to generate automatic garbage classification. The motivation behind this research comes from the utmost need to perform better waste management and the environmental burden is reduced. With increasing urbanization and consumption, traditional sorting methods are becoming unmanageable in the face of increasing types of waste. Automated sorting methods will increase operational efficiency and enable better segregation, resulting in better recovery and waste diversion from landfills.

Deep reinforcement learning (DRL) [2, 3] is a powerful capability for rapidly handling complex classification tasks. Apart from the fact that DRL is a machine learning approach, it has several unique advantages compared to traditional machine learning approaches. DRL combines deep learning and reinforcement learning techniques by which an agent can learn optimal strategies while interacting with a certain environment. Regarding the classification task, DRL deals with unstructured high-dimensional data; images, text, and audio are no exceptions [4]. In such cases, it may be challenging for the traditional methods to capture the intricate patterns and relationships. Among the main strengths of DRL in this classification task is its ability to learn hierarchical representations of data. Classification tasks are further implemented via the available deep neural networks, which automatically extract the desired features from raw inputs. Thus, the DRL models can distinguish subtle differences between classes. Consequently, the DRL models improve classification accuracy and enhance their invariance to variations and noise in data.

This study introduces a new method combining reinforcement learning with computer vision to realize autonomous waste sorting [5]. Unlike the previous tradition-based methods using handcrafted features or supervised learning, our method directly learns from raw image data and interactions with the waste sorting environment. Through RL, we model the waste sorting problem as a sequential decision-making problem in which our system autonomously recognizes and sorts the objects of waste into the most fitting bins. The innovation of this work lies in the application of RL to the field of waste management, where we face the challenges involved in the real sorting environment. Moreover, the proposed method has several advantages over traditional methods. Using RL helps adapt dynamically to changes in conditions and various waste items to achieve high accuracy and efficiency in sorting. In addition, since our model learns from interaction with the environment, it is efficient in generalizing unseen waste materials and variations of sorting setups. Hence this study integrates deep learning and computer vision to establish the waste classification model, however, most of the existing research focuses on the supervised learning approach that constrains model flexibility in changing circumstance. There is scarce literature on RL methods especially DQLNs, applied to waste classification problems. This study aims to fill the gap with a reinforcement learning structure with feature extraction on a CNN and the DQLN for

battle field decision. This general approach of employing machine learning-based methods means that the model iteratively provides ocular classification enhancement by engaging with a simulated environment and doing so across variably-type wastes and across difficult classes of wastes such as in the case of classing cardboard and paper. Further, we propose a form of structure in the reward function so that the model is capable of learning the best classification policies apart from the training process making the classification training general enough to do well in new data sets.

The main contributions of this study are threefold: Three contributions follow: (1) the combination of reinforcement learning with CNN-based classification to achieve an adaptive learning process in the context of waste segregation; (2) a reward-based approach in order to classify and segregate waste objects that improves model accuracy and insensitivity to variation in waste characteristics; and (3) a comprehensive analysis towards the effectiveness of RL in coping with complex and ambiguous waste types compared to most common supervised methods. Altogether, these contributions enhance the novelty of the approach outlined herein and reflect the utility of the proposed method to the real-world automated waste segregation systems, where flexibility and a high level of accuracy play a determining role in environmental care. The study aims to design and deploy an autonomous waste sorting system using deep reinforcement learning to classify images. The implementation is based on principles where the system learns to make informed decisions using the cues and feedback from the environment to increase the rate of sorting and minimize errors, resulting in improved effectiveness in managing waste. The study focuses on the feasibility and efficacy of reinforcement learning in the waste management domain. Extensive experimentation and evaluation are conducted on a comprehensive waste dataset to illustrate our approach's effectiveness in automating waste sorting tasks. Therefore, this paper lays the foundation for intelligent and sustainable waste management systems using deep reinforcement learning techniques.

## 2. RELATED WORK

For effective waste management and classification, reinforcement learning methods are prevalent. Weerasekara et al. [6] suggested using deep reinforcement learning for inventory management in disassembly systems, targeting issues caused by time-variant inventory fluctuations and high costs. The demonstrated experiments in the disassembly of televisions have confirmed that DRL reduces inventory buildup by 21% and unmet demand by 12%, outperforming other benchmarks, including Multiple Elman Neural Networks. In one study, Rakesh et al. [7] present segregation and collection of medical waste and notification using Internet of Things (IoT) based smart dust bins. In another study, Okafor et al. [8] presented a comparative model-free deep reinforcement learning system for cooperative object sorting in cluttered environments. It combines primitive policies (pushing, grasping, placing) using lightweight deep neural networks, exploring results from 12 custom instances with pixel-wise Q-valued critic networks (PQCN), utilizing backbone networks such as DenseNet121 or MobileNetV3, suggesting that combining these backbones with fully convolutional networks (FCN) and training through dual transfer learning yields the most optimal performance, notably in generalization during testing. This research sets a benchmark for evaluating DRL algorithms in sorting tasks across various industries like manufacturing, construction, and waste management.

A robotic system is used for grasping and recognizing objects in cluttered environments, achieving high success rates through object-agnostic grasping and cross-domain image classification [9]. A study in Coimbatore, India, highlights insufficient electronic waste recycling and stresses the importance of public education, identifying factors influencing recycling behavior and emphasizing legal support for promoting recycling and behavioral change [10]. In another research, machine learning improves municipal trash management by addressing sorting, routing, and real-time monitoring, recognizing materials, forecasting waste production, and identifying operational problems [11], requiring integration with other initiatives and legislative support for sustainability.

One of the studies presents an approach for efficiently utilizing reinforcement learning; the paper introduces two policy networks, CPNet and FPNet, to handle object detection in large images without modifying the detector's architecture. were trained to balance accuracy and maximize low-resolution image usage with a coarse detector. Experiments on xView satellite images show a 2.2×runtime efficiency increase and a 70% reduction in high-resolution image dependency [12]. Additionally, the approach achieves a 40% runtime increase on the Caltech pedestrian dataset. A novel deep reinforcement learning approach enhances object detection on low-quality images without retraining. It uses detection results to create a reward function, improving image quality and recognition. An image enhancement tool chain (IETC) [13] offers flexible algorithm selection. Experimental results validate effectiveness in various challenging environments.

Another approach uses a novel technique that efficiently selects relevant experiences for training agents, enhancing exploration-exploitation trade-off and accelerating learning convergence [14]. Filtering experiences based on state similarity and a hyper-parameter improves future returns across diverse environments. A study presented a goal exploration process - policy gradient (GEP-PG), a novel strategy for reducing exploration inefficiencies in continuous action domains that combines two forms of DDPG with the goal exploration process. Unlike gradient-descent-based techniques like DDPG, which are efficient in fine-tuning policies, evolutionary methods are not as robust in their exploration [15]. Combining these methods, GEP-PG performs better than DDPG variations on the more significant Half-Cheetah benchmark and low-dimensional misleading reward tasks. Another research involves efficient training of agents in multi-agent reinforcement learning (MARL) environments by sampling past experiences from a replay buffer. Filtering samples based on the currently observed state enables quick convergence and mimics human learning processes of generalization [16]. The method generalizes MARL algorithms and analyzes their performance at different samples taken in the learning process. In a multistep progressive image rectification scheme for fisheye images, the problem is treated as a Markov decision process and Deep Q Networks is used as the solution [17].

Most of the conventional supervised learning techniques, as well as basic RL techniques, make decisions in isolation, meaning that a given decision may depend more on previous decisions than on a single input data sample [18]. However, in

comparison to standard deep learning techniques, RL has the advantage of being able to learn over the abstract and the more concrete level, so the model gains a hierarchy view on the classification task. This is important in segregation of waste where, within similar categories of waste, variations can be very high and specific static labels do not guarantee good segregation. Another replacement that works well as a special type of a neural network is a Deep Q Learning (DQL), which het well on simple classic problems of binary classification, but is not hierarchical enough to properly tackle the distinction between different classes of waste. As for the weaknesses of DQL, the approach tends to work more on immediate rewards than on disadvantageous long run outcomes and the deficiencies associated with other dependencies cannot accommodate exceptional systematization of tasks for breaking down of primary tasks or events, thus making it unfit for complex scenarios that require tiered decisions. On the other hand, RL provides a taxonomy of sub-goals that corresponds to both short-term decision and long-term improvements of accuracy. Few advantages of RL techniques are as follows:

- Hierarchical Structuring of Decisions [19]: RL provides structural layers of decision making to the classification model, which makes sense as the hierarchical structure of a classification problem is often as complex as it is in the real world, e.g. above simple metal and non-metal classification, there are sub categories of materials that definitely require recycling but do not fall under any general or special category of recycling mentioned in the paper.
- Enhanced Adaptability [20]: Since sub-goals are used, the model is better suited for changes in waste materials so that in cases where a simple texture or color distinction is not enough, it will have a better accuracy.
- Improved Computational Efficiency: Focusing the learning in the model at multiple levels may lead to improved convergence and decreased computational complexity, thanks to the fact that the learning of more general patterns at higher levels precedes the fine learning in lower levels.

However, RL is somewhat harder for the implementation than the standard RL or supervised learning because of the high-timely structure of RL that needs simultaneous tuning of sub-goal policies to avoid suboptimal decisions. Furthermore, DRL may occasionally fail to meet the temporal complexity of balancing learning between successive layers, especially with limited data for specific sub- objectives and may take time in training the network. Lastly, the attainment of consistent convergence in HRL is slightly more arduous and might call for optimal structures of rewards that correspond with the broad category label.

## 3. DEEP REINFORCEMENT LEARNING FOR WASTE IMAGE CLASSIFICATION

Deep Q-Learning (DQL), a type of deep reinforcement learning (DRL) is employed, along with Convolutional Neural Networks (CNNs) for waste object classification [1, 5]. DQL is a type of Q-learning where Q-value approximations are used to estimate action-value functions and these approximations can be used in deep neural networks, so DQL is good for high dimensional state spaces like images. It is particularly beneficial where the task comes with many visual inputs since CNN takes charge of dig out relevant features from images then DQL component of the model is charged with extracting optimal classification actions from the derived features.

### 3.1 Deep Q-Learning (DQL)

DQL is supposed to gain the maximum end cumulative reward through estimating a Q- function that bring out relations between states, images of wastes. For still image classification, DQL is highly appropriate since it updates the model decisions via exploration-exploitation trade-off policy of epsilon-greedy [21]. This policy is beneficial in the first stages of learning classification of images by the model to begin with before the model refines the strategies that work and avoid those that do not work. In the long run, DQL concentrates more on exploitation, using the actions that have provided the highest rewards in the previous experience. Together with replay memory, this approach is useful in stabilizing the learning process and eliminating high correlation between training samples such that the model generalizes appropriately for different image inputs.

### 3.2 Convolutional neural network (CNN)

CNNs are connected with DQL because they showed promising results in an image classification problem. CNNs use convolutional and pooling layers to minimize image complexity while retaining characteristics that differentiate the various sorts of waste materials [22]. This architecture has multiple convolution layers for obtaining low and high abstraction level representations, with which DQL will improve classifications over time.

DQL is particularly useful for image classification tasks because it incorporates feature extraction, decision making as well as iterated learning through reinforcement in a single model, CNN-DQL. Compared to traditional supervised learning, DRL works on a reinforcement learning structure in which it is awarded a certain sum of reward after each trial, conventionally an integer or a real number; this makes it well suited to a problem that requires a model to gradually fix its mistake as it is tested iteratively. This characteristic becomes particularly helpful for the tasks, such as waste classification, where the contents of classes may have shared visuals and a simple classification method may fail. The DRL model is contextual and learn through the interactions with actual environment hence can improve the classification performance of data inputs that are always unique in real-world applications. This is true since the combination of DQL with CNNs and the hierarchical structure offers significant capabilities for waste classification based on images, while at the same time ensuring that feature-driven image processing is complemented by adaptive learning processes that allow the model to improve its decision-making without a constant reference to the additional data. This choice of algorithm improves model performance in visually complicated environments; therefore, it is suitable and effective when pursuing this work's goals.

### 3.3 Deep Q-Networks (DQN)

In reinforcement learning, DQNs are a powerful class of algorithms used to solve problems where an agent engages with an environment to optimize or maximize performance in

terms of cumulative rewards [23]. Traditionally Q-learning relies on keeping account of all state-action pairings' Q-values in a Q-table. However, the enormous amount of memory required for continuous state or action spaces renders this method infeasible. By approximating the Q-value function using deep neural networks DQNs overcome this constraint [24]. The DNN learns to predict Q-values rather than explicitly storing them. For image-related tasks, the images must first be represented in a form suitable for neural networks. This involves the traditional preprocessing of images, such as resizing them to a fixed size and then normalizing the pixel values.

The architecture of a DQN has input layers, hidden layers, and output layers. The DQN architecture consists of a convolution neural network and one or more fully connected layers. CNNs are effective for processing spatial data like images as they can capture patterns and spatial hierarchies [25]. The input layer takes the state representation of images as input. The multiple layers of neurons in hidden layers learn to approximate the Q-values. The output layer is responsible for producing the Q-values for the possible actions.

The training process for a DQN is focused mainly on the replayed memory, the temporal difference error, and the bellman optimality equation. The agent saves experiences (state, action, reward, next state) in a replay memory buffer [26]. At every time step, the agent selects a batch of experiences from this buffer for training. The loss function for DQN is based on the temporal error. TD error represents the difference between the predicted Q-value and the Q-target, computed using the Bellman equation. The Q-target estimated using the Bellman equation is represented in the Eq. (1) as.

$$Q(s,a) = r + \gamma \, max \, Q(s',a') \tag{1}$$

where,

(s): current state
(a): chosen action
(r): immediate reward
(s'): next state

The highest Q-value for the next state is obtained by forwarding it through the DQNs that can handle continuous state spaces efficiently by approximating the Q-value function using the function approximation [27]. The storing experiences in a replay buffer helps stabilize training. Once the DQN is trained, the network can classify new images by inputting them, analyzing the output, and selecting the class with the highest probability output.

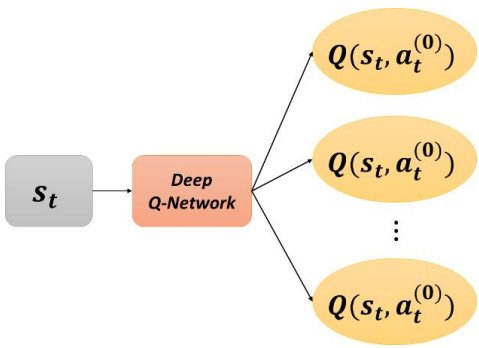


**Figure 1.** Computation of state-action values [28]

A calculation enhancement was suggested, where the Deep Q-network (DQN) agent, as illustrated in Figure 1, computes all the action values for a state at a given moment. Then, by combining it with a ϵ-greedy policy, the neural network that predicted the Q-values for each state-action pair can be utilized to explore the environment arbitrarily and greedily, training a DQN that can predict the values of the state actions.

Primarily, an environment is sampled to collect "experiences" for the DQN. Inspired by psychology research, these sampled events are called episodes attached to a replay memory [29]. Replay memory episodes are retrieved, and a loss function generated from the equation is constructed to update the Q-value estimate as shown in Eq. (2).

$$L(s,a,r,s') = (r + \gamma max_a Q(s',a') - Q(s,a)) \tag{2}$$

where, following action an in-state s', the agent arrives in state $s$. We suppose that in the loss, the Q-value estimation is predicated on choosing the best course of action, with the next state being determined by taking the most significant state-action value among the feasible options.

As the agent keeps assessing the network and the old, less-useful replay memories decay, the training replay memory is continuously fed fresh sampled events. When the memory fills up, the earliest episodes are removed, simulating a first-in, first-out queue to guarantee that only pertinent and recent memories are present in memory for the agent to be taught from. Prioritized experience replay [30] is another innovation that involves prioritizing transitions depending on various measures according to their relevance, such as temporal difference error that they experience, rather than sampling them at random for replay memory during training.

## 4. METHODOLOGY

The dataset and the proposed methodology for waste classification using deep RL are explained in this section.

### 4.1 Dataset description

The dataset used in this study is "Garbage Classification" dataset [31]. This dataset has been obtained from the Kaggle repository which contains images of different waste materials labeled into six classes: cardboard, biological waste, plastic, paper, trash, and metal. All of these come from distinct types of waste that people encounter in their daily lives. It selects a diversified and inclusive dataset where all the images are high-resolution. Images are taken under different environmental conditions and settings. The sample image from every class can be seen in Figure 2. Each image is annotated with its corresponding waste class label. The multi-class classification of waste ensures comprehensive training and model evaluation, ensuring robustness and generalization across diverse types of waste. Throughout the study, a total of 2532 images were used. Below is the breakdown of each class and its characteristics, along with the number of images in each category:

1) Cardboard (393 images): The images of cardboard material, which is generally recyclable and has a specific smooth surface and almost brown hue and has some marks that can easily be identified, like folds and wrinkles.
2) Glass (491 images): Images of glass dictate wastes that contain recycle capacity but have explicit risks of handling due to their fragility. Possible Glass items in the dataset are glass bottles and jars having

translucent or colored body and reflective surfaces in addition to varying translucency which aids in differentiation.

3) Metal (400 images): The metallic wastes consist of waste containers such as cans, tins and any manufactured metallic item. These images tend to take on the characteristic of metals and include mirrored aspects, more specifically, they possess metallic textures that make their surface differ in color as well as in reflectiveness depending on the metal type.
4) Paper (584 images): Paper waste involves newspaper, magazines and any other paper products. This class has the largest number of images, this showing the scale and variety of paper waste. Despite the fact that paper can be easily mistaken for cardboard, it is usually smoother in thickness and is not as rigid, as cardboard is.
5) Plastic (472 images): The images of the plastic class are of bottles, bags, and containers and other products manufactured from the plastic. The plastic materials are ubiquitous in waste and if not sorted appropriately, they are difficult to recycle. Plastic items in images can be of different colors, sizes, even shapes, but the material always has some special shiny appearance and flexibility.
6) Trash (127 images): The trash class consists of different items that are usually unfit to be recycled or are composite waste items that could not be sorted into any of the other classes. Other waste products may include food packaging, used products, and any other waste products that cannot be recycled. A lower number of images is attributed to this class due to the difficulty experienced in categorizing samples of non-recyclable or mixed material waste.


**Figure 2.** Image classes from the dataset

Every class is designed in such a manner that has given the model an understanding of the differences in the form and texture of wastes to enable it classify them properly. This data structure helps the model develop recognition of the difference between recyclable items (Cardboard, glass, metal, paper, plastic) and general waste, which in a certain perspective may require a different method of disposal or recycling. The idea to balance the class distribution, although bins have different numbers of examples, guarantees that the model sees enough samples during the training while retaining the real presentation of such material in usual waste flows. All images were rescaled to 100×100 pixels and then normalized to 0 and 1 to allow the training stage to run faster.

### 4.2 Combined algorithmic approach

This study leverages a multi-algorithm system that integrates CNN and DQLN in a reinforcement leaning system. Each of these has particular tasks which in turn help the model to learn, classify, and improve on decisions pertaining to waste object classification.

4.2.1 Convolutional neural network (CNN): Feature extraction [32]

As in most CNNs, the CNN part in the proposed network acts as a feature extractor, which extracts special and rich features from raw image inputs and helps the classification network to classify the inputs. The CNN structure uses convolutional layers to search for spatial hierarchies of the input image features. This process obtains features for different classes of waste, including the texture, shape, and color, which are necessary for classification between such classes as glass, metal, paper, and plastic. It optimizes the extraction of the features by ensuring that computation time is not exceedingly spent. The last set of layers of the CNN flatten the image representations into vectors that is taken to the fully connected layers. These layers give output values in form of probability values corresponding to each respective class, which is then fed to the DQLN. This specific CNN plays a pivotal part in the program because it helps translate high-dimensional image data into a format that the DQLN can use to learn from through reinforcement techniques.

4.2.2 Deep Q-Learning network (DQLN): Decision-making and action selection [1]

The DQLN is a reinforcement learning algorithm utilized to decide about optimal actions (or classifications) in line with a learned policy. This component allows the model to engage the environment, test the classification action and modify policy in relation to experience gained.

In the case of waste classification, concerning decision-making, the agent chooses actions, which are in our case the classes, that yield the maximum cumulative reward. Reinforcement learning enables the model to act at the given environment by making a classification, and then it receives reinforcement against the classification made which either encourages (by a reward) or discourages (with a penalty) the model to perform better next time. When training, the use of an epsilon greedy policy is implemented where actions are made randomly with a certain probability of epsilon and based on the learnt Q-values with a probability of (1-epsilon). This approach aids the model to learn the right categorization approach instead of limiting its abilities to a few unproductive early categorization techniques while searching for the better approach. The DQLN earns a reward of +1 if a classification is accurate and the nodes lose -1 if the classification is wrong. In the long run, this reward system teaches the agent to always

select action that yields the maximum cumulative reward thus "learning" the right classification.

Compared to other deep learning networks incorporated in the model, the learning from the rewards characteristic contributes to the flexibility of the data inputs in real-life waste classification instances for instance, the object on-grade may differ over some specific time which was not endemic in the means that were used in a static setting.

4.2.3 Integrated CNN-DQLN approach

The interplay between CNN and DQLN brings synergy into the system by allowing feature extraction to be followed by reinforcement-based decision making. Both of them contributes its function in order to form the idea of interaction learning for high-dimensional image data in the model. Compared to traditional supervised CNNs, this integrated CNN-DQLN model provides unique advantages:

- Enhanced Generalization: By exploring the DQLN the model learns classification by experimenting without being directly supervised and thus learns on its own. It makes it possible for it to generalize to new images because by looking at the extracted features it learns how it can estimate the feature values for a new image.
- Efficient Learning with Feedback Loops: Incorporation of reward mechanism in DQLN makes the model to be constantly updated from experience and classify appropriately based on total experience. This feedback loop affords robustness against minor perturbations in waste objects and also when the object is different from the training set.
- Improved Handling of Ambiguous Classes: When classes possess similar visual features, they can overlap (for example, cardboard with paper), and the capability of learning to navigate and refine itself within the DQLN can prevent an incorrect categorization of classes.

**4.3 Model implementation**

4.3.1 Initialization

The environment is initialized with the preprocessed data. The DQLN Agent is initialized with the shape of the state (image) and the number of actions (classes).

4.3.2 Experimental design

Through the trajectory, each episode is executed, for each episode, the training loops work as follows: The environment is reset to an initial state using the reset method, randomly selecting an image from the training data. The state is expanded to include a batch dimension and is then passed to the agent. The agent selects the action by analyzing the current state using the act method. The agent takes a step based on the action performed using the step method, which returns the next state, reward, and whether the episode is done. The next state is expanded to include a batch dimension and stored in the agent's memory along with the current state, action, reward, and done flag. The agent replays experiences from memory and updates Q-values using the replay method, as shown in Figure 3.

4.3.3 Environment

The reset method randomly selects an image from the training data as the initial state. The step method accepts an action (class label) and returns whether the episode is finished, the next state, and the reward. In this simplified example, the reward is 1 if the action matches the label of the image and 0 otherwise. The episode is considered done after each step.

4.3.4 Agent

The act method takes an action (class label) depending on the current state. The replay method updates the agent's Q-values based on experiences stored in its memory.

4.3.5 Action space

The number of possible actions the agent can take defines the action space. The action space consists of the dataset's number of classes or labels.

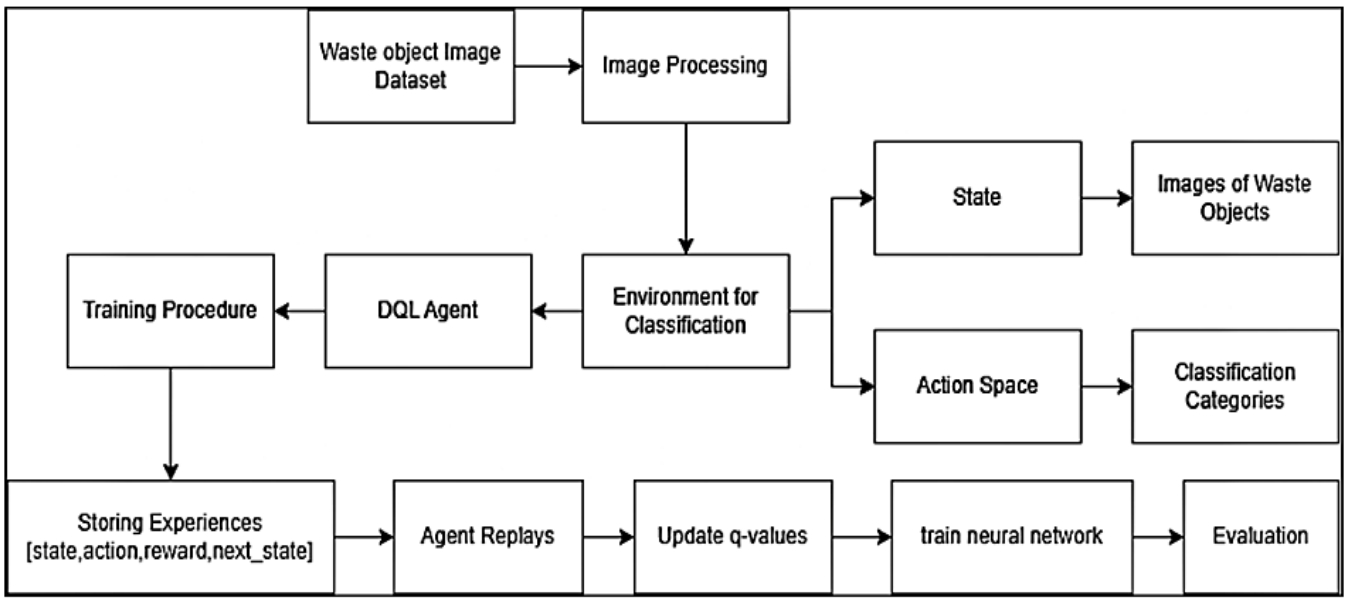


**Figure 3.** Proposed deep Q-learning framework

### 4.4 State representation and transition

The input pre-processed images represent the states. Each image represents a scenario of the environment at a given time. The state is then passed to the agent as input to the neural network model. The environment class provides the relation between the agent and the environment. To initialize the environment to an initial state, the reset method is used, and the step method is used to perform an action and transition to the next state based on the agents' action. When the reset method is called, the environment is initialized to an initial state, and an image from the dataset is randomly selected as the initial state. An action is carried out by the agent in the environment when the step method is called. The input is the action, which is the class label. The reward is determined based on whether the action matches the current state image's label; hence, the episode is considered done after each step.

### 4.5 Training procedure

DQN is a value-based deep reinforcement learning system that uses a convolutional neural network (CNN) front end with a fully connected layer at the end to translate the visual input sequence to the action value functions. The neural network architecture comprises fully linked layers after various convolutional layers. The input layer is the convolutional layer, which has 32 filters with ReLU activation functions and 3×3 kernel sizes. It takes coevolutionary inputs in the shape of 100×100×3 to take the RGB images converted to having a size of 100×100 pixels. The model architecture defined consists of three convolutional layers each having 32, 64 and 128 filters and a 2x2 max pooling layer with a ReLU activation function after each convolutional layer to reduce the spatial dimensions. After feature extraction, the model has two dense layers with 256 neurons and ReLU activation, followed by 128 neurons with ReLU activation function and a dropout of 0.5 to avoid overfitting. The output is flattened into a one-dimensional vector. The dense layer's activation function is ReLU, which contains 64 units. The linear activation function and the number of actions is represented in units in the output layer. The last layer contains 6 neurons, which represent the classes of waste that the model can classify.

Key hyperparameters used are; the learning rate of 0.001 with Adam optimizer controlling how fast learning happens, gamma equal to 0.99 to balance between immediate and future rewards, epsilon for exploration. The epsilon, or exploration rate, is set to initial epsilon of 1.0 and decreases by the factor of 0.995 every episode while the minimum epsilon is set to 0.01 in order to eventually promote exploitation over exploration as learning goes on. Also, fixed-size replay memory of 10,000 experiences is used to store and sample experiences, with the batch size of 32 for each update, to improve learning stability by removing sample correlation. The model underwent training for 1000 episodes in which replay was incorporated so as to randomly draw from the agent's prior experiences so that learning is stabilized.

## 5. RESULTS AND DISCUSSIONS

This section describes the results obtained from the experiments where a reinforcement learning environment is utilized to train the agent for waste object classification. Considering the classification task, the evaluation metric used is accuracy. The accuracy metric below shows what proportion of negative and positive classes are accurately classified and is calculated using the Eq. (3).

$$Accuracy = \frac{T_p + T_N}{P + N} \tag{3}$$

### 5.1 Models performance overview

The model yielded an accuracy of 73.09% on the training set and an accuracy of 72.70% on the test set as shown in Table 1. From these results we can understand that the model does not overfit to the training data and appears to generalize when tested on the validation data with relatively small difference between the training and validation accuracy.

The agent transitioned from exploitation to exploration from episodes out of the first couple of hundred, indicating a subsequent enhancement in performance and hence the observed convergence of the localization accuracy on the validation and test sets. The model yielded the best results on clearly separable classes such as glass and plastic in all probability due to disparities in terms of visual attributes. But when it came to classes such as paper and cardboard which are much more similar in appearance, it had a problem.

**Table 1.** Training and test accuracy

| Metric | Value |
|---|---|
| Training Accuracy | 73.09% |
| Test Accuracy | 72.70% |

### 5.2 Training and convergence behavior

In this section, the training and convergence behavior of the model is explained through an analysis of the learning curve and epsilon decay effect.

Figure 4 plots the training and test accuracies across multiple episodes. It shows how the model's performance changes for the training and testing phase. The learning curves show an initial steeply rising segment for accuracy, which then levels off. This pattern is commonplace in reinforcement learning and other model training frameworks prevalent in artificial intelligence, where early exploration leads to slow incremental gains, which tend to flatten out as the model moves to exploitation.

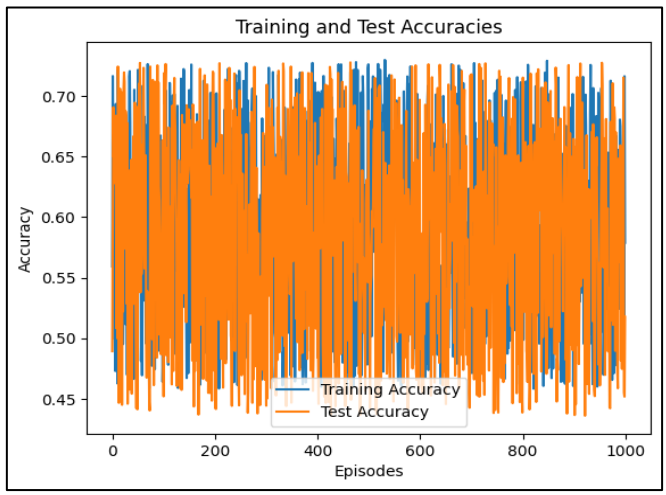


**Figure 4.** Training and testing accuracies through episodes

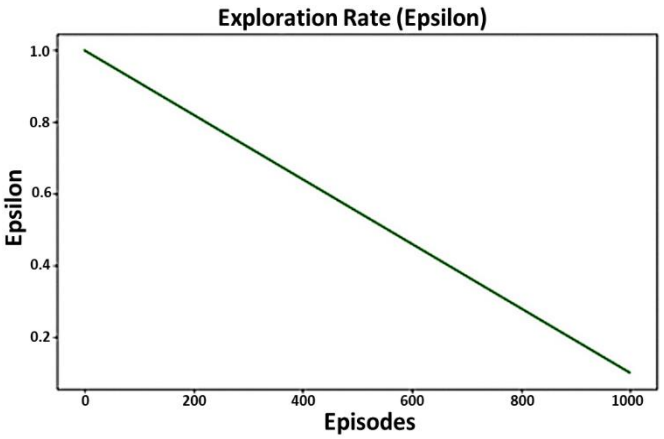


**Figure 5.** Exploration rate decay

The exploration rate epsilon decays throughout the episodes for further exploration and exploitation and the decaying value can be analyzed by the visualization in Figure 5. The epsilon greedy policy with the decay of exploration was by far the most influential factor in the learning process. Exploration at the initial stage (large epsilon) enabled the agent to exert a broad and deep search for action space and gather a larger number of samples in the experience replay memory. The lower epsilon was, the higher was the focus on exploitation, the confidence interval decreased and accuracy of the model increased.

### 5.3 Reward accumulation and agent behavior

Figure 6 shows the total reward achieved by the DQN agent for classifying the waste object images. It gets a reward of +1 for correct classification and -1 for incorrect classification. Throughout the 1000 episodes, the agent's reward value is around 100. In the long run, the agent produces more cumulative rewards per episode, implying that images were classified with higher frequency and accuracy. The fact that total rewards have risen also corresponds with the improvements that have been observed in the accuracy of the model. Action Analysis of the action frequency revealed that with learning, the agent's behavior became selective, and the agent selected the actions that were most in accordance with the classification. This shift underlines the agent's ability to memorize specific characteristics of each class and choose actions correspondingly.

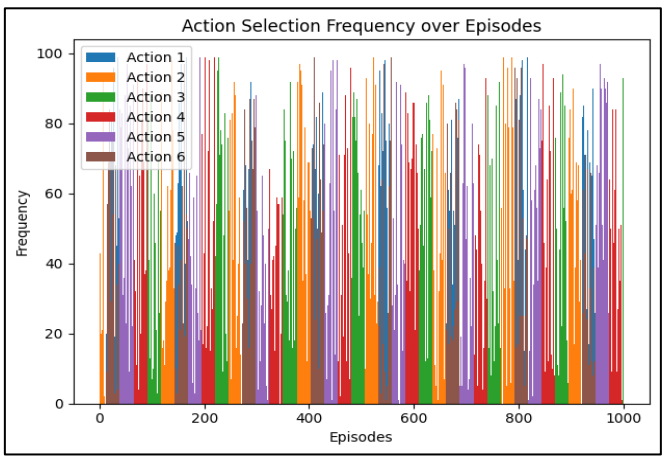


**Figure 6.** Action selection frequency over episodes

### 5.4 Model complexity and overfitting mitigation

The model's higher accuracy of some classes indicates that selective discriminative features relating to specific types of waste were learned. The high accuracy on classes such as metal or glass shows that the identified patterns were unique texture and color for these material types. The lower accuracy in similar classes, including paper and cardboard, is attributed to what the authors described as a tough problem because the two materials are often similar in color and texture. This misclassification tendency could be challenged by including further preprocessing steps such as converting a feature selection step or data augmentation. The use out of dropout layers at the fully connected layers was also effective in preventing overfitting to the training data set. Standard supervised models may entail high regularization, but the reinforcement learning structure of the DQLN neither significantly overfit nor overly generalize during its performance. However, the given model profited from a high overall accuracy; there exists possibility of bringing its accuracy to a higher level by using further enhancements such as refining deep convolutional layer sizes, or enhancing memory base for experience replay.

### 5.5 Practical application to real-world waste segregation systems

The findings of this research can be used as a basis for establishing an automated waste segregation system utilizing DRL. Exploiting the classification model designed in this research, a helpful tool must be designed for sorting waste items in recycling stations, factories, or waste disposal areas. The subsequent interaction with new data is also a strength of the model since real-world conditions may differ and can complicate the classification of waste materials and can be used as:

1). Integration with Physical Systems: To operationalize this model into a practical application, an integration with a robotic system that can sort, collect, and segregate waste according to the DRL model would be required. Incorporating a camera system, images of incoming wastes would be taken and analyzed based on information processed by the model to detect waste categories. Items could then, for instance, be sorted by classification by use of robotic arms or conveyor belts, thus minimizing the need for humans to be involved in the segregation of wastes.

2). Real-Time Processing Requirements: In actual usage, these predictions must be made as the waste is being processed and to cater to large throughput facilities. While our model showcased a high degree of success in a simulated setting, if the program were to be deployed, there would be further improvements that can be made to the code, including the speed of image processing and/or the design of the prediction algorithm. This may encompass techniques to remove unrequired model dimensions, employing different worth processors, generally known as accelerators (e.g., GPU or TPU), or adopting slim architectural designs that offer desirable levels of both precision and velocity.

3). Adapting to Variable Waste Conditions: Waste materials in real-life scenarios may be contaminated with dirt, blurry, or only partially covered, and this presents a formidable task for classification. To overcome this issue, the model could incorporate other approaches to learning, such as incremental learning, whereby the model is updated after some time by

training it with data from new waste items. Moreover, such strategies as data augmentation (for instance, the generation of occlusions or distortions) and feature adaptation, transferring the model from one set of conditions to another, deepened the understanding of robustness.

The potential challenges and solutions for the real-world system could be:

1). Data Diversity and Generalization: One implementational issue associated with this model is that wastes produced vary widely in regions or facilities. However, the real waste streams can contain material that may not fall under these classes and, hence, may be classified wrongly. Possible controls could be to increase the scope of 'learning' so that many more materials are within the training set or add an 'other/unknown' designation so that materials not part of the known classes on the system could be sent to the manual check.

2). Model Drift and Continuous Learning: Waste characteristics change because of changes that are occasionally made to the content or packaging material, and because of these reasons, variations would affect the model. In response to this, it would be possible to implement a continuous learning framework to enable the model to retrain with new data gathered from the deployment environment regularly. _feedback mechanism of reinforcement learning would also help the model to modify its judgments over time and remain informative and accurate.

3). Hardware and Energy Constraints: Training DRL models would pose a challenge to facilities with fewer computational resources and which may require significant resources to deploy the models at scale. Possible solutions are choosing edge computing devices that are compatible with deep learning, like Nvidia Jetson or Google Coral, in order not to involve central servers at the classification stage. The model's applicability in hardware with limited resources would also be achieved by compressing the model by reducing its size or complexity using advanced techniques such as quantization and pruning.

Although the use of DRL applied in segregation of wastes may offer promising solutions in accuracy and efficiency in operations in sorting facilities, it has great opportunity to offer its rated contribution to sustainability by increasing the number of recycles and minimizing contamination in wastes. Possible extensions of this work could consider applications for a higher classification by recyclability grades or for categorizing waste in terms of environmental harm, to extent efforts to integrate waste management.

### 5.6 Limitations and challenges

It is critical to acknowledge certain limitations although the present work offers important contributions to understanding the applicability of deep reinforcement learning to garbage sorting. The really essential component is that we rely on the dataset for the testing and training of DQN models. The current dataset used lacks the number of classes which shows the total number of variations possible in waste collection in the real world. For instance, the differences between paper and cardboard images can be somewhat challenging for the model to make, meaning the classification is not going to be completely accurate all of the time. For future work, a more diverse and wide range of waste images should be incorporated, along with more waste classes and image variations. There may be an issue of longer training time the application of reinforcement learning because of the requirement of successive interaction as well as exploration. It can be a little time-consuming at times due to the heavy computation involved.

## 6. CONCLUSION

The research shows how deep reinforcement learning can be used for detection and classification of waste objects into different categories. The system can self-process and identify correct waste objects by simulating the waste sorting process as sequencing of decision outcomes. This would lead to higher sorting rates and lower errors. The study underlines the capacity of deep reinforcement learning to spark the waste revolution recognized with intelligence and sustainability in the face of environmental challenges. The Deep Q Network (DQN) model's effectiveness, with an accuracy of around 73%, demonstrates the model's ability to manage the complexity of various waste types. Using the approach that involves extended experimentation and assessment as a base, there are successes achieved: high relative waste sorting accuracy and efficiency. It presents an optimistic outlook for a future that is more sustainable. There can be an increase in public knowledge of waste management and responsible consumption by promoting the use of clever garbage sorting systems. By delving further into the application of deep reinforcement learning in this field, meaningful progress towards a circular economy can be achieved, where waste is not seen as a problem, but as a valuable asset ready to be repurposed. Collecting and categorizing this data may be a challenging job. Furthermore, waste compositions can differ depending on location and can also change daily, which requires our model to be flexible in order to manage these variations without requiring significant retraining. Future studies could explore some more complex environments with multiple linked episodes and by applying data augmentation to increase robustness by increased training of model on various images.

In the future, this field could involve research into improving the above-mentioned deep reinforcement learning model and its applicability to waste commodity detection and classifying. Experimenting with this kind of depth functions as transfer learning and meta-learning could help achieve higher performance when handling heterogeneous waste substances and environmental circumstances. Additionally, applying advanced techniques of reinforcement learning like double DQN or dueling DQN may enhance the model's performance and accuracy. Moreover, implementing sensor systems that can access real-time data and artificial intelligent systems can help the system adjust to dynamic waste compositions and resolve sorting problems.

## REFERENCES


[1] Duhayyim, M.A., Elfadil Eisa, T.A., Al-Wesabi, F.N., Abdelmaboud, A., Hamza, M.A., Zamani, A.S., Rizwanullah, M., Marzouk, R. (2022). Deep reinforcement learning enabled smart city recycling waste object classification. Computers, Materials & Continua, 71(3): 5699-5715. https://doi.org/10.32604/cmc.2022.024431

[2] Fang, W., Pang, L., Yi, W.N. (2020). Survey on the

application of deep reinforcement learning in image processing. Journal on Artificial Intelligence, 2(1): 39-58. https://doi.org/10.32604/jai.2020.09789

[3] Liang, H.Q. (2020). A precision advertising strategy based on deep reinforcement learning. Ingénierie des Systèmes d'Information, 25(3): 397-403. https://doi.org/10.18280/isi.250316

[4] Wang, J., Yan, Y., Zhang, Y., Cao, G., Yang, M., Ng, M.K. (2020). Deep reinforcement active learning for medical image classification. In Medical Image Computing and Computer Assisted Intervention-MICCAI 2020: 23rd International Conference, Lima, Peru, pp. 33-42. https://doi.org/10.1007/978-3-030-59710-8_4

[5] Rajani, K.R., Gaddam, A., Gaddam, J. (2022). IoT based smart waste management using deep reinforcement learning. Information Systems. https://www.preprints.org/manuscript/202211.0190.

[6] Weerasekara, S., Li, W., Isaacs, J., Kamarthi, S. (2024). Reinforcement learning for disassembly task control. Computers & Industrial Engineering, 190: 110044. https://doi.org/10.1016/j.cie.2024.110044

[7] Rakesh, U., Ramya, V., Murugan, V.S. (2023). Classification, collection, and notification of medical waste using IoT based smart dust bins. Ingénierie des Systèmes d'Information, 28(1): 149-154. https://doi.org/10.18280/isi.280115

[8] Okafor, E., Oyedeji, M., Alfarraj, M. (2024). Deep reinforcement learning with light-weight vision model for sequential robotic object sorting. Journal of King Saud University-Computer and Information Sciences, 36(1): 101896. https://doi.org/10.1016/j.jksuci.2023.101896

[9] Zeng, A., Song, S., Yu, K.T., Donlon, E., Hogan, F.R., Bauza, M., Ma, D., Taylor, O., Liu, M., Romo, E., Fazeli, N., Alet, F., Dafle, N.C., Holladay, R., Morona, I., Nair, P.Q., Green, D., Taylor, I., Liu, W., Funkhouser, T., Rodriguez, A. (2022). Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching. The International Journal of Robotics Research, 41(7): 690-705. https://doi.org/10.1177/0278364919868017

[10] Vijayan, R.V., Krishnan, M.M., Parayitam, S., Duraisami, S.P.A., Saravanaselvan, N.R. (2023). Exploring e-Waste recycling behaviour intention among the households: Evidence from India. Cleaner Materials, 7: 100174. https://doi.org/10.1016/j.clema.2023.100174

[11] Munir, M.T., Li, B., Naqvi, M. (2023). Revolutionizing municipal solid waste management (MSWM) with machine learning as a clean resource: Opportunities, challenges and solutions. Fuel, 348: 128548. https://doi.org/10.1016/j.fuel.2023.128548

[12] Uzkent, B., Yeh, C., Ermon, S. (2020). Efficient object detection in large images using deep reinforcement learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. Snowmass, CO, USA, pp. 1824-1833. https://doi.org/10.1109/WACV45572.2020.9093447

[13] Ye, J., Wu, Y., Peng, D. (2024). Low-Quality image object detection based on reinforcement learning adaptive enhancement. Pattern Recognition Letters, 182: 67-75. https://doi.org/10.1016/j.patrec.2024.04.019

[14] Nicholaus, I.T., Kang, D.K. (2022). Robust experience replay sampling for multi-agent reinforcement learning. Pattern Recognition Letters, 155: 135-142. https://doi.org/10.1016/j.patrec.2021.11.006

[15] Colas, C., Sigaud, O., Oudeyer, P.Y. (2018). Gep-pg: Decoupling exploration and exploitation in deep reinforcement learning algorithms. In International Conference on Machine Learning, PMLR, pp. 1039-1048. https://doi.org/10.48550/arXiv.1802.05054

[16] Rodrigues Gomes, E., Kowalczyk, R. (2009). Dynamic analysis of multiagent Q-learning with ε-greedy exploration. In Proceedings of the 26th Annual International Conference on Machine Learning, Canada, pp. 369-376. https://doi.org/10.1145/1553374.1553422

[17] Zhao, J., Wei, S., Liao, L., Zhao, Y. (2021). DQN-Based gradual fisheye image rectification. Pattern Recognition Letters, 152: 129-134. https://doi.org/10.1016/j.patrec.2021.08.025

[18] Golazad, S., Mohammadi, A., Rashidi, A., Ilbeigi, M. (2024). From raw to refined: Data preprocessing for construction machine learning (ML), deep learning (DL), and reinforcement learning (RL) models. Automation in Construction, 168: 105844. https://doi.org/10.1016/j.autcon.2024.105844

[19] Zhou, S.K., Le, H.N., Luu, K., Nguyen, H.V., Ayache, N. (2021). Deep reinforcement learning in medical imaging: A literature review. Medical Image Analysis, 73: 102193. https://doi.org/10.1016/j.media.2021.102193

[20] Sharma, S., Guleria, K. (2022). Deep learning models for image classification: Comparison and applications. In 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, pp. 1733-1738. https://doi.org/10.1109/ICACITE53722.2022.9823516

[21] Aydin, İ., Sevi, M., Güngören, G., İrez, H.C. (2022). Signal synchronization of traffic lights using reinforcement learning. In 2022 International Conference on Data Analytics for Business and Industry (ICDABI), Sakhir, Bahrain, pp. 103-108. https://doi.org/10.1109/ICDABI56818.2022.10041559

[22] Mao, W.L., Chen, W.C., Wang, C.T., Lin, Y.H. (2021). Recycling waste classification using optimized convolutional neural network. Resources, Conservation and Recycling, 164: 105132. https://doi.org/10.1016/j.resconrec.2020.105132

[23] Talaat, F.M. (2022). Effective deep Q-networks (EDQN) strategy for resource allocation based on optimized reinforcement learning algorithm. Multimedia Tools and Applications, 81(28): 39945-39961. https://doi.org/10.1007/s11042-022-13000-0

[24] Li, S.E. (2023). Reinforcement Learning for Sequential Decision and Optimal Control. Berlin/Heidelberg, Germany: Springer, pp. 1-449. https://doi.org/10.1007/978-981-19-7784-8

[25] Fan, J., Wang, Z., Xie, Y., Yang, Z. (2020). A theoretical analysis of deep Q-learning. In Learning for Dynamics and Control, PMLR, pp. 486-489. https://doi.org/10.48550/arXiv.1901.00137

[26] Chen, S.A., Tangkaratt, V., Lin, H.T., Sugiyama, M. (2020). Active deep Q-learning with demonstration. Machine Learning, 109(9): 1699-1725. https://doi.org/10.1007/s10994-019-05849-4

[27] Yang, Y., Hao, J., Chen, G., Tang, H., Chen, Y., Hu, Y., Fan, C., Wei, Z. (2020). Q-Value path decomposition for deep multiagent reinforcement learning. In International Conference on Machine Learning, PMLR, pp. 10706-

10715. https://doi.org/10.48550/arXiv.2002.03950
[28] Iosifidis, A., Tefas, A. (Eds.). (2022). Deep learning for robot perception and cognition. Academic Press. https://doi.org/10.1016/C2020-0-02902-6
[29] Hayes, T.L., Krishnan, G.P., Bazhenov, M., Siegelmann, H.T., Sejnowski, T.J., Kanan, C. (2021). Replay in deep learning: Current approaches and missing biological elements. Neural Computation, 33(11): 2908-2950. https://doi.org/10.1162/neco_a_01433
[30] Yuan, W., Li, Y., Zhuang, H., Wang, C., Yang, M. (2021). Prioritized experience replay-based deep q learning: Multiple-reward architecture for highway driving decision making. IEEE Robotics & Automation Magazine, 28(4): 21-31. https://doi.org/10.1109/MRA.2021.3115980
[31] CCHANG. (2018). Garbage Classification, Kaggle. https://doi.org/10.34740/KAGGLE/DS/81794.
[32] Nnamoko, N., Barrowclough, J., Procter, J. (2022). Solid waste image classification using deep convolutional neural network. Infrastructures, 7(4): 47. https://doi.org/10.3390/infrastructures7040047