Continual Learning for Embodied Agents: Methods, Evaluation and Practical Use

a Systematic Literature Review

Andrei Dascalu[1]

Supervisor(s): Chirag Raman[1]**, Ojas Shirekar**[1]

[1]**EEMCS, Delft University of Technology, The Netherlands**

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 23, 2024

Name of the student: Andrei Dascalu
Final project course: CSE3000 Research Project
Thesis committee: Chirag Raman, Ojas Shirekar, Willem-Paul Brinkman

Abstract

Continual learning (CL) enables intelligent systems to continually acquire, adapt, and apply knowledge, representing a dynamic paradigm in AI. For embodied agents—interacting with their environment physically and cognitively—CL enhances adaptability and reduces training costs significantly. In this literature review, we contribute by focusing on the application of CL in such agents, showcasing the approaches, means of evaluation and practical uses of this cognitive framework in real-world scenarios. We conclude that while CL holds promise for embodied agents, there exists a notable gap between the theoretical evaluation of CL and the complex real-world scenarios these agents operate in.

1 Introduction

In the field of artificial intelligence, continual and lifelong learning is a paradigm that refers to an agent's capacity to learn continuously, accumulate the knowledge learned in the past, and use or adapt it to help future learning and problem solving [1]. Unlike traditional static methods of training, continual learning (CL) is facilitated by incremental training over an infinite stream of data [2]. This approach avoids periodic full retraining to accommodate new tasks or fresh data, potentially reducing computational and energy demands [3].

The CL paradigm is rooted in the principles of cognitive psychology, mirroring the human ability to sequentially learn new concepts [4]. Recent studies suggest that continual learning in humans is facilitated by a mix of synaptic plasticity and consolidation [5]. Synaptic or structural plasticity is a fundamental mechanism in the brain that involves the strengthening or weakening of synapses, which allows the brain to learn and adapt to new information [5], [6]. Consolidation, on the other hand, represents the brain's attempt to stabilize a memory over time, which helps to establish long-term memory by maintaining synaptic stability [5], [7].

In the context of continual learning, embodied agents represent entities that interact with and perceive their environment in a human-like manner, utilizing both physical and cognitive capabilities. Embodied agents, which can be either physical robots or virtual agents, integrate sensory input to navigate and perform tasks within their environments. Virtual agents, like their physical counterparts, engage with their environments through simulated sensory and motor processes.

Unlike humans, agents employing continual learning are subject to forgetting how to solve past tasks entirely after learning new tasks. This phenomenon is known as *catastrophic forgetting* (CF) [8]. This issue stems from the delicate balance between learning plasticity and memory stability [9], akin to the principles of synaptic plasticity and consolidation that form the basis of human memory. Excessive learning plasticity can result in rapid adaptation but may lead to a loss of information about past tasks. Conversely, excessive memory stability may help the agent retain past knowledge, at the cost of making the agent rigid and less adaptable to new tasks, as it becomes resistant to changing its learned parameters.

There have been successful attempts to tackle the issue of CF through multiple defined methods [10], [11], [12]. While previous studies provide a solid theoretical overview of such frameworks and methods ([4], [9]), the challenge lies in effectively incorporating these findings into practical applications. Hence, it is crucial to explore the extent to which these learning methods have been applied in practice.

Motivation The motivation for this research stems from the need to overcome the limitations of current AI systems, which, despite their impressive results, rely on static models that require restarting the training process whenever new data emerges [4]. In an ideal scenario, we should aim for models that are capable of dynamically learning, motivated by internal factors rather than solely by external rewards. Furthermore, deriving insights from human intelligence is not only essential for advancing AI from both conceptual and practical perspectives, but it also opens the possibility of creating agents that can plan, adapt, and act effectively in various environments, not just recognize patterns. This approach bridges the gap between AI and cognitive science, deepening our understanding of how human intelligence works and paving the way for more adaptable and intelligent agents.

Contribution The aim of this systematic review is to provide an overview of the advancements made in the topic of CL, with a focus on the practical applications in embodied agents. Through this review, we will not only present the frameworks commonly adopted in CL, but also pave the way to future advancements by giving concrete examples of what has been achieved so far in practice, and whether practical usages of CL methods match their theoretical expectations.

The main research question that this study focuses on is "*How has continual and lifelong learning been incorporated into embodied agents, mirroring the human capacity to incrementally acquire new knowledge?*". In addition to the main question, three sub-questions were devised to support the exploration and understanding of the research topic:

- **Q1** What methods and algorithms facilitate continual and lifelong learning in embodied agents, and what are their advantages, drawbacks and cognitive inspiration?
- **Q2** How is the performance of systems that are capable of lifelong learning evaluated?
- **Q3** How has continual and lifelong learning been integrated into embodied agents in practice?

2 Method

A systematic literature review was conducted in accordance to the PRISMA[1] reporting guideline. PRISMA is an evidence-based set of 27 items designed to help authors transparently report why the systematic review was conducted, what methods were used, and what has been found. This reporting guideline was employed in an effort to ensure the reproducibility and transparency in this paper. The subsequent sections will be structured as follows: Section 2.1 gives an overview of how the literature search was carried out and which databases were selected. Section 2.2 showcases the selection criteria based on which the filtering was performed. Section 2.3 describes the literature screening process and the final selection of papers.

[1] PRISMA: Preferred Reporting Items for Systematic Reviews and Meta-Analyses. Available at: http://www.prisma-statement.org/

2.1 Selection Criteria

In order to assess the eligibility of the identified articles from the initial literature search, the following exclusion and inclusion criteria were established:

Inclusion Criteria:

- Study published after 2015
- Journal article/conference proceeding written in English
- Study focuses on the integration of CL in embodied agents
- Study contains details on method used for achieving CL

Exclusion Criteria:

- Full-text study is not available
- Article not written in English
- Study not directly relevant to continual and lifelong learning in the context of embodied agents

2.2 Literature search

To conduct the literature search, three databases were taken into consideration: IEEE Xplore [2], Scopus [3] and Web of Science [4]. IEEE Xplore is a research database that focuses on computer science and technology related articles and proceedings, and was selected for its potential to contain papers relevant to the topics of interest. Scopus, a comprehensive multidisciplinary database, was selected to ensure a broad spectrum of findings, not confined solely to computer science, but also incorporating insights from cognitive science. Web of Science, another multidisciplinary database, was selected to complement the search, providing coverage for papers that may not have been included otherwise.

A search string structure has been devised in order to query the three databases. The string is composed of multiple phrases related to the main research question, each phrase being devised into multiple related terms. These phrases were concatenated using the AND operator, with the related terms being concatenated with the OR operator inside the phrase. The search string that was used across the three databases can be found in **Figure 1**.

("life-long learning" OR "lifelong learning" OR "continual learning" OR "incremental learning" OR "sequential learning") AND (environment OR "3D environment*" OR "virtual environment*") AND (agent* OR multi-agent OR "multi agent" OR "intelligent agent*" OR "autonomous agent*" OR "embodied agent*" OR robot) AND NOT (education)

Figure 1: Search string used in database search

The search string was adjusted to use the "NOT" operator instead of the "AND NOT" operator for Web of Science due to syntax differences. The search was limited to papers and articles published after and including 2015. Additionally, Scopus results were restricted to the field of 'Computer Science', and the search was carried out within "Abstract, Title, and Keywords". The full search was performed on 6/5/2024. Results were recorded in Zotero, and duplicates were removed.

Additionally, Google Scholar was used to acquire literature with a focus on cognitive sciences. This decision was made due to the emphasis of the search query on the computational aspects of continual learning, rather than on the cognitive background of the framework.

2.3 Literature selection

Literature selection was performed according to the PRISMA guideline, and encompassed three stages. The first stage is *identification*, where relevant articles and papers were recorded using the query across the aforementioned databases, and the results were filtered for duplicates. The second step is *screening*, where the remaining articles were first screened based on the selection criteria. The resulting articles were sought for retrieval, and then assessed for eligibility based on the full-text content, namely the abstract, introduction and conclusion. The *included* step represents the final number of papers included in the literature review, accounting for the 24 papers resulting from the database search, along with 25 records identified through citation chaining, and 5 papers using Google Scholar. **Figure 2** illustrates the process, along with the number of records included in each step.

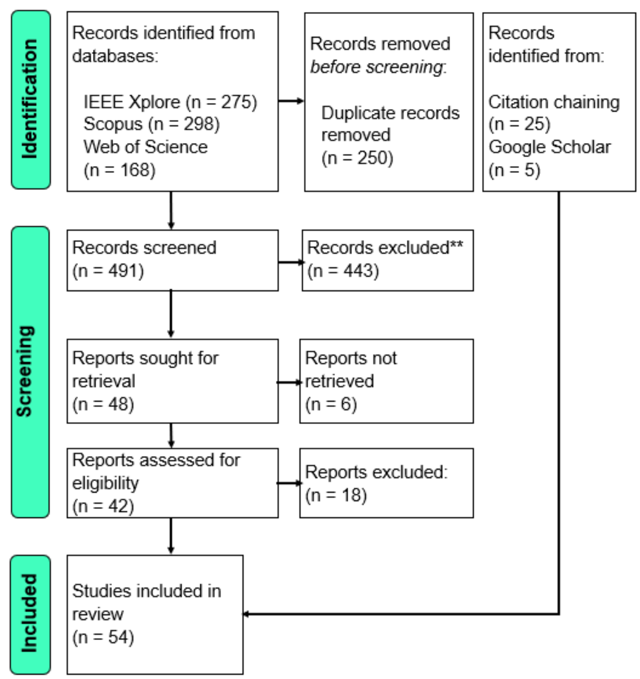

Figure 2: PRISMA flow diagram

3 Results

This chapter will discuss the results for each of the proposed research sub-questions addressed in Section 1. Section 3.1 will explain the different approaches to achieving continual learning, along with advantages, disadvantages and characteristics. Section 3.2 will elaborate on the current methods of performance evaluation of CL systems. Finally, Section 3.3 will offer examples of how CL is employed in practice.

[2] Available at: https://ieeexplore.ieee.org

[3] Available at: https://www.scopus.com

[4] Available at: https://www.webofscience.com/wos/

3.1 Q1 - Methods and approaches for CL

Van de Ven et al. [13] categorize continual learning into three main scenarios based on how training data batches are divided and task identities are handled during testing. In **Task-IL**, models are aware of task identities during testing, making it the least challenging scenario. **Domain-IL** does not provide task identity during testing but focuses on solving each task with varying input distributions, resembling adaptive agent scenarios. **Class-IL** requires models to both solve tasks and infer task identities, reflecting challenges in learning new classes incrementally with limited information.

In navigating these scenarios, an important aspect of continual learning is the ability to leverage knowledge across tasks. **Forward transfer** is the "ability to transfer knowledge from past tasks to improve the learning and efficiency of future (related) tasks", as defined by Wickramasinghe [14, p. 4]. Conversely, **backward transfer** refers to the "ability to transfer knowledge from future tasks to past (related) tasks to enhance their performance" [14, p. 4]. Effective continual learning methods aim to maximize positive forward transfer while ensuring that positive backward transfer is also achieved.

Traditionally, continual learning approaches have been split into three main categories [4], [15]. These categories are: **(i) Replay methods**, **(ii) Regularization methods**, and **(iii) Parameter isolation methods**. Whilst other studies such as that of Wang et al. [9] introduce the notion of **optimization-based** and **representation-based approaches**, these two categories will not be considered for this literature survey. The reason for excluding them lies in their relatively less common usage in the broader literature on CL.

The commonality between the three mentioned approaches lies in their shared goal of mitigating CF. However, there are still fundamental differences, with each approach offering unique advantages/disadvantages in varying scenarios. These aspects, along with the cognitive inspiration from each which will be described in the following sections. Note that we will only detail upon on the most fundamental and significant works within the three primary approaches. A summary of the findings can be found in **Table 1**.

(i) Replay-based approach

Among the surveyed studies, twelve papers discuss a method that resembles replay-based approaches [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [15], [26]. This type of approach involves the storage and consequent replay of past data samples during training to prevent forgetting. The core idea involves retaining a representative subset of prior tasks within a compact memory buffer, which is subsequently used during training alongside new tasks. The main scenario for the application of this approach is task-incremental learning (Task-IL) [15], although certain proposals were also made to target class-incremental learning (Class-IL) [16], [21]. Replay-based approaches can be sub-categorized into *rehearsal* and *generative* methods.

Rehearsal methods such as iCaRL (Incremental Classifier and Representation Learning) [16] construct an augmented training set with stored exemplars and current examples. During training, the network outputs for previous classes are stored. The loss function combines classification loss for new classes and distillation loss, minimizing forgetting by reproducing previous class scores while adapting to new data. Another method, GEM [17], uses an episodic memory to store a subset of examples from each task. It enforces constraints that prevent increasing the loss on past tasks by projecting gradient updates to ensure positive backward transfer upon learning new tasks. A-GEM [18] enhances GEM by ensuring the average loss over all past tasks does not increase, using a single constraint, making it more computationally efficient.

Generative methods, also known as pseudo-rehearsal, focus on using generative models to produce representative data samples to be used during rehearsal. The DGR framework [19] involves training a "scholar", which is composed of two components. The first component is a generator, which is a deep generative model that recreates realistic samples from past tasks. The second component is a solver, which is a task-solving model trained on a mix of current task data and replayed data from the generator. Other proposals such as FearNet [20] go a step further in taking inspiration from cognitive science. FearNet is comprised of three brain-inspired modules: recent memory, long-term storage, and a decision-making subsystem. FearNet's architecture includes neural networks inspired by hippocampal complex (HC) and medial prefrontal cortex (mPFC), with a basolateral amygdala (BLA) subsystem determining which memory to use. It addresses CF by consolidating recent memories into long-term storage via pseudorehearsal, enabling the network to revisit past memories without storing previous training examples.

The benefits of replay-based approaches revolve around their capacity to mitigate CF. According to Bagus et al., replay-based methods outperform other methods under various constraints, with small and fixed memory and computation overhead [27]. They scale well across many tasks and are universal, meaning they can be applied in almost any scenario without changes. Additionally, replay-based methods are simple to implement and explain, using old sub-task samples during new sub-tasks to prevent forgetting, similar to joint training over all seen sub-tasks.

However, drawbacks include a memory overhead due to the need for storing past data samples. While generative replay addresses this issue, they may introduce complexity [4] and the risk of generating unrealistic samples. Storing samples directly, as is the case with rehearsal methods, rather than using generative models, can lead to privacy concerns [27]. Moreover, replay approaches are prone to overfitting. This can come as a result of either insufficient diversity per class when sampling from memory [28], or from using unrealistic examples that do not capture the true data distribution [27].

The cognitive inspiration behind replay methods stems from the concept of *hippocampal learning*. Specifically, the human brain uses mechanisms of replay during sleep to reinforce and consolidate memories. The hippocampal system records experiences as they occur throughout the day and subsequently replays these stored memories back to the neocortex overnight [29]. According to Rasch et al., during sleep, particularly during the slow-wave sleep phase, the hippocampus reactivates recent experiences, which are then replayed to the neocortex [30]. This replay process is believed to facilitate the transfer of memories from short-term to long-term storage, thereby helping to prevent forgetting and integrate new learning with existing knowledge. Kemker and Kanan clearly draw inspiration from the hippocampal learning process in FearNet [20], with consolidation taking place during "sleep phases". During these sleep phases, generated representative samples are replayed in conjunction with recent data to reinforce previous learning and facilitate the integration of new information into the existing knowledge base.

(ii) Regularization approach

Six of the surveyed papers include a method that resembles regularization approaches [12], [31], [11], [32], [33], [34]. Regularization-based approaches impose constraints on network parameters to balance the retention of old knowledge while acquiring new information. Regularization strategies work by introducing additional terms to the loss function, which guide the optimization process to preserve important aspects of previously learned tasks. The main scenario for this approach is Task-IL [12]. Other methods focus on the Class-IL scenario [32]. Regularization approaches can be sub-categorized into *weight-regularization* and *knowledge-distillation* methods.

Weight-regularization methods operate by minimizing the amount of changes/drift of relevant weights related to past tasks. One such example is Elastic Weight Consolidation (EWC) [12]. In this framework, Kirkpatrick et al. address the challenge of retaining old knowledge by selectively constraining important weights from undergoing significant changes. Here, the Fisher Information Matrix is used to assess the sensitivity of each parameter to changes in data likelihood. Parameters with higher Fisher information values are more important for previously learned tasks. EWC introduces a quadratic penalty term to the loss function, constraining parameters from deviating significantly from their values on previous tasks. R-EWC (Rotated EWC) [31] improves EWC by rotating the parameter space to align with the principal axes of the Fisher Information Matrix, making the diagonal approximation more accurate. This reduces the loss of important correlations between parameters. Similarly to EWC, Aljundi et al. introduce MAS (Memory Aware Synapses) [11], calculating the importance of each network parameter based on how much it affects the output. It then uses this importance to add a regularization term to the loss function, penalizing changes to crucial parameters during task learning.

Knowledge-distillation methods work by transferring knowledge from a "teacher" model to a "student" model. This is achieved by training the student model to mimic the outputs (often the soft probabilities) of the teacher model. In Learning without Forgetting (LwF) [32], the Knowledge Distillation Loss is used to align the outputs of the current model with the recorded outputs of the old tasks. This modified cross-entropy loss ensures that the new model's predictions on old tasks remain close to the original network's predictions.

The benefits of regularization methods lie in their space efficiency. As seen with EWC, the method does not need to store past data samples and does not expand the network. Moreover, these methods ensure positive forward and backward transfer, and often outperform other methods [14].

The drawbacks of regularization methods stem from the balance they must maintain between retaining old knowledge and learning new information. High regularization strength can make the network rigid, hindering adaptation to new tasks. Conversely, low regularization strength may cause the network to undergo too much change, leading to forgetting of previously learned tasks. Wickramasinghe et al. point out that regularization methods rely on knowing the task identity at inference time, which limits their applicability in scenarios where this information is unavailable or uncertain [14].

The cognitive inspiration for this method is *synaptic plasticity*. This process refers to the ability of synapses, the connections between neurons, to change their strength in response to activity [35]. Synaptic plasticity can be divided into two main forms, namely long-term potentiation (LTP), which strengthens synapses through high-frequency stimulation, and long-term depression (LTD), which weakens them via low-frequency stimulation, decreasing synaptic efficiency. This concept is mirrored in CL frameworks such as EWC [12], where critical synaptic weights are conserved, akin to how LTP stabilizes important synapses. This prevents drastic changes to essential weights, maintaining learned knowledge while accommodating new information, similar to the balance achieved by LTP and LTD.

(iii) Parameter isolation approach

Among the surveyed studies, eight papers discuss a method that resembles parameter isolation approaches [36], [37], [38], [39], [40], [41], [42], [10]. Parameter isolation approaches function by assigning distinct sets of parameters to different tasks, ensuring no overlap in parameter usage across tasks. The main scenario for this type of approach is Task-IL [42]. These approaches can either function by adding parameters dynamically as new tasks are encountered (growing the network), or by restricting the network architecture and utilizing subsets of available parameters for different tasks. Thus, we make distinction between two sub-categories, namely *dynamic-architecture* and *fixed-architecture* methods.

Dynamic-architecture methods work by incrementally adding new parameters to the neural network as new tasks are introduced. This ensures sufficient capacity to learn and retain new information. Wang et al. [37] describe a framework where new layers or units are added to the existing network structure, enhancing its depth or width and thereby improving its ability to transfer knowledge and perform well on novel tasks with limited data. Yoon et al. propose the DEN framework [38], which involves selectively retraining relevant neurons to adapt to new tasks, expanding the network by adding new neurons when the existing capacity is insufficient, and duplicating neurons to prevent semantic drift. It utilizes group sparsity regularization to decide the number of new neurons required and timestamped inference to manage parameters across different learning stages. Another dynamic-architecture framework is "learn-to-grow" [39]. It builds upon a base model, iteratively adding task-specific parameters while allowing reuse of prior knowledge. The framework incorporates elastic penalties for parameter reuse. It's implementation includes two components: neural structure optimization using differentiable architecture search, and parameter learning/fine-tuning. The optimization process selects between reusing, adapting, or creating new layers which prevents exponential growth in model size. Progressive networks [40] extend the dynamic architecture approach, by adding a new neural network column for each task. When a new task is introduced, a new column is added with randomly initialized parameters. Each layer in the new column receives inputs from both its previous layer and the corresponding layers of all previously trained columns via lateral connections.

Fixed-architecture methods do not modify the actual structure of the network, but instead use a predetermined network architecture and focus on optimizing the parameters within this fixed structure. Fernando et al. propose PathNet, a framework that facilitates continual learning by evolving modular pathways within a deep neural network [41]. It consists of layers, each containing modules (neural networks) whose outputs are summed before passing to the next layer. During training, pathways are randomly initialized and evolved using tournament selection or parallel evaluation. Once a task is learned, optimal pathways are fixed while other parameters are reinitialized for subsequent tasks. Similar to PathNet, PackNet [42] freezes older pathways while reusing older neurons for new tasks. It uses network pruning to create free parameters for new tasks without increasing network capacity. A standard network is trained for a primary task and then pruned. New tasks are added sequentially, using existing and new parameters, with iterative pruning to remove a fixed percentage of weights from each layer. In contrast, HAT (Hard Attention to the Task) [10] prevents forgetting by using task-specific attention mechanisms to conditionally activate neurons, preserving pathways crucial for each task.

The benefits of parameter-isolation approaches include minimizing interference between tasks, thereby preserving previously learned knowledge and preventing catastrophic forgetting. Moreover, methods that involve dynamic architectures grow the network in response to new tasks, and allow for increased model capacity over time without the need for re-training or freezing of parameters. This approach guarantees maximal stability by fixing parameter subsets of previous tasks, thus preventing stability decay [4].

The drawbacks include of this approach are related to scalability. In the case of dynamic architectures, the connection of new models to all previous ones may result in quadratic parameter growth concerning the number of tasks, posing scalability concerns [43]. Additionally, dynamic architectures pose an expansion feasibility issue. If given enough computational resources, dynamic architectures could in theory fully mitigate the issue of catastrophic forgetting [14]. However, such assumptions are highly unrealistic under practical scenarios. Whilst fixed-architecture methods tackle the growth issue, the explicit allocation of network capacity per task represents a limitation in the total number of possible tasks [4].

The cognitive inspiration for parameter-isolation methods stems from *selective learning* and *brain modularity*. Studies on selective attention suggest that the brain prioritizes processing of salient stimuli while suppressing irrelevant information [44]. This concept serves as a potential inspiration for parameter isolation, which prioritizes the preservation of parameters crucial for previously learned tasks while minimizing future interference. Similar to how parameter isolation methods assign subsets of parameters per task during learning, the human biological brain also utilizes modular architectures to manage cognitive functions. Such modular organization is present early in brain development, which persists and evolves through adulthood [45]. As Kelkar and Medaglia note, "cognitive competences such as choosing one's food habits, spatial navigation, seeing, and face recognition are supported by the modules" [45, p. 7]. This closely relates to how different tasks use distinct subsets of network parameters to prioritize task-relevant information in such approaches.

3.2 Q2 - Performance evaluation of CL

The evaluation of CL methods primarily focuses on assessing how well these methods can learn new tasks sequentially without forgetting previously learned tasks. Studies have explored evaluation strategies and datasets to measure the effectiveness of CL approaches, with thirty-three papers discussing the proposed model's evaluation. **Table 2** contains an overview of the methods, task settings and evaluation metrics that these studies employed.

(i) Task Settings in Continual Learning Evaluation

Findings reveal that continual learning evaluations are conducted under several task settings, which dictate the approach to training and benchmarking. For instance, *Incremental learning* settings involve models being trained on new tasks or classes sequentially while attempting to retain previously learned knowledge. This involves adapting to new data distributions without forgetting previous tasks. For example, Kim et al. [46] employ rotated MNIST, where digits in the MNIST dataset are rotated by certain angles. Similarly, iCarl [16] is evaluated on iCIFAR-100 and iILSVRC benchmarks, demonstrating the model's ability to incrementally learn new classes while retaining old ones.

Supervised learning involves training models on labeled datasets, typically in a batch fashion, without considering sequential learning. However, in the context of continual learning, supervised learning can be used as a baseline for comparing against incremental learning methods. For instance, CBCL-PR [21] uses CIFAR-100 in a supervised setting to evaluate class-incremental learning performance. Another example is FearNet [20], which compares several supervised methods on CIFAR-100 and CUB-200 datasets.

Reinforcement learning focuses on training agents to interact with an environment to achieve a goal, maximizing cumulative rewards. In continual learning, RL tasks typically involve learning sequentially over multiple episodes or tasks while retaining knowledge from previous experiences. Rusu et al. [40] evaluated CL in RL tasks such as synthetic Pong variants and Atari games, demonstrating the ability of progressive networks to adapt to new games while retaining performance on previously learned games. Similarly, EWC [12] was tested on Atari 2600 games, showing how it helps in retaining learned policies.

Online adaptation entails the updating of agent/system behavior in response to new data or changes in the environment. This scenario is particularly relevant in real-world robotics, where robots must adapt to dynamic contexts. For example, Tannenberg et al. [22] discuss the adaptation of

Table 1: Comparative summary of continual learning approaches: benefits, drawbacks, and cognitive inspirations

	Sub-category	Benefits	Drawbacks	Reference	Cognitive Inspiration
Replay	Rehearsal	• Mitigation of CF • Simple implementation • Universal applicability	• Memory overhead • Privacy concerns • Prone to overfitting	[15], [16], [17], [18], [23], [26]	Hippocampal learning
	Generative	• Mitigation of CF • Less memory overhead	• Complexity introduced by generative models • Risk of generating unrealistic samples	[19], [20], [21], [22], [24], [25]	
Regularization	Weight-regularization	• Space efficiency • Positive forward and backward transfer • Outperforms other methods in some scenarios	• Balance required for regularization strength • Requires task identity at inference	[12], [31], [11], [33], [1]	Synaptic Plasticity
	Knowledge-distillation	• Space efficiency • Positive forward and backward transfer	• Balance required for regularization strength • Task identity required at inference	[32]	
Parameter isolation	Dynamic-architecture	• Minimized interference between tasks • Network capacity adaptation • Positive forward transfer	• Scalability concerns with parameter growth • Feasibility issues with network expansion	[37], [38], [39], [40]	Selective learning & Brain Modularity
	Fixed-architecture	• Minimized interference between tasks • No network growth • Maximal stability	• Explicit allocation of network capacity per task • Not suitable for long task sequences	[41], [42], [10], [36]	

a KUKA LWR arm to new obstacles using online learning, successfully transferring from simulation to real-world settings. Wang et al. [33] evaluate a motion planner trained with reinforcement learning, demonstrating a robot's ability to navigate changing target positions and avoid obstacles in real time. Similarly, Liu et al. [47] test LLfN on a Clearpath Jackal robot, focusing on dynamic adaptation.

(ii) Metrics for Continual Learning Evaluation

Among the most commonly employed metrics cited in surveyed papers is *accuracy* — the proportion of correct predictions made by the model on a given dataset ([46], [17], [18], [39], [19], [21] [24], [16], [20], [12], [31], [32], [11], [37], [38], [41], [10], [48]). Additionally, *forgetting* is frequently assessed ([21], [23], [17], [16], [19], [20], [12], [31], [10], [25]), which depicts the degree to which learning new tasks negatively impacts the model's performance on previously learned tasks. *Transfer* is another commonly used metric ([23], [17], [40], [41]), indicating the model's capacity to apply knowledge from one task to improve performance on another. This can either be in the form of forward transfer (building on previous knowledge) or backward transfer (interference with previous learning). **Appendix A** contains a full list of the metrics, including additional context-specific metrics, along with their definitions and sources.

3.3 Q3 - Practical uses of CL

In this section, we will introduce a categorization of practical uses of CL in the context of embodied agents. Here, we make the distinction between (i) embodied *physical* agents, which operate in the real world with tangible interactions, and (ii) embodied *virtual* agents, which exist within digital or simulated environments. For each scenario, we will provide specific examples of CL being employed in practice.

(i) Embodied Physical Agents

Embodied physical agents, such as robots, autonomous vehicles, and drones, interact with their surroundings through sensors and actuators. These agents require CL to adapt to dynamic and unpredictable environments, learn new tasks without forgetting previous ones, and improve performance over time based on continuous feedback. Nine papers [47], [51], [25], [52], [21], [24], [48], [22], [23] discuss the adaptation of CL methods inside embodied physical agents.

Lifelong navigation: This category involves a robot learning to navigate through multiple environments, using a fixed global planner. Liu et al. [47] describe a robot learning to navigate multiple environments using a policy which maps sensor inputs to actions. In this scenario, GEM [17] is used to prevent CF. In a similar vein, Dhakan et al. validate their modular continuous learning framework using a mobile robot in a dynamic environment [51]. The robot autonomously discovers and learns maintenance goals, such as navigating to specific locations. Using a goal discovery module, the robot identifies and clusters potential goals. A goal management module prioritizes these goals, while a learning module employs reinforcement learning to achieve them. Traoré et al. [25] explore CL in a real-life reinforcement learning scenario with a mobile robot. The robot is tasked with sequentially learning two navigation tasks: Target Reaching (TR) and Target Circling (TC). To achieve this, the robot employs distillation - combining knowledge from individual task policies into a single flexible policy capable of solving both tasks.

Table 2: Evaluation methods, task settings, and metrics commonly employed in continual learning research.

Evaluation Method (Examples)	Studies	Task Setting	Evaluation Metrics
Standard Image Classification Datasets (MNIST, CIFAR-100, CIFAR-10, FashionMNIST, NotMNIST, CORe-50)	[46], [17], [19], [12], [31], [39], [41], [10], [21], [20], [18], [38]	Incremental, Supervised learning	Accuracy, Forgetting, Transfer
Large-scale and Complex Image Datasets (ImageNet, Places365, iILSVRC, AudioSet, YouTube-8M, Caltech-101, CUB-200, Flowers, Stanford Cars, Stanford-40 Actions)	[16], [17], [32], [42], [24]	Incremental	Accuracy, Forgetting, Transfer
Synthetic Datasets (Permuted MNIST, Rotated MNIST)	[46], [17], [18], [12], [39], [41]	Incremental	Accuracy, Forgetting, Transfer
Domain-Specific Datasets (Street View House Number (SVHN), MIT indoor scene, AWA (Animals with Attributes), TrafficSigns, SUN-397)	[19], [41], [10], [20], [11], [37], [18], [38]	Incremental	Accuracy, Forgetting
Reinforcement Learning (Atari games, 3D maze tasks)	[40], [12], [41]	Reinforcement learning	Average score per episode, Task performance
Robotic and Real-World (Pepper robot, KUKA LWR arm simulation, Musculoskeletal system simulation, Drone object recognition)	[24], [22], [49], [23], [34], [50], [47], [51], [25], [52], [21], [48], [36], [33]	Incremental, Online adaptation	Accuracy, Forgetting, Task performance, Task completion, Adaptation

Thrun et al. [52] also describe an experiment using a HERO-2000 robot for autonomous navigation and mapping, using a CL model grounded in explanation-based neural network learning. The robot is equipped with sonar sensors, a laser rangefinder, and a video camera to gather environmental data and update its internal map while navigating.

Object classification: For classifying objects, the CBCL-PR framework [21] is applied in a scenario where a robot learns to classify household objects continuously with minimal labeled examples from humans. Ayub et al. [24] also implement CL on a humanoid Pepper robot [53] for household object classification, using Few-Shot Continual Active Learning (FoCAL). FoCAL integrates active learning and CL techniques, selecting informative samples based on prediction uncertainty (entropy). Another study by Ayub et al. [48] explores human teaching patterns with a CL robot for object recognition. Users labeled objects via a GUI, and the robot stored these images for training. During testing, the robot used its CL model to predict and point to objects correctly.

Robotic Limb Adaptation: In a study by Tannenberg et al. [22], a robotic arm is tasked with dynamically adapting its motion planning in response to changing environmental constraints, utilizing a form of CL that integrates intrinsic motivation signals to guide its online model adaptation process. In a similar fashion, Powers et al. [23] showcase a kitchen robot that learns tasks sequentially using limited demonstrations. The kitchen robot was tasked with picking up bottles, opening and closing a toaster oven, and adapting to varying kitchen setups. To help with mitigating CF, the authors describe the use methods such as replay buffers and regularization techniques, as discussed in Chapter 3.1.

(ii) Embodied Virtual Agents

Embodied virtual agents are digital entities operating within virtual or simulated environments, interacting with users and/or other virtual elements via graphical interfaces. Unlike physical agents using sensors and actuators in the real world, virtual agents exist solely in computer-generated settings. Despite these differences, virtual agents play a role in the *Sim2real* (simulation to reality) process. *Sim2real* leverages virtual environments to train and develop agents for deployment in real-world scenarios, offering advantages in cost, safety, and scalability. Five papers [36], [34], [49], [33], [50] discuss CL methods adapted for embodied virtual agents.

Simulated Robotic Task Learning: Say et al. propose a proof of concept for a practical application of CL by implementing a robotic arm in a simulated environment to autonomously learn and switch between multiple tasks [36]. The model uses a novel task arbitration mechanism based on Learning Progress (LP), allowing the agent to decide which task to engage with next and when to transfer knowledge. This approach enables the robot to learn interleaved and partially, mimicking human developmental learning. The tasks the robot had to perform involved predicting the effects of pushing a soccer ball in various environments.

Analogously, Chen et al. [34] recently explored continual reinforcement learning with a simulated musculoskeletal system and redundant robotic arm. Their method uses recurrent neural networks (RNNs) and shared hyperparameters to facilitate sequential task learning. To prevent CF, the authors employe a regularization-like method with projectors and Orthogonal Weight Modification (OWM). Projectors align new learning with previous knowledge, while OWM updates weights without interfering with past learning. Using this approach, the robotic arm is able to learn and perform tasks such as reaching and manipulating objects in different ranges and gravity environments, without forgetting how to perform previous actions. In a study by Rusu et al. [49], progressive networks are employed to transfer learning from simulation to real-world robotic tasks. The study involves training simulated robotic arms in the MuJoCo physics simulator and transferring the learned policies to physical robots, such as the Jaco arm. The agents were tasked with performing a reacher task, maneuvering towards visual targets in specified areas. Progressive networks were utilized to expand the network architecture for each new task while preserving connections to previously learned features.

Simulated Lifelong Navigation: Lifelong navigation has been explored through virtual embodied agents, as exemplified in the study by Wang et al. [33]. In this study, a virtual agent uses EWC [12] to retain knowledge from previous navigation tasks. The agent is first trained in a simple environment, then progresses to more complex ones, with simulations conducted in V-REP (Virtual Robot Experimentation Platform). Continuous interaction with the environment involved updating an actor and a critic network (responsible for determining actions and evaluating their value, respectively), with EWC adjustments ensuring important weights were preserved. Navigational lifelong techniques are also investigated in drones. Brown et al. [50] explore Uncertainty Modulated Learning (UML) [54] for lifelong learning in drones. UML measures uncertainty, enabling adaptive behavior and self-supervised learning. In AirSim simulations, drone agents performed object recognition tasks, adapting based on uncertainty signals to improve navigation and performance.

4 Discussion & Concluding Remarks

Discussion: In this study, we explored the application of the continual learning cognitive framework in embodied agents by addressing three key research sub-questions. These subquestions provided a broad overview of the approaches for CL, the methods of performance evaluation of CL systems, and practical examples of CL in real-world scenarios.

Firstly, **Q1** results highlighted that the methods and means for achieving CL in autonomous agents vary significantly. The three main approaches — replay-based, regularization-based, and parameter isolation methods — each offer distinct advantages and disadvantages, and their effectiveness is context-dependant. Replay-based methods, inspired by the human hippocampal system, utilize stored or generated samples from past tasks to mitigate forgetting. Regularization-based methods, which draw from synaptic plasticity principles, impose constraints on network parameters to retain old knowledge. Parameter isolation methods, based on the modularity of the brain, assign distinct parameter subsets to different tasks, either through dynamic or fixed architectures.

Moving forward, **Q2** explored evaluating CL methods, crucial for assessing their usability and efficacy. Evaluation criteria vary across incremental learning, supervised learning, reinforcement learning, and online adaptation settings. Of particular interest in embodied agent contexts is the online adaptation setting, where agents continuously adapt to real-time changes, typical for autonomous systems. Each setting employs specific metrics to gauge various aspects such as accuracy, forgetting rates, and transfer capability, which are among the most commonly highlighted in the findings.

Finally, **Q3** findings showcased practical applications for both physical and virtual embodied agents. The current field of physical embodied agents mainly consists of navigation and object classification tasks, being limited to simple scenarios such as indoor navigation, and various household object classification settings. However, more complex agents in the form of robotic limbs have been shown to employ CL in order to adapt to dynamic environments in real time. Despite this, with only a limited number of studies to account for such practical applications of physical agents, there remains ample room for further development and exploration. Embodied virtual agents share a similar trajectory, with literature largely being focused on small scale virtual navigation scenarios and simulated robotic tasks. Such scenarios do however hold significant potential for *Sim2real* experiments, where insights gained from simulated environments can be tested and refined before being applied in real-world settings.

One crucial remark that we would like to acknowledge is the apparent discrepancy between the evaluation and the practical applications for CL in embodied agents. While CL evaluation methods primarily focus on dataset benchmarks (see **Appendix B**), the practical applications in embodied agents often involve more complex and dynamic environments that pose unique challenges not fully captured by traditional metrics. For instance, evaluations rely heavily on image classification datasets like MNIST, CIFAR, and ImageNet, which are essential and convenient for benchmarking, but do not capture the full spectrum of real-world scenarios faced by embodied agents. In contrast, embodied agents operate in unpredictable environments where factors such as sensor noise, hardware limitations and uncertainty play significant roles. This observation is also echoed by Lesort et al., who mention that while CL is inherently born for robotics, "most of CL approaches are not robotics related and rather focus on experiments on image processing or simulated environments" [43, p. 20].

While part of the studies make an effort to demonstrate the potential uses for CL in practical scenarios, these are fewer in number compared to those that primarily focus on standardized dataset evaluation. This poses an issue as it limits our understanding of how CL algorithms perform in real-world conditions that embodied agents must navigate, potentially hindering the development of robust, adaptable systems capable of handling the complexities of real-world environments.

Conclusion: While CL shows great potential for creating adaptive and intelligent embodied agents, significant gaps remain between theoretical evaluations and practical applications. Current studies predominantly rely on standardized datasets, failing to capture the complexities of the real world. Future research should prioritize developing and testing CL algorithms in dynamic and unpredictable settings to ensure robustness and adaptability. Bridging this gap is essential for advancing the practical deployment of CL in embodied agents, ultimately leading to more capable and resilient autonomous systems. Nonetheless, the field is rapidly evolving, with promising advancements already demonstrating the potential for transformative applications in practical scenarios.

5 Responsible Research

To ensure that this paper adheres to commonly adopted standards of academic integrity, we have decided to follow the principles recommended by the Netherlands Code of Conduct for Research Integrity [55]. The proposed principles are *honesty*, *scrupulousness transparency*, *independence* and *responsibility*. In the following sections, we will present the steps and measures taken to adhere to these principles.

5.1 Reproduciblity & Transparency

To ensure the *reproducibility* and *transparency* of this research, we followed the PRISMA reporting guideline for systematic reviews. All methods for gathering and selecting the papers were detailed in Chapter 2. Moreover, a diagram was provided to illustrate the exact amount of papers included at each step during the literature search process. The exact query and specific search parameters and modifications were additionally provided. All the papers utilized in the study were cited and can be found in the References section. All the findings were presented objectively within the Results section of the paper, and no data that could potentially affect or skew the results was withheld during this process. Furthermore, our use of large language models (LLM's) is made explicit in **Appendix C** in order to account for how these models were employed throughout the study and how they influenced the research. This was done in an effort to adhere to the principle of *honesty*.

We would also like to acknowledge and address our utilization of Google Scholar as a search platform for literature identification. The use of articles identified through Google Scholar has been restricted to supporting or background information for **Sections 3.1** and **Chapter 1**. This was done in order to reduce confirmation bias, which represents a tendency to seek and favor information that confirms one's preexisting beliefs or hypotheses. Confirmation bias was additionally prevented by limiting the results to only the papers identified through either the queried databases and citation chaining. *Scrupulousness* was achieved by carefully evaluating papers at each step during the process. Stringent inclusion/exclusion criteria were utilized, and papers were filtered based on their relevancy to the topic. Each paper selected for inclusion in the final study underwent analysis in full-text format, and was subjected to multiple reviews and examinations when conducting the information extraction.

Furthermore, this study achieved *independence*, meaning that the research was conducted without undue influence or bias resulting from personal motives. Our affiliation with TU Delft University implies a commitment to scholarly standards and practices. As a student, the author conducted the research as part of the CSE3000 Research Project course. Thus the research was performed within an educational framework, where the primary goal is learning and knowledge dissemination rather than serving any personal agenda or interest.

5.2 Limitations

Several limitations were encountered during the research. Firstly, the study was conducted over a period of approximately 9 weeks, which represents a relatively short time frame for conducting a systematic literature review. As a consequence, the author had to take certain measures to ensure the feasibility, integrity and validity of the work. Firstly, the papers were strictly filtered based on hard inclusion/exclusion criteria to ensure that a manageable amount of papers would be evaluated in the end, with the possibility of maintaining a balanced workload. Moreover, studies were limited to only the English language to align with the author's proficiency and ensure accurate interpretation and analysis. These measures do impose the risk some relevant studies may have been inadvertently excluded from this review.

Secondly, the author of this study does not possess extensive expertise in the field of continual learning and embodied agents. As a third-year bachelor student pursuing studies in Computer Science and Engineering (CSE), the author's knowledge and experience are still developing, which may have influenced the depth and breadth of the review.

Thirdly, due to time constraints and resource limitations, the search was confined to a specific set of databases relevant to the study, and did not include other potentially relevant sources such as grey literature or unpublished studies. This narrow scope may have resulted in the omission of certain information or alternative perspectives on the topic.

Lastly, the methodologies/proposed frameworks of certain papers were not detailed extensively in this paper due to word limit constraints. For a comprehensive understanding, readers are encouraged to refer to the following papers: [22], [24], [25], [1], [36], [26]. These papers were included for the sake of reproducibility and to ensure a robust foundation for future investigations.

References

[1] Z. Chen, B. Liu, R. Brachman, P. Stone, and F. Rossi, *Lifelong Machine Learning*, 2nd ed. Morgan & Claypool Publishers, 2018.

[2] E. Verwimp, R. Aljundi, S. Ben-David, M. Bethge, A. Cossu, A. Gepperth, T. L. Hayes, E. Hüllermeier, C. Kanan, D. Kudithipudi, C. H. Lampert, M. Mundt, R. Pascanu, A. Popescu, A. S. Tolias, J. van de Weijer, B. Liu, V. Lomonaco, T. Tuytelaars, and G. M. van de Ven, "Continual learning: Applications and the road forward," 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2311.11908

[3] S. V. Mehta, D. Patil, S. Chandar, and E. Strubell, "An empirical investigation of the role of pre-training in lifelong learning," 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2112.09153

[4] M. Delange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, "A continual learning survey: Defying forgetting in classification tasks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1–1, 2021. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2021.3057446

[5] M. S. Gazzaniga, R. B. Ivry, and G. R. Mangun, *Cognitive Neuroscience: The Biology of the Mind*. W. W. Norton, 2013.

[6] L. F. Abbott and S. B. Nelson, "Synaptic plasticity: taming the beast," *Nature Neuroscience*, vol. 3, no. 11, pp. 1178–1183, Nov 2000. [Online]. Available: https://doi.org/10.1038/81453

[7] E. T. Cowan, A. C. Schapiro, J. E. Dunsmoor, and V. P. Murty, "Memory consolidation as an adaptive process," *Psychonomic Bulletin & Review*, vol. 28, no. 6, pp. 1796–1810, Dec 2021. [Online]. Available: https://doi.org/10.3758/s13423-021-01978-x

[8] R. M. French, "Catastrophic forgetting in connectionist networks," *Trends in Cognitive Sciences*, vol. 3, no. 4, pp. 128–135, 1999. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364661399012942

[9] L. Wang, X. Zhang, H. Su, and J. Zhu, "A comprehensive survey of continual learning: Theory, method and application," 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2302.00487

[10] J. Serrà, D. Surís, M. Miron, and A. Karatzoglou, "Overcoming catastrophic forgetting with hard attention to the task," 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1801.01423

[11] R. Aljundi, F. Babiloni, M. Elhoseiny, M. Rohrbach, and T. Tuytelaars, "Memory aware synapses: Learning what (not) to forget," 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1711.09601

[12] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.1611835114

[13] G. M. Van De Ven and A. S. Tolias, "Three scenarios for continual learning," *arXiv (Cornell University)*, Jan. 2019. [Online]. Available: https://arxiv.org/abs/1904.07734

[14] B. Wickramasinghe, G. Saha, and K. Roy, "Continual learning: A review of techniques, challenges and future directions," *IEEE Transactions on Artificial Intelligence*, vol. 1, no. 01, pp. 1–21, dec 5555. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/TAI.2023.3339091

[15] H. Bae, S. Song, and J. Park, "The present and future of continual learning," in *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, 2020, pp. 1193–1195. [Online]. Available: http://dx.doi.org/10.1109/ICTC49870.2020.9289549

[16] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "icarl: Incremental classifier and representation learning," 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1611.07725

[17] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," 2022. [Online]. Available: https://doi.org/10.48550/arXiv.1706.08840

[18] A. Chaudhry, M. Ranzato, M. Rohrbach, and M. Elhoseiny, "Efficient lifelong learning with a-gem," 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1812.00420

[19] H. Shin, J. K. Lee, J. Kim, and J. Kim, "Continual learning with deep generative replay," 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1705.08690

[20] R. Kemker and C. Kanan, "Fearnet: Brain-inspired model for incremental learning," 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1711.10563

[21] A. Ayub and A. R. Wagner, "Cbcl-pr: A cognitively inspired model for class-incremental learning in robotics," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 15, no. 4, p. 2004–2013, Dec. 2023. [Online]. Available: http://dx.doi.org/10.1109/TCDS.2023.3299755

[22] D. Tanneberg, J. Peters, and E. Rueckert, "Efficient online adaptation with stochastic recurrent neural networks," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 2017, pp. 198–204. [Online]. Available: https://doi.org/10.1109/HUMANOIDS.2017.8246875

[23] S. Powers, A. Gupta, and C. Paxton, "Evaluating continual learning on a home robot," 2023. [Online]. Available: https://proceedings.mlr.press/v232/powers23a.html

[24] A. Ayub and C. Fendley, "Few-shot continual active learning by a robot," 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2210.04137

[25] R. Traoré, H. Caselles-Dupré, T. Lesort, T. Sun, N. Díaz-Rodríguez, and D. Filliat, "Continual reinforcement learning deployed in real-life using policy distillation and sim2real transfer," 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1906.04452

[26] D. Rolnick, A. Ahuja, J. Schwarz, T. P. Lillicrap, and G. Wayne, "Experience replay for continual learning," 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1811.11682

[27] B. Bagus and A. Gepperth, "An investigation of replay-based approaches for continual learning," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, Jul. 2021. [Online]. Available: http://dx.doi.org/10.1109/IJCNN52387.2021.9533862

[28] E. Verwimp, M. De Lange, and T. Tuytelaars, "Rehearsal revealed: The limits and merits of revisiting samples in continual learning," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 9365–9374. [Online]. Available: https://doi.org/10.48550/arXiv.2104.07446

[29] J. L. McClelland, B. L. McNaughton, and R. C. O'Reilly, "Why there are complementary learning systems in the hippocampus and neocortex: Insights

from the successes and failures of connectionist models of learning and memory." *Psychological Review*, vol. 102, no. 3, p. 419–457, 1995. [Online]. Available: https://doi.org/10.1037/0033-295X.102.3.419

[30] B. Rasch and J. Born, "About sleep's role in memory," *Physiological Reviews*, vol. 93, no. 2, p. 681–766, 2013. [Online]. Available: https://doi.org/10.1152/physrev.00032.2012

[31] X. Liu, M. Masana, L. Herranz, J. V. de Weijer, A. M. Lopez, and A. D. Bagdanov, "Rotate your networks: Better weight consolidation and less catastrophic forgetting," 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1802.02950

[32] Z. Li and D. Hoiem, "Learning without forgetting," 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1606.09282

[33] N. Wang, D. Zhang, and Y. Wang, "Learning to navigate for mobile robot with continual reinforcement learning," in *2020 39th Chinese Control Conference (CCC)*, 2020, pp. 3701–3706. [Online]. Available: https://doi.org/10.23919/CCC50068.2020.9188558

[34] J. Chen, Z. Chen, C. Yao, and H. Qiao, "Neural manifold modulated continual reinforcement learning for musculoskeletal robots," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 16, no. 1, pp. 86–99, 2024. [Online]. Available: https://doi.org/10.1109/TCDS.2022.3231055

[35] A. Citri and R. C. Malenka, "Synaptic plasticity: Multiple forms, functions, and mechanisms," *Neuropsychopharmacology*, vol. 33, no. 1, p. 18–41, 2007. [Online]. Available: https://doi.org/10.1038/sj.npp.1301559

[36] H. Say and E. Oztop, "A model for cognitively valid lifelong learning," *2023 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2023. [Online]. Available: https://doi.org/10.1109/robio58561.2023.10355028

[37] Y.-X. Wang, D. Ramanan, and M. Hebert, "Growing a brain: Fine-tuning by increasing model capacity," 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1907.07844

[38] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, "Lifelong learning with dynamically expandable networks," 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1708.01547

[39] X. Li, Y. Zhou, T. Wu, R. Socher, and C. Xiong, "Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 3925–3934. [Online]. Available: https://proceedings.mlr.press/v97/li19m.html

[40] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," 2022. [Online]. Available: https://doi.org/10.48550/arXiv.1606.04671

[41] C. Fernando, D. Banarse, C. Blundell, Y. Zwols, D. Ha, A. A. Rusu, A. Pritzel, and D. Wierstra, "Pathnet: Evolution channels gradient descent in super neural networks," 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1701.08734

[42] A. Mallya and S. Lazebnik, "Packnet: Adding multiple tasks to a single network by iterative pruning," 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1711.05769

[43] T. Lesort, V. Lomonaco, A. Stoian, D. Maltoni, D. Filliat, and N. Díaz-Rodríguez, "Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges," 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1907.00182

[44] R. Desimone and J. Duncan, "Neural mechanisms of selective visual attention," *Annual review of neuroscience*, vol. 18, pp. 193–222, 02 1995. [Online]. Available: https://doi.org/10.1146/annurev.ne.18.030195.001205

[45] A. Kelkar and J. D. Medaglia, *Evidence of Brain Modularity*. Cham: Springer International Publishing, 2018, pp. 1–10. [Online]. Available: https://doi.org/10.1007/978-3-319-16999-6_2422-1

[46] M. Kim and W. Saad, "Analysis of the memorization and generalization capabilities of ai agents: Are continual learners robust?" 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2309.10149

[47] B. Liu, X. Xiao, and P. Stone, "A lifelong learning approach to mobile robot navigation," 2021. [Online]. Available: https://doi.org/10.48550/arXiv.2007.14486

[48] A. Ayub, J. Mehta, Z. D. Francesco, P. Holthaus, K. Dautenhahn, and C. L. Nehaniv, "How do human users teach a continual learning robot in repeated interactions?" 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.00123

[49] A. A. Rusu, M. Vecerík, T. Rothörl, N. Heess, R. Pascanu, and R. Hadsell, "Sim-to-real robot learning from pixels with progressive nets," in *1st Annual Conference on Robot Learning, CoRL 2017, Mountain View, California, USA, November 13-15, 2017, Proceedings*, ser. Proceedings of Machine Learning Research, vol. 78. PMLR, 2017, pp. 262–270. [Online]. Available: http://proceedings.mlr.press/v78/rusu17a.html

[50] R. Brown, A. Brna, J. Cook, S. Park, and M. Aguilar-Simon, "Uncertainty-driven control for a self-supervised lifelong learning drone," in *IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*, 2022, pp. 5053–5056. [Online]. Available: https://doi.org/10.1109/IGARSS46834.2022.9884516

[51] P. Dhakan, K. E. Merrick, I. Rañó, and N. H. Siddique, "Modular continuous learning framework," in *2018*

Joint IEEE 8th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob). IEEE, 2018, pp. 107–112.

[52] S. Thrun and T. M. Mitchell, “Lifelong robot learning,” *Robotics and Autonomous Systems*, vol. 15, no. 1, pp. 25–46, 1995, the Biology and Technology of Intelligent Autonomous Agents. [Online]. Available: https://www.sciencedirect.com/science/article/pii/092188909500004Y

[53] “A mass-produced sociable humanoid robot: Pepper: The first machine of its kind,” vol. 25. [Online]. Available: https://doi.org/10.1109/mra.2018.2833157

[54] A. P. Brna, R. C. Brown, P. M. Connolly, S. B. Simons, R. E. Shimizu, and M. Aguilar-Simon, “Uncertainty-based modulation for lifelong learning,” *Neural Networks*, vol. 120, pp. 129–142, 2019, special Issue in Honor of the 80th Birthday of Stephen Grossberg. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608019302722

[55] “Netherlands code of conduct for research integrity,” https://www.nwo.nl/en/netherlands-code-conduct-research-integrity, 2018, accessed: 2024-06-02.

A Evaluation Metrics for Continual Learning

Table 3: Metrics, definitions, and sources for evaluation of CL systems

Metric	Definition	Source
Accuracy & Variations	Degree of correctness of the model's predictions compared to the ground truth labels. Commonly measured as the proportion of correct predictions made by the model on the test partition of each dataset.	[46], [17], [18], [39], [19], [21], [24], [16], [20], [12], [31], [32], [11], [37], [38], [41], [10], [48]
Forgetting	The extent to which learning new tasks negatively impacts performance on previously learned tasks.	[21], [23], [17], [16], [19], [20], [12], [31], [10], [25]
Transfer	The model's ability to apply learned knowledge from one task to another. Can either be Forward Transfer (FWT) - measurement in performance on future tasks after knowledge learned from previous tasks, or Backward Transfer (BWT) - measurement in performance on previously learned tasks after learning new tasks.	[23], [17], [40], [41]
Generalization	The ability of a model to perform well on unseen data.	[46]
Efficiency	The computational resources and time required by the model to learn and perform tasks.	[38]
Capacity	The model's ability to accommodate new information without significant degradation of performance on previous tasks.	[38]
Performance Change Over Time	The variation in model performance as it learns new tasks over time.	[38]
Model Size	The memory footprint of the model, typically in terms of the number of parameters.	[39], [42]
Classification Error	The proportion of incorrect predictions made by the model.	[41]
Training Time	The duration taken by the model to train on the dataset(s).	[41]
Effect Prediction Error (Mean Absolute Error, MAE)	The average magnitude of errors in predictions, without considering their direction.	[36]
Normalized Mean Reward	The average reward normalized over a set of tasks or episodes, reflecting overall task performance.	[25]
Traversal Time	The average time taken for navigation from start to goal.	[47]
Recovery Behaviors	The number of instances where the robot engages in actions to recover from navigation errors.	[47]
Collisions	The number of times the robot impacts obstacles during navigation.	[47]
Learning Efficiency	The speed and stability with which the system learns new environments while retaining knowledge of previous ones.	[47]

B Distribution of evaluation methods

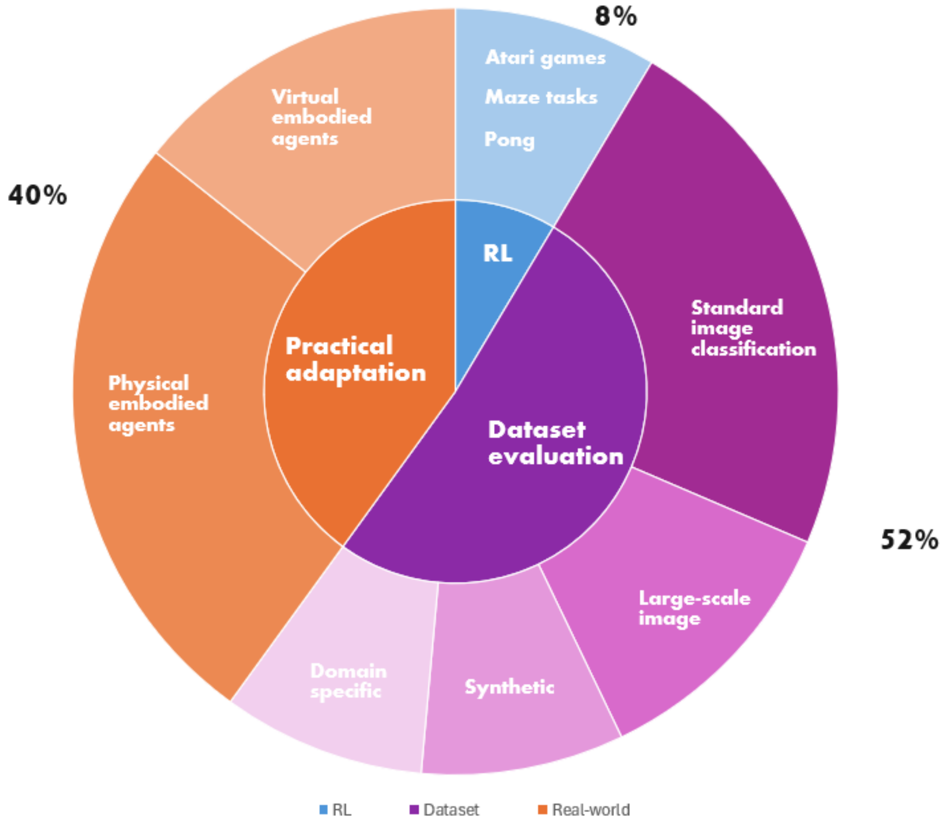

Figure 3: Distribution of evaluation methods used in CL research. Studies that include reinforcement learning evaluation: [40], [12], [41]. Studies that include practical adaptation/real world adaptation: [24], [22], [49], [23], [34], [50], [47], [51], [25], [52], [21], [48], [36], [33]. Studies that include dataset evaluation: [46], [17], [19], [12], [31], [39], [41], [10], [21], [20], [18], [38], [32], [42], [24], [16], [11], [37]

C Use of Large Language Models

This research mainly utilized LLM's (Large Language Models) as a means of synonym identification, word completion and word removal. We used OpenAI's GPT-3.5 [5] model for this purpose. We utilized several prompts that follow a similar structure, examples of which are provided below in **Figures 4, 5, 6, 7, 8, and 9**.

1. Synonym prompts

Replace the word "following" so that there is less repetition: "In the following section, we will analyze the methods available for CL. Following this, in section 2.2, we will provide an overview of the ..."

Figure 4: Example of prompt for finding an appropriate word in a given sentence

Find a synonym for "experiences" : "The hippocampal system records experiences as they occur throughout the day and subsequently replays these stored memories back to the neocortex overnight"

Figure 5: Example of prompt for finding synonyms for a certain word in a paragraph to avoid repetition

2. Word completion prompts

Please find a word for _ in this sentence: "Lorem ipsum _ dolor, consectetur adipiscing..."

Figure 6: Example of prompt for finding an appropriate word in a given sentence

Find words for _ in this sentence: "Lorem _ dolor _, consectetur elit..."

Figure 7: Example of prompt for finding an appropriate word in a given sentence

3. Word removal prompts

"While generative replay addresses this issue, it may introduce unnecessary complexity and the risk of generating unrealistic samples." Are there unnecessary words that can be removed?

Figure 8: Example of prompt for listing irrelevant/filler words in a given sentence

"While generative replay addresses this issue, it may introduce unnecessary complexity and the risk of generating unrealistic samples." Remove filler words and please list the removed words.

Figure 9: Example of prompt for removing irrelevant/filler words in a given sentence

[5] For more information about the GPT-3.5 model, visit https://www.openai.com