Continual Reinforcement Learning in 3D Non-stationary Environments

Continual Reinforcement Learning in 3D Non-stationary Environments

Vincenzo Lomonaco
University of Bologna
Italy
vincenzo.lomonaco@unibo.it

Karan Desai
University of Michigan
United States
kdexd@umich.edu

Eugenio Culurciello
Purdue University
United States
euge@purdue.edu

Davide Maltoni
University of Bologna
Italy
davide.maltoni@unibo.it

Abstract

High-dimensional always-changing environments constitute a hard challenge for current reinforcement learning techniques. Artificial agents, nowadays, are often trained off-line in very static and controlled conditions in simulation such that training observations can be thought as sampled i.i.d. from the entire observations space. However, in real world settings, the environment is often non-stationary and subject to unpredictable, frequent changes. In this paper we propose and openly release CRLMaze, a new benchmark for learning continually through reinforcement in a complex 3D non-stationary task based on ViZDoom and subject to several environmental changes. Then, we introduce an end-to-end model-free continual reinforcement learning strategy showing competitive results with respect to four different baselines and not requiring any access to additional supervised signals, previously encountered environmental conditions or observations.

1. Introduction

In the last decade we have witnessed a renewed interest and major progresses in reinforcement learning (RL) especially due to recent deep learning developments [3]. State-of-the-art RL agents are now able to tackle fairly complex problems involving high-dimensional perceptual data, which were even unthinkable to solve without explicit supervision before [34, 45].

However, much of these progresses have been made in very narrow and isolated tasks, often in simulation with thousands of trials and with the common assumption of a stationary, fully-explorable environment from which to sample observations i.i.d. or approximately so. Even in the case of more complex tasks and large environments, a common technique known as *memory replay* [34, 16, 23] is adopted, consisting in storing old observations in an external memory buffer to simulate an i.i.d. sampling. Roughly the same result can be also achieved through multiple replicas of the agent randomly spawned in the environment and collecting several different observations at the same time, hence approximating the coverage of the entire observations space [33].

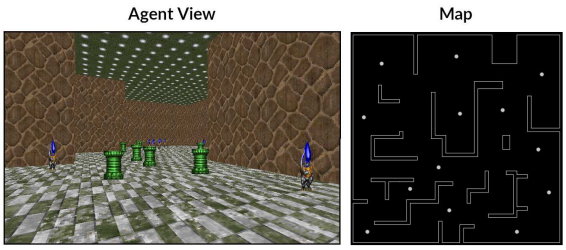

Figure 1. The 3D maze environment developed with *ZDoom* and *Slade3*. On the left, an example image from the point of view of the agent is reported. On the right, the planar view of the maze structure is shown. White points on the map represent random spawning points used by the agent during both training and test episodes. Better viewed in colors.

Nevertheless, dealing with single agents in the real-world and subject to computational and memory constraints these solutions suddenly appear less practical. This is especially true with *always-changing* environments and multi-task settings where re-sampling is impossible and storing old observations is no longer an option since it would require a constant grow in terms of memory consumption and computational power needed to re-process these observations. On the other hand, if the memory replay buffer is limited in size, the agent suddenly incurs in the phenomenon known in literature as *catastrophic forgetting*, being unable to re-

tain past knowledge and skills in previously encountered environmental conditions or tasks [31, 41, 13, 24].

Learning continually from data is a topic of steadily growing interest for the machine learning community and concerns itself with the idea of improving adaptation and generalization capabilities of current machine learning models by providing efficient updating strategies when new observations become available without *storing*, *re-sampling* or *re-processing* the previous ones (or as little as possible). While much of the focus and research efforts in continual learning have been devoted to multi-task settings (where a single model is exposed to a sequence of distinct and well-defined tasks over time) [36, 6], several practical scenarios would also benefit artificial agents that learn continually in complex non-stationary reinforcement environments.

In this paper, we focus on the more complex problem of a single task, constantly changing over time. As it has been shown in some supervised contexts, the clear separation in tasks (i.i.d. by parts), along with the presence of a supervised *"task label' t* [11], greatly helps taming the problem of forgetting [30, 2]. We argue that learning without any notion of task or distributional shift (both during training and inference), at least from an external oracle, is a more natural approach worth pursuing for improving the autonomy of every artificial learning agent.

The original contributions of this paper can be summarized as follows:

- We design and openly release a new benchmark, *CRLMaze* based on VizDoom [20], for assessing continual reinforcement learning (CRL) techniques in an always-changing object-picking task. *CRLMaze* is composed of 4 scenarios (*Light*, *Texture*, *Object*, *All*) of incremental difficulty and a total of 12 maps. To the best of our knowledge, this is one of the first attempts to scale continual reinforcement learning to complex 3D non-stationary environments.

- We provide 4 continual reinforcement learning baselines for each scenario.

- We propose an end-to-end, model-free continual reinforcement learning strategy, *CRL-Unsup*, which is agnostic to the changes in the environment and does not exploit a *memory replay* buffer or any distribution-specific *over-parametrization*, showing competitive results with respect to the supervised baselines (see section 4). The core insight of our strategy is to consolidate past memories through regularization as in [21], but proportionally to the difference between the expected reward and the actual reward (hence encoding a novel environmental condition in which the agent is unable to operate).

All the environments and the code to reproduce and expand the experiments discussed in this paper are available at: https://github.com/vlomonaco/crlmaze.

The rest of the paper is organized as follows: in Section 2, the *CRLMaze* benchmark is described; In Section 3, the CRL strategies used for the experiments reported in Section 4 are outlined. Finally, in Section 5, key questions and future work in this area are discussed.

2. CRLMaze: a 3D Non-stationary Environment

Continual Learning (CL) in reinforcement learning environments is still in its infancy. Despite the the obvious interest in applying CL to less supervised settings and the early, promising results in this context [40, 48], reinforcement learning tasks constitute a much more complex challenge where it is generally more difficult to disentangle the complexity introduced by distributional shifts from those introduced by the lack of a strong supervision.

It is also worth noting that state-of-the-art reinforcement learning algorithms and current hardware computational capabilities does not make experimentations and prototyping easily accomplished on complex environments where physical simulation constitute an heavy computational task per se. In a continual learning context, the problem becomes even harder since an exposition of the same model to sequential streams of observations is needed (and cannot be parallelized by definition). This is why recent reinforcement learning algorithms for continual learning have been tested only on arguably simple tasks of low/medium input space dimension and complexity [21, 1, 35].

Nevertheless, at the same time, state-of-the-art reinforcement learning algorithms have started tackling more complex problems in 3D static environments. *VizDoom* [20], followed soon after by other research platforms like *DeepMind Labs* [4] and *Malmo* [17], allowed researchers to start exploring new interesting research directions with the aim of scaling up current reinforcement learning algorithms.

VizDoom is a reinforcement learning API build around the famous *ZDoom* game engine and providing all the necessary utilities to train a RL agent in arbitrary complex environments. This framework is particularly interesting since it has been open-sourced to both Windows and Unix systems and it was already built on the idea of flexibility and customizability, allowing users to create custom maps and modify behaviorial responses of the environment through the simple *Action Code Script* (ACS) language.

In this paper, we propose an original 3D ViZDoom environment for continual reinforcement learning and an object-picking task named ***CRLMaze***[1] (see Fig. 1). The task consists of learning how to navigate in a complex maze and

[1] In particular, we used *Slade3* as the environment editor.

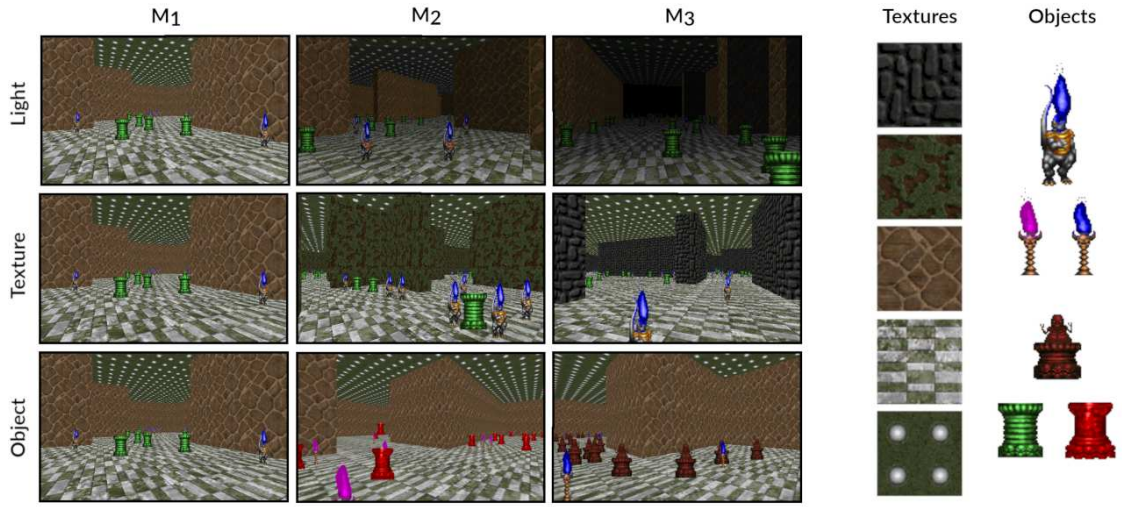

Figure 2. On the left, the environmental changes for each of scenario (*Light*, *Texture*, *Object*) in the 3D *CRLMaze*. For the *All* scenario, each map is the composition of all the environment variations introduced in the respective maps of all the other scenarios. In all the cases, changes are not gradual but occur abruptly at three (equidistant) points in time. On the right, textures and objects used for the *CRLMaze* scenarios. Better viewed in colors.

pick up *"column bricks"* while avoiding *"flaming lanterns"* (see Fig. 2). However, the environment in this case is *non-stationary* meaning that is subject to several environmental changes leading to major difficulties for standard reinforcement learning algorithms.

For properly assessing novel continual reinforcement learning strategies in the aforementioned 3D complex environment we split the benchmark in four different scenarios of incremental difficulty with respect to different environmental changes (see Fig. 2):

- ***Light***: In this scenario the illumination of the environment is altered over time. While intuitively this scenario may appear as one of the easiest, as we will see in the experimental section 4, it constitutes one of the most difficult since visual features from the environment do not change only in terms of RGB pixel magnitudes by a scalar factor, but also in terms of agent visibility (i.e. the radius in the 3D space up to which the RGB colors saturates to complete black), as shown in Fig. 2 (top row).

- ***Texture***: In this scenario walls textures are changed over time. The ability to pass over invariant features of the background is often taken for granted in many supervised tasks with state-of-the-art deep architectures [38]. However, as shown in the past, reinforcement learning agents are quite fragile also with respect to minor environmental changes [18].

- ***Object***: In this scenario the shapes and colors of the objects are changed over time. Invariance with respect to object shapes and colors is another important property every learning system should possess when facing real-world conditions where surrounding objects appearances are subject to constant changes due to deterioration and substitutions.

- ***All***: In this environment lights, textures as well as objects are subject to change over time. This scenario is also proposed with the idea of providing a comprehensive scenario for 3D environments in complex non-stationary settings, combining all the environmental condition variations proposed in the previous ones.

For all the scenarios, changes are not gradual but happening at three specific points equidistant in time (the total number of training episodes is fixed and considered a property of the environment) and practically implemented as different ZDoom *maps* faced sequentially ($M_1 \rightarrow M_2 \rightarrow M_3$, see Fig. 2). The agent is randomly spawned at fixed positions depicted as white points in Fig. 1 with a random visual angle. The environment starts with 75 randomly spawned *column* objects and 50 *lantern* objects. Catching a column increases the reward of 100 once collected while touching a lantern decreases the reward of 200. Even if in our exploratory experiments we noted that a shaping reward is not necessary to train the agents up to convergence, a weak shaping reward of 0.7 has been added when the *go-forward* action is chosen to improve environment exploration and ultimately speed-up learning convergence. A new object for each category is also randomly spawned every 3 ticks for roughly maintaining the amount of objects in the environment stable as when the objects are collected by the agent they disappear.

Table 1. Some common environments used for continual or meta reinforcement learning. The proposed benchmark, CRLMaze (bottom), shows significant advancements in terms of task complexity and non-stationary elements.

Environment Name	Input Dim.	3D	Non-Stationary Elements
Locomotion Environment [1]	14	yes	2 over 16 joints torques are scaled down by a constant factor.
MiniGrid [7]	6×6×3	no	"Competencies" introduced in a curriculum.
Catcher [39]	256×256×3	no	Vertical velocity of pellet increased of 0.03 from default 0.608.
Flappy Bird [39]	288×512×3	no	Pipe gap decreased 5 from default 100.
Krazy World [46]	10×10×3	no	Randomly generated worlds from the same distribution.
Mazes [46]	20×20	no	Randomly generated mazes from the same distribution.
Arcade Learning Environment [29]	210×160×3	no	60 different games available in the plaform.
CoinRun [10]	64×64×3	no	Randomly generated maps with 3 levels of difficulty.
CoinRun Platform [10]	64×64×3	no	Randomly generated maps from the same distribution.
RandomMazes [10]	[3×3 - 25×25]	no	Randomly generated mazes of different sizes.
CRLMaze	**320×240×3**	**yes**	Light/visibility, walls textures, object shape and colors are changed within the same object picking task.

In Tab. 1, we report some of the common environments and platforms used in the context of continual and meta-reinforcement learning and compare them with the proposed *CRLMaze*.

While still acknowledging the limited number of environmental variations introduced in the benchmark, we believe CRLMaze shows a significant advancement in terms of task complexity and non-stationary elements introduced with respect to recent environments made available by the community, being them usually of low input dimensionality, not often running on a complex 3D engine and with very limited non-stationary dynamics. For example, in [39], the *Catcher* environment is based on simple 2D physics and the only non-stationary element introduced is a change in the vertical velocity of the pallet which is only slightly increased of 5% from its default value.

3. Continual Reinforcement Learning Strategies

Learning over complex and large non-stationary environments is a hard challenge for current reinforcement learning systems. Recent works in this research area include meta-learning [12, 35, 1], hierarchical learning [50, 49] and continual learning approaches [21, 43, 37]. While both meta-learning and hierarchical learning work around the idea of imposing some structural dependencies among the learned concepts, continual learning is generally agnostic with this regard, being more focused on addressing the non-stationary nature of the underlying distributions [36].

Consolidating and preserving past memories while being able to generalize and learn new concepts and skills is a well known challenge for both artificial and biological learning systems, generally acknowledged as the *plasticity-stability* dilemma [32]. Since gradient-based architectures are generally skewed towards plasticity and prone to *catastrophic forgetting*, much of the research in continual learning with deep architectures has been devoted to the integration of consolidation processes in order to improve stability [6, 14].

However, the general focus of continual reinforcement learning research has been devoted to multi-task scenarios [21, 43] where consolidation can be achieved more easily and only when there is a change of task. *CRLMaze* constitutes a step forward in the evaluation of new continual reinforcement learning strategies that have to deal with substantial, unpredictable changes in the environment *within the same task* and without any additional supervised signal indicating (virtual or real) shifts in the underlying input-output distribution. We regard at this situation as the most realistic (and difficult) setting every agent should be able to deal within real-world conditions, where learning is mostly unsupervised and autonomous.

In recent literature, this problem has been tackled by using external generative models of the environment in order to detect big changes in input space [21]. However, recent evidences in behavioral experiments on rats suggests, more generally, behavioral correlates of synaptic consolidation especially when the subject is exposed to novel or strong external stimuli (e.g. a foot shock) [8, 9]. Following this inspiration, in this paper we propose a new strategy ***CRL-Unsup***, where the central idea is to consolidate memory only when a substantial difference between the expected reward and the actual one is detected, i.e. when the agent encounters an unexpected situation.

Hence, distributional shifts can be detected just by looking at the ability of the agent to actually perform the task: this can be approximated and practically implemented as the difference between a short-term (r^s_{mavg}) and a long-term (r^l_{mavg}) reward moving average that, when goes under a particular threshold (η), triggers the memory consolidation procedure[2]. The long-term moving average encodes the ex-

[2]It is interesting to note that a similar technique is also the basis of the *MACD indicator* [44] widely used in automated trading systems to detect

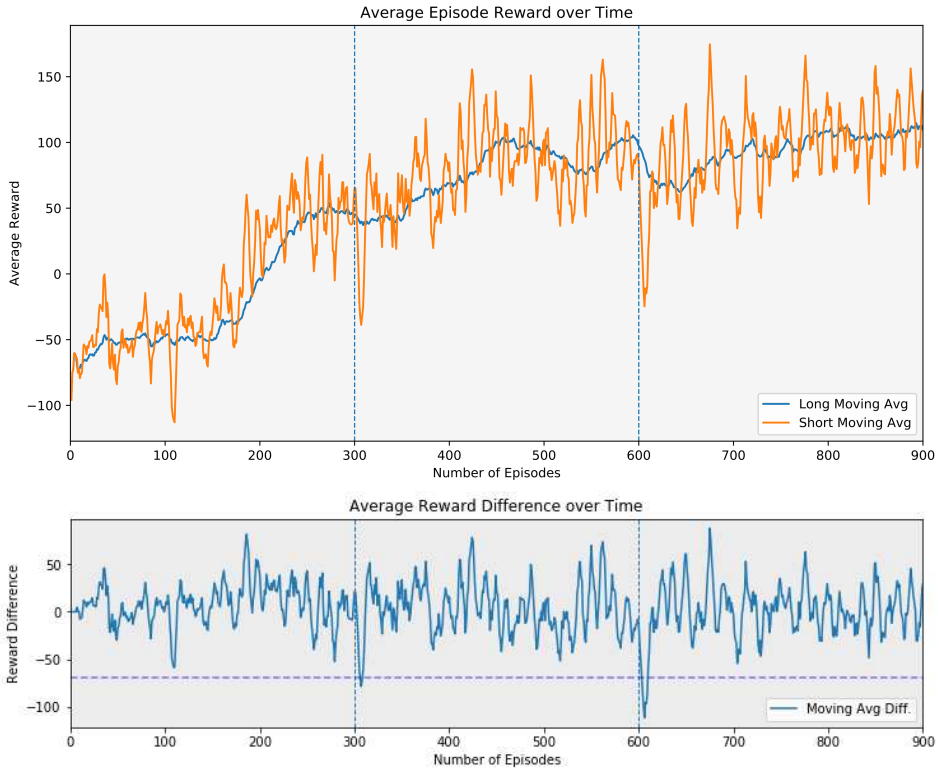

Figure 3. Short and long-term moving average (computed over 6 and 50 episodes, respectively) of the average cumulative reward *during training* in a single run of the *light* scenario. Dotted lines indicate when the environment is changed. In this example, the difference between the short-term and long-term moving average goes under $\eta = -70$ when the environment changes.

pected reward over a longer timespan, while the shorter one, an average of the currently received rewards where noise has been partially averaged out. This approach may not only signal changes in the environment affecting the performance of the agent but also possible changes in the reward function or instabilities of the learning process which may be mitigated through consolidation (similar to the regularization loss introduced in PPO [42]). Moreover, we do not use neither any distribution-specific over-parametrization nor any kind of memory replay as deemed necessary in [21, 43].

For consolidation in *CRL-Unsup* we employ the end-to-end regularization approach firstly introduced in [21] and known as *Elastic Weight Consolidation* (EWC): the basic idea is to preserve the parameters proportionally to their importance in the approximation of a specific distribution (i.e. the Fisher information). More efficient consolidation techniques through regularization derived from EWC have been recently proposed [43, 51, 30, 27]. However, for simplicity, we used its basic implementation. In eq. 1 and eq. 2 the loss function L of the *CRL-Unsup* strategy for a single consolidation step is reported[3] where L_{A2C} is the standard A2C loss function composed of the value and policy loss as defined in [33]; λ is an hyper-parameter encoding the strength of the consolidation (i.e. reducing plasticity); F_k is the Fisher information for the weight θ_k while θ_k^* indicates the optimal weight to consolidate. However, since the Fisher information can not be computed on the new data distribution, F is computed at fixed steps in times and only the latest one is used in the regularization term when the threshold is exceeded.

$$L = L_{A2C} + \frac{\lambda}{2} \cdot \sum_k F_k(\theta_k - \theta_k^*)^2 \tag{1}$$

$$\lambda = \begin{cases} \alpha & \text{if } r^s_{mavg} - r^l_{mavg} \leq \eta \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

In Figure 3, an example in the *light* scenario of the short and long-term moving average (computed over 6 and 50 episodes, respectively) of the training average cumulative reward is reported. In order to better compare and understand the performance of the aforementioned strategy, on each of the considered environmental changes (i.e. *light*, *texture*,

changing market conditions and issue buy/sell signals.

[3]Please note that in the basic EWC implementation a regularization term for each consolidation step needs to be added to the loss with a different F_k and θ_k^* for each weight θ_k (see interesting discussion in [43]).

object, *all*) four different baselines are here introduced and assessed:

1. ***Multienv***: This approach can be considered as a reference baseline and not properly a CRL strategy since it consists in training the agent over all the possible environmental conditions (i.e. maps M_1, M_2 and M_3) of each scenario at the same time. Having access to all the maps at the same time makes the distribution stationary and eliminates the *catastrophic forgetting* problem. It will be considered as an upper bound for the other strategies as generally acknowledged in continual learning [36, 26, 28].

2. ***CRL-Naive***: This approach, like the homonym strategy in the supervised context [26, 30], consists in just continuing the learning process without variations and indifferently w.r.t. the changes in the environment. Learning through reinforcement in complex non-stationary environments without any *memory replay* is known to suffer from catastrophic forgetting, instability and convergence difficulties while learning. This strategy is usually considered as a lower bound.

3. ***CRL-Sup***: This approach can be considered as a second baseline in which the distributional shift supervised signal (i.e. when the map changes) is actually provided to the model for memory consolidation purposes. In this case the standard application of EWC with the loss described in eq. 1 is performed but is perfectly synchronized with the end of the training on each map.

4. ***CRL-Static***: In this strategy, the memory is consolidated (i.e. the regularization term added) at fixed steps in time, independently of the changes in the environment. As we will see in the experiments results, this may be very difficult to tune and rather inefficient, depending on the memory consolidation technique used. In fact, when learning from scratch an early and "blind" consolidation of memory may also hurt performance and actually hamper the ability of learning in the future.

4. Experiments and Results

For all the experiments we use a simple batched-A2C with synchronous updates [47], but only *within* the same *map* (the actual environment with fixed static settings), so that when the map changes the model cannot access in any way previous environmental conditions. The architecture of the agent used for these experiments is a plain 3-layers ConvNet (3×3 kernels with 32 feature maps each) with ReLU activations, followed by a fully connected layer encodings the three possible actions A = {*turn-left*, *turn-right* and *move-forward*}[4]. Each training and test episode has a fixed runtime of 1000 ticks. However, the agent is allowed to make an action every 4 frames, maintaining the action chosen based on the first frame fixed for the other three. This allows a smoother interaction with the environment and allows to the agent to not stall in ambiguous situations even if completely stateless (e.g. in front of a wall).

For the batched-A2C implementation, the synchronous gradient update takes place every 20 frames (covering 80 ticks of the total 1000 ticks of the full episode length) and 20 different agents are spawned in parallel in 20 *ViZDoom* instances of the same environment. The discount factor is fixed to $\gamma = 0.99$ for all the environments. More details about the experimental procedure, implementation details and all the hyper-parameters used are available in the section 4.1.

In order to evaluate and compare the performance of each strategy we use the A metric, defined in [11] as an extension of [28]. Performance are evaluated at the end of the training on each map M_i on 300 testing episodes, 100 for each different map M_j, even the ones not already encountered. Given the test cumulative reward matrix $R \in \mathbb{R}^{3\times3}$, which contains in each entry $R_{i,j}$ the *test episodes* average cumulative reward of the model on map M_j after observing the last *training episode* from map M_i; A can be defined as follows:

$$A = \frac{\sum_{i\geq j}^{N} R_{i,j}}{\frac{N(N+1)}{2}} \tag{3}$$

where $N = 3$ and A is essentially the average of the lower triangular matrix of R, which roughly encodes how the model is performing on the current environmental change and the already encountered ones, on average. In Table 2 the average cumulative reward and the A metric results for the *Light*, *Texture*, *Object*, *All* scenarios and the 4 different CRL strategies is reported. It is worth noting that, the scenarios difficulty can vary substantially from a cumulative reward average of ∼200 for the agents trained in the *light* scenario to ∼600 for the *Object* one, which turns out to be the easiest one in our experiments.

By considering the average A metric across all the scenarios for each strategy (at the bottom of Table 2) or in the last column of Fig. 4, it is possible to compare the different strategies independently of the peculiarities of each specific scenario. In this case we can observe how the *CRL-Sup* strategy constitutes, as we would expect, the best approach in terms of absolute A performance. However, the proposed *CRL-Unsup* strategy, while not exploiting any additional supervised signal, reasonably approximates its performance with a gap of ∼50 cumulative reward points. The *CRL-Static* and *CRL-Naive* approaches perform similarly on the A metric, but while the *CRL-Naive* approach is almost consistently better on the last map M_3 at the end of the training, it seems

[4]Input frames with an original resolution of 320×240 are downscaled to 160×120.

Table 2. Average cumulative reward matrix R and A metric result for each scenario and CRL strategy. Results highlighted in **black** and blue represent the best and the second best performing strategies on each scenario. A gray background is used in the cells involved in the computation of the A metric. Results are computed over 10 runs for each strategy and benchmark for a total of 160 runs.

		CRL-Naive			**CRL-Sup**			**CRL-Static**			**CRL-Unsup**		
		M_1	M_2	M_3	M_1	M_2	M_3	M_1	M_2	M_3	M_1	M_2	M_3
Light	M_1	93	-460	144	283	-350	322	242	-613	-36	334	-298	491
	M_2	-987	528	-842	-236	545	642	-987	909	-551	-987	1090	-537
	M_3	-892	1063	938	-232	116	615	-800	832	1106	-892	426	818
	A		123,96			181,97			**217,4**			131,65	
		M_1	M_2	M_3	M_1	M_2	M_3	M_1	M_2	M_3	M_1	M_2	M_3
Texture	M_1	877	544	57	1196	1058	385	1049	822	152	1105	836	72
	M_2	-115	1360	504	-80	1415	867	-6	1150	479	186	1283	631
	M_3	-283	-263	1422	-243	-194	1352	-218	-176	1121	-215	-156	1252
	A		499,81			574,39			486,63			**575,87**	
		M_1	M_2	M_3	M_1	M_2	M_3	M_1	M_2	M_3	M_1	M_2	M_3
Object	M_1	930	-974	-1005	1365	-664	-989	1129	-953	-1006	1308	-695	-995
	M_2	962	1045	-988	1160	1221	-934	781	944	-937	992	1080	-959
	M_3	-758	-214	1013	254	-125	878	-676	-242	845	54	-131	840
	A		496,67			**792,59**			463,7			690,8	
		M_1	M_2	M_3	M_1	M_2	M_3	M_1	M_2	M_3	M_1	M_2	M_3
All	M_1	1268	-1000	-1000	1579	-1000	-1000	1132	-1001	-1000	1518	-998	-1000
	M_2	-490	1346	-991	301	904	-999	-503	1044	-999	-301	1103	-1006
	M_3	-764	-219	815	-389	-370	758	-496	-197	680	-286	-332	695
	A		325,88			**464,04**			276,4			399,59	
Avg. A		361,57 ± 154,18			**503,24** ± 219,96			361,03 ± 116,30			449,47 ± 210,78		

more sensitive to forgetting than the *CRL-Static* approach on previously encountered maps.

Results for each specific scenario roughly confirm this trend with exception of the *light* scenario (the most difficult) where a *CRL-Static* approach seems to prevail even the *CRL-Sup* one. We postulate that, in this case, a more frequent consolidation in the direction of the natural gradient as shown in [42] may help to stabilize learning in complex environments even within the same environmental conditions.

Finally, in Fig. 4, the average A metric for each strategy is reported along with the *Multienv* upper bound. In this case the upper bound is not an A metric but simply the average test cumulative reward (on all the test maps) of a agent trained simultaneously on the three environmental conditions of each scenario. It is worth noting the conspicuous gap w.r.t. the best performing continual reinforcement learning strategy of each scenario, suggesting the need of further research on CL approaches for RL.

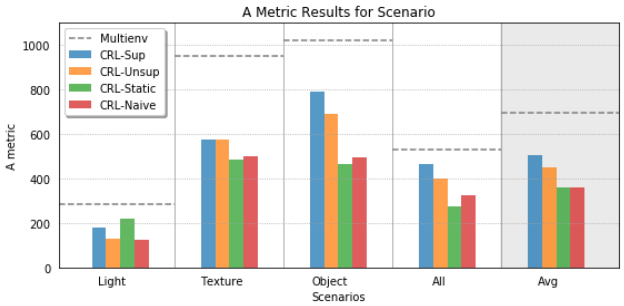

Figure 4. A metric results for each CRL strategy and scenario. Dotted lines indicate the average cumulative reward on 300 test episodes for the *multienv* upper-bound. Better viewed in colors.

4.1. Implementation Details

In this section additional details about the experiments are reported. All the code, environments and setup scripts to reproduce the experiments are openly released at: https://github.com/vlomonaco/crlmaze. In order to properly compare the performance of the proposed CRL strategies a total of 200 runs (10 for each CRL strategy and scenario) has been conducted for more than 40 hours of computation on a single machine with 32 CPU cores and 1 NVIDIA GTX Titan X.

Hyper-parameters used in the experiments are reported instead in Tab. 3. Hyper-parameters have been chosen for each strategy in order to maximize the A metric at the end of each run. *Parallel instances* indicates the number of ViZDoom instances and agents running in parallel for the roll-outs always fixed to 20. *Episode size* is the number of frames (not considering the skip-rate of 4 as explained in section 4) after which a weights update is performed. The r^s_{mavg} and r^l_{mavg} size parameters represent instead the

number of training episodes to consider for the short and long-term moving average, respectively.

Focusing only on the strategies employing consolidation η, α and λ are the parameters already described in section 4 while *Fisher freq.*, *Fisher clip* and *Fisher sample size* represent respectively *i)* the computing frequency of the fisher matrix in terms of training episodes, *ii)* the clipping value of the importance magnitude as described in [30], and *iii)* the number of episodes used to estimate the fisher information of each weight.

For more details about the experiments please refer to the extended preprint [25].

5. Discussion and Conclusions

In this work we introduced and openly released a new environment and benchmark for easily assessing continual reinforcement learning algorithms on a complex 3D non-stationary environment. The preliminary experiments introduced in section 4 on four different scenarios and 5 different strategies show that the proposed unsupervised approach without any distributional shift supervised signal, external model or distribution-specific over-parametrization is not only possible but may be competitive with respect to a standard *supervised* counterpart.

However, as we observed in some experiments (where there is a noticeable gap between the *CRL-Sup* and *CRL-Unsup* strategies), the detection of a timely consolidation signal can be sometimes critical. For example, in case of *positive forward transfer* followed by a possible *negative backward transfer* [28] (i.e. being able to perform well on new conditions but impacting negatively on learned knowledge about the previous ones) memory consolidation can not take place just by looking at the training cumulative reward curve since steadily growing. This problem may be tackled by looking at an additional regularization loss for reconstructing the input frame (or predicting the next one) since changes in the input space may be more evident. In this way, while still using a single end-to-end model and constructing more robust features [15, 22], it would be possible to integrate the benefit of both approaches when learning continuously.

In the future we plan to expand this work in several other directions. Firstly by moving towards a more flexible and more principled solution where the consolidation is proportional to the expected reward difference encoded directly in the loss function. Secondly by integrating more accurate synaptic plasticity models as shown in [19, 5] and going beyond mere consolidation processes which tend to quickly saturate the model learning capacity.

Finally, we plan to extend our evaluation where existing environmental changes are discretized by providing additional training maps for each category and by adding a new environmental change category where the size of the maze is substantially varied.

Table 3. Specific hyper-parameters used for each strategy and scenario.

		Naive	**Sup**	**Static**	**Unsup**	**Multienv**
Light	Parallel instances	20	20	20	20	20
	Learning rate	6e-5	9e-5	9e-5	9e-5	6e-5
	Discount factor	0.99	0.99	0.99	0.99	0.99
	Episode Size	20	20	20	20	20
	Train episodes	300	300	300	300	600
	Test episodes	100	100	100	100	100
	r^l_{mavg} size	n.d.	n.d.	n.d.	50	n.d.
	r^s_{mavg} size	n.d.	n.d.	n.d.	6	n.d.
	η	n.d.	n.d.	n.d.	-80	n.d.
	α	n.d.	n.d.	n.d.	10e7	n.d.
	λ	n.d.	10e7	10e5	n.d.	n.d.
	Fisher freq.	n.d.	300	100	100	n.d.
	Fisher clip	n.d	10e-7	10e-7	10e-7	n.d.
	Fisher sample size	n.d	100	100	100	n.d.
Texture	Parallel Instances	20	20	20	20	20
	Learning rate	9e-5	2e-4	2e-4	2e-4	9e-5
	Discount factor	0.99	0.99	0.99	0.99	0.99
	Episode Size	20	20	20	20	20
	Train episodes	300	300	300	300	600
	Test episodes	100	100	100	100	100
	r^l_{mavg} size	n.d.	n.d.	n.d.	50	n.d.
	r^s_{mavg} size	n.d.	n.d.	n.d.	6	n.d.
	η	n.d.	n.d.	n.d.	-50	n.d.
	α	n.d.	n.d.	n.d.	5e6	n.d.
	λ	n.d.	5e6	5e6	n.d.	n.d.
	Fisher freq.	n.d.	300	100	100	n.d.
	Fisher clip	n.d	10e-7	10e-7	10e-7	n.d.
	Fisher sample size	n.d	60	60	60	n.d.
Object	Parallel Instances	20	20	20	20	20
	Learning rate	9e-5	2e-4	2e-4	2e-4	2e-4
	Discount factor	0.99	0.99	0.99	0.99	0.99
	Episode Size	20	20	20	20	20
	Train episodes	500	500	500	500	2600
	Test episodes	100	100	100	100	100
	r^l_{mavg} size	n.d.	n.d.	n.d.	50	n.d.
	r^s_{mavg} size	n.d.	n.d.	n.d.	6	n.d.
	η	n.d.	n.d.	n.d.	-60	n.d.
	α	n.d.	n.d.	n.d.	3e6	n.d.
	λ	n.d.	3e6	3e6	n.d.	n.d.
	Fisher freq.	n.d.	500	100	100	n.d.
	Fisher clip	n.d	10e-7	10e-7	10e-7	n.d.
	Fisher sample size	n.d	60	60	60	n.d.
All	Parallel Instances	20	20	20	20	20
	Learning rate	9e-5	2e-4	2e-4	2e-4	2e-4
	Discount factor	0.99	0.99	0.99	0.99	0.99
	Episode Size	40	40	40	40	40
	Train episodes	500	500	500	500	2600
	Test episodes	100	100	100	100	100
	r^l_{mavg} size	n.d.	n.d.	n.d.	50	n.d.
	r^s_{mavg} size	n.d.	n.d.	n.d.	6	n.d.
	η	n.d.	n.d.	n.d.	-100	n.d.
	α	n.d.	n.d.	n.d.	1e6	n.d.
	λ	n.d.	7e6	3e6	n.d.	n.d.
	Fisher freq.	n.d.	500	166	166	n.d.
	Fisher clip	n.d	10e-7	10e-7	10e-7	n.d.
	Fisher sample size	n.d	60	60	60	n.d.

While still in their infancy we can foresee a new generation of reinforcement learning algorithms which can learn continually in complex non-stationary environments, opening the door to artificial learning agents which can autonomously acquire new knowledge and skills in unpredictable, real-word settings.

References

[1] Maruan Al-shedivat, Trapit Bansal, Yuri Burda, Ilya Sutskever, Igor Mordatch, and Pieter Abbeel. Continuous Adaptation via Meta-Learning in Nonstationary and Competitive Environments. In *International Conference on Learning Representations (ICLR)*, pages 1–21, 2018. 2, 4

[2] Rahaf Aljundi, Klaas Kelchtermans, and Tinne Tuytelaars. Task-Free Continual Learning. *arXiv preprint arXiv:1812.03596*, pages 1–14, 2018. 2

[3] Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. Deep Reinforcement Learning: A Brief Survey. *IEEE Signal Processing Magazine*, 34(6):26–38, nov 2017. 1

[4] Charles Beattie, Joel Z Leibo, Denis Teplyashin, Tom Ward, Marcus Wainwright, Andrew Lefrancq, Simon Green, Amir Sadik, Julian Schrittwieser, Keith Anderson, Sarah York, Max Cant, Adam Cain, Adrian Bolton, Stephen Gaffney, Helen King, Demis Hassabis, Shane Legg, and Stig Petersen. DeepMind Lab. pages 1–11, 2016. 2

[5] Marcus K Benna and Stefano Fusi. Computational principles of synaptic memory consolidation. *Nature Neuroscience*, 19(12):1697–1706, dec 2016. 8

[6] Zhiyuan Chen and Bing Liu. *Lifelong Machine Learning*. Morgan & Claypool Publishers, nov 2018. 2, 4

[7] Maxime Chevalier-Boisvert, Dzmitry Bahdanau, Salem Lahlou, Lucas Willems, Chitwan Saharia, Huu Nguyen Thien, and Yoshua Bengio. BabyAI: A Platform to Study the Sample Efficiency of Grounded Language Learning. *7th International Conference on Learning Representations, (ICLR)*, pages 1–18, 2019. 4

[8] Claudia Clopath. Synaptic consolidation: An approach to long-term learning. *Cognitive Neurodynamics*, 6(3):251–257, 2012. 4

[9] Claudia Clopath, Lorric Ziegler, Eleni Vasilaki, Lars Büsing, and Wulfram Gerstner. Tag-trigger-consolidation: A model of early and late long-term-potentiation and depression. *PLoS Computational Biology*, 4(12), 2008. 4

[10] Karl Cobbe, Oleg Klimov, Chris Hesse, Taehoon Kim, and John Schulman. Quantifying Generalization in Reinforcement Learning. 2018. 4

[11] Natalia Díaz-Rodríguez, Vincenzo Lomonaco, David Filliat, and Davide Maltoni. Don't forget, there is more than forgetting: new metrics for Continual Learning. *Continual Learning Workshop at NIPS*, 2018. 2, 6

[12] Chelsea Finn, Aravind Rajeswaran, Sham Kakade, and Sergey Levine. Online Meta-Learning. In *International Conference on Learning Representations (ICLR)*, pages 1–14, 2019. 4

[13] Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999. 2

[14] Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An Empirical Investigation of Catastrophic Forgeting in Gradient-Based Neural Networks. *arXiv preprint arXiv:1312.6211*, 2013. 4

[15] Karl Moritz Hermann, Felix Hill, Simon Green, Fumin Wang, Ryan Faulkner, Hubert Soyer, David Szepesvari, Wojciech Marian Czarnecki, Max Jaderberg, Denis Teplyashin, Marcus Wainwright, Chris AparXiv preprint arXiv:1706.06551ps, and Demis Hassabis. Grounded Language Learning in a Simulated 3D World. *arXiv preprint arXiv:1706.06551*, 2017. 8

[16] David Isele and Akansel Cosgun. Selective Experience Replay for Lifelong Learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*, pages 3302–3309, 2018. 1

[17] Matthew Johnson, Katja Hofmann, Tim Hutton, and David Bignell. The malmo platform for artificial intelligence experimentation. In *IJCAI International Joint Conference on Artificial Intelligence*, volume 2016-Janua, pages 4246–4247, 2016. 2

[18] Ken Kansky, Tom Silver, Eldawy Miguel, and Xinghua Lou. Schema Networks: Zero-shot Transfer with a Generative Causal Model of Intuitive Physics. *International Conference on Machine Learning*, 2017. 3

[19] Christos Kaplanis, Murray Shanahan, and Claudia Clopath. Continual Reinforcement Learning with Complex Synapses. In *Proceedings of the 35th International Conference on Machine Learning*, pages 2497—-2506, 2018. 8

[20] Michal Kempka, Marek Wydmuch, Grzegorz Runc, Jakub Toczek, and Wojciech Jaskowski. ViZDoom: A Doom-based AI research platform for visual reinforcement learning. In *2016 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–8. IEEE, sep 2016. 2

[21] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. 114(13):3521–3526, 2017. 2, 4, 5

[22] Timothee Lesort, Natalia Diaz-Rodriguez, Jean-Franois Goudou, and David Filliat. State representation learning for control: An overview. *Neural Networks*, 108:379–392, 2018. 8

[23] Ruishan Liu and James Zou. The Effects of Memory Replay in Reinforcement Learning. *2018 56th Annual Allerton Conference on Communication, Control, and Computing, Allerton 2018*, pages 478–485, 2019. 1

[24] Vincenzo Lomonaco. *Continual Learning with Deep Architectures*. Phd thesis, University of Bologna, 2019. 2

[25] Vincenzo Lomonaco, Karan Desai, Eugenio Culurciello, and Davide Maltoni. Continual Reinforcement Learning in 3D Non-stationary Environments. *Arxiv pre-print arXiv:1905.10112v1*, 2019. 8

[26] Vincenzo Lomonaco and Davide Maltoni. CORe50: a New Dataset and Benchmark for Continuous Object Recognition. In Sergey Levine, Vincent Vanhoucke, and Ken Goldberg, editors, *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pages 17–26. PMLR, 2017. 6

[27] Vincenzo Lomonaco, Davide Maltoni, and Lorenzo Pellegrini. Fine-Grained Continual Learning. pages 1–14, 2019. 5

[28] David Lopez-paz and Marc'Aurelio Ranzato. Gradient Episodic Memory for Continuum Learning. In *Advances in neural information processing systems (NIPS 2017)*, 2017. 6, 8

[29] Marlos C. Machado, Marc G. Bellemare, Erik Talvitie, Joel Veness, Matthew Hausknecht, and Michael Bowling. Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. *IJCAI International Joint Conference on Artificial Intelligence*, 2018-July:5573–5577, 2018. 4

[30] Davide Maltoni and Vincenzo Lomonaco. Continuous learning in single-incremental-task scenarios. *Neural Networks*, 116:56–73, aug 2019. 2, 5, 6, 8

[31] Michael McCloskey and Neal J. Cohen. Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. *Psychology of Learning and Motivation - Advances in Research and Theory*, 24(C):109–165, 1989. 2

[32] Martial Mermillod, Aurélia Bugaiska, and Patrick Bonin. The stability-plasticity dilemma: investigating the continuum from catastrophic forgetting to age-limited learning effects. *Frontiers in psychology*, 4(August):504, 2013. 4

[33] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. *American Journal of Health Behavior*, pages 1928—-1937, 2016. 1, 5

[34] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–33, 2015. 1

[35] Anusha Nagabandi, Ignasi Clavera, Simin Liu, Ronald S. Fearing, Pieter Abbeel, Sergey Levine, and Chelsea Finn. Learning to Adapt in Dynamic, Real-World Environments Through Meta-Reinforcement Learning. pages 1–17, mar 2018. 2, 4

[36] German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, may 2019. 2, 4, 6

[37] German I. Parisi and Vincenzo Lomonaco. *Online Continual Learning on Sequences*, pages 197–221. Springer International Publishing, Cham, 2020. 4

[38] Xingchao Peng, Baochen Sun, Karim Ali, and Kate Saenko. Learning Deep Object Detectors from 3D Models. In *2015 IEEE International Conference on Computer Vision (ICCV)*, volume 2015 Inter, pages 1278–1286. IEEE, dec 2015. 3

[39] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Continual Learning by Maximizing Transfer and Minimizing Interference. (NeurIPS):1–24, 2018. 4

[40] Mark Ring. *Continual Learning in Reinforcement Environments*. PhD thesis, 1994. 2

[41] Anthony Robins. Catastrophic Forgetting, Rehearsal and Pseudorehearsal. *Connection Science*, 7(2):123–146, 1995. 2

[42] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 5, 7

[43] Jonathan Schwarz, Jelena Luketina, Wojciech M. Czarnecki, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & Compress: A scalable framework for continual learning. In *Proceedings of the 35th International Conference on Machine Learning*, pages 4528—-4537, 2018. 4, 5

[44] Seyed Hadi Mir Yazdi and Ziba Habibi Lashkari. Technical analysis of Forex by MACD Indicator. *International Journal of Humanities and Management Sciences (IJHMS)*, 1(1998):159–165, 2013. 4

[45] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354–359, 2017. 1

[46] Bradly C. Stadie, Ge Yang, Rein Houthooft, Xi Chen, Yan Duan, Yuhuai Wu, Pieter Abbeel, and Ilya Sutskever. Some Considerations on Learning to Explore via Meta-Reinforcement Learning. (NeurIPS), 2018. 4

[47] Adam Stooke and Pieter Abbeel. Accelerated Methods for Deep Reinforcement Learning. *arXiv preprint arXiv:1803.02811*, pages 1–16, 2018. 6

[48] Sebastian Thrun and Tom M Mitchell. Lifelong Robot Learning. *The biology and technology of intelligent autonomous agents*, pages 165—-196, 1995. 2

[49] Alexander Sasha Vezhnevets, Simon Osindero, Tom Schaul, Nicolas Heess, Max Jaderberg, David Silver, and Koray Kavukcuoglu. FeUdal Networks for Hierarchical Reinforcement Learning. 2016. 4

[50] Bohan Wu. Model Primitive Hierarchical Lifelong Reinforcement Learning. 2018. 4

[51] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual Learning Through Synaptic Intelligence. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 3987–3995, Sydney, Australia, 2017. 5