Grid World Transfer Learning - from Small to Big Challenges

Florian Fingscheidt, Merle Krauss, Mathis Pöhlsen
https://github.com/F10rian/RL_Project.git

Abstract

We investigate how transfer learning impacts the convergence speed of Deep Q-Networks (DQNs) when fine-tuning across grid-world environments of varying sizes. By pretraining agents on a small 5×5 grid and transferring to larger 7×7 and 9×9 grids, we compare transfer learning against training from scratch. Our results show that transfer learning accelerates early training, with the greatest benefits observed when transferring to substantially larger environments. Furthermore, incorporating curriculum learning with an intermediate environment further improves learning speed and final performance. These findings demonstrate that transfer and curriculum learning are effective strategies to enhance sample efficiency and performance of DQNs in grid-world navigation tasks.

1 Introduction

Deep Reinforcement Learning (DRL) agents such as Deep Q-Networks (DQNs) require long training time to reach good performance. Here, transfer learning has good potential to accelerate training Zhu et al. [2023]. We want to investigate how transfer learning affects convergence speed when fine-tuning DQNs across grid-world environments of varying sizes.

Therefore, we pretrained a DQN agent in a simple grid environment and experimented with transfer and curriculum learning to larger environments. We could observe that the transferred DQN agent reached good performance with significantly fewer training steps than the baseline trained from scratch in most cases. Furthermore, the agent trained with curriculum learning could outperform both the baseline and the agent that was transferred directly from the baseline.

2 Related Work

We build upon the work of de la Cruz et al. [2016], who investigated how transfer learning can accelerate DRL by reusing policies across related tasks. Building on the insights of Chevalier-Boisvert et al. [2023], who applied transfer learning to PPO agents in grid-world navigation, we instead focus on DQNs, which are particularly susceptible to slow training and therefore yield considerable potential for improvement through faster learning, to investigate whether the performance gains observed with PPO also hold for algorithms with different learning dynamics.

3 Approach

For our grid environments, we used Gym's MiniGrid Chevalier-Boisvert et al. [2023], which is a reinforcement learning environment suite designed for research on navigation tasks in discrete grid-worlds. We use a 5×5 grid environment to pretrain our agent to navigate to a defined goal in an empty grid, and subsequently transfer the learned policy to larger grids of 7×7 (medium) and 9×9 (large) sizes to evaluate generalization capabilities. All environments are used in a fully observable

setting by applying the FullyObsWrapper from the MiniGrid package, which provides the agent with a complete, unbounded view of the environment state. We further extend this wrapper so that, in addition to the agent's position, the observation also encodes the agent's facing direction, ensuring the policy is aware of orientation-dependent actions. Finally, we apply ImgObsWrapper to remove unnecessary information from the observations.

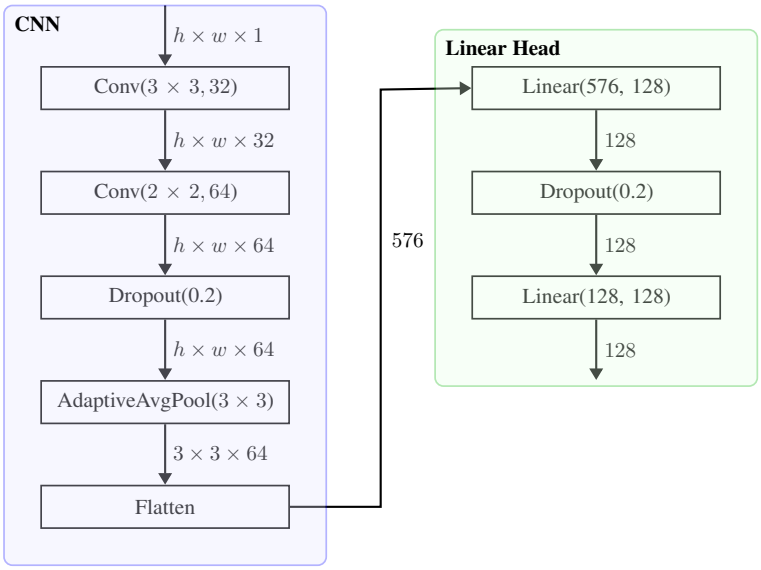

Figure 1: Custom CNN-based feature extractor architecture used as the DQN's input processing module.

We used the DQN implementation from Stable Baselines3 Raffin et al. [2021] as our reinforcement learning agent. Stable Baselines3 is a widely used library offering implementations of state-of-the-art RL algorithms. For our agent, we use the built-in CnnPolicy but replace its default feature extractor with a custom designed neural network tailored specifically for processing the image-based observations from our MiniGrid environments. Our custom feature extractor shown in Figure 1 consists of two key components: a convolutional neural network (CNN) for spatial feature extraction from input images, followed by a linear projection head that maps these features into a compact latent representation optimized for the DQN's decision-making process.

The CNN consists of two convolutional layers, where each is followed by ReLU activation functions. To mitigate overfitting, a 2D dropout layer is applied after the convolutional blocks. Next, an adaptive average pooling layer resizes the feature maps to a fixed spatial dimension of 3×3, ensuring a consistent output size regardless of input variability. This is crucial to later transfer to different grid sizes. The pooled features are then flattened and passed through a linear head consisting of a fully connected linear layer with ReLU activation, followed by another dropout layer for regularization. Finally, a linear layer projects the output to the desired feature dimension. This architecture is designed to extract meaningful spatial features while maintaining regularization to improve generalization when transferring to different environments.

4 Experiments

Our experiments aim to study the effects of transferring a DQN pretrained in a small source environment to larger target environments. We therefore compare our transferred DQN to a baseline DQN trained from scratch in the target environment in order to assess whether transfer learning enables the agent to achieve superior performance more rapidly. To evaluate training progress, we track the running maximum of the mean reward per episode. This choice reflects our focus on the peak performance an agent achieves and the point in training when that peak occurs, without considering any subsequent decline in performance. This corresponds to a real-world scenario, where we are usually also only interested in the best model after training. Performance is assessed using three metrics: 1) the area under the curve (AUC) of the mean reward over episodes, which captures the

overall reward accumulation during training; 2) the maximum mean reward per episode, representing the final peak performance attained; and 3) the number of steps required for the transferred agent to reach the baseline's maximum mean reward per episode, indicating whether the transferred agent learns faster than the baseline. Together, these measures provide a comprehensive view of both the performance and the learning speed of the transferred DQN compared to training from scratch.

For each experiment, training is performed for a total of 100,000 steps. To ensure statistical significance, we repeat each training run with 20 different random seeds. In the baseline setting, this means training 20 independent DQN agents from scratch in the target environment. In the fine-tuning setting, we first pretrain 20 DQN agents on the source environment and then transfer each pretrained model to the target environment, where it is fine-tuned for an additional 100,000 steps. All training runs use the same hyperparameters to maintain comparability. The only exceptions occur in the fine-tuning phase, where both the learning rate and the exploration rate are slightly reduced compared to pretraining or baseline training. This adjustment mitigates the risk of overwriting useful knowledge acquired during pretraining and prevents excessive exploration from destabilizing a policy that is already partially adapted to the task. By lowering these parameters, we aim to allow the fine-tuned agent to refine its existing policy more efficiently rather than relearning from scratch.

We examined two experimental scenarios to evaluate the effect of transfer learning in a medium and a large environment. In the first scenario, we investigated the transfer from the small 5×5 grid environment to the medium 7×7 grid. We therefore trained a baseline on the 7×7 grid as well as a transfer agent which was pretrained on the 5×5 environment and then fine-tuned on the 7×7 environment.

In the second scenario, we examined the transfer from the 5×5 grid-world to the large 9×9 grid-world. Here, the baseline involved direct training in the 9×9 environment, while the transfer condition involved pretraining in the 5×5 environment followed by fine-tuning in the 9×9 environment. Here, we also explored a curriculum learning approach in which agents were pretrained in the small 5×5 environment, transferred to the medium 7×7 environment for intermediate training and subsequently transferred again to the large 9×9 environment for final fine-tuning. This design allowed us to test whether a gradual increase in environmental complexity yields additional performance benefits compared to direct transfer from the smallest to the largest environment.

4.1 Results for Transfer Learning from 5×5 to 7×7 environment

For our experiment of transferring from 5×5 to 7×7 we observe several notable trends. Figure 2 shows the average performance across 20 training runs with different random seeds, including a 95% confidence band representing the interquartile range.

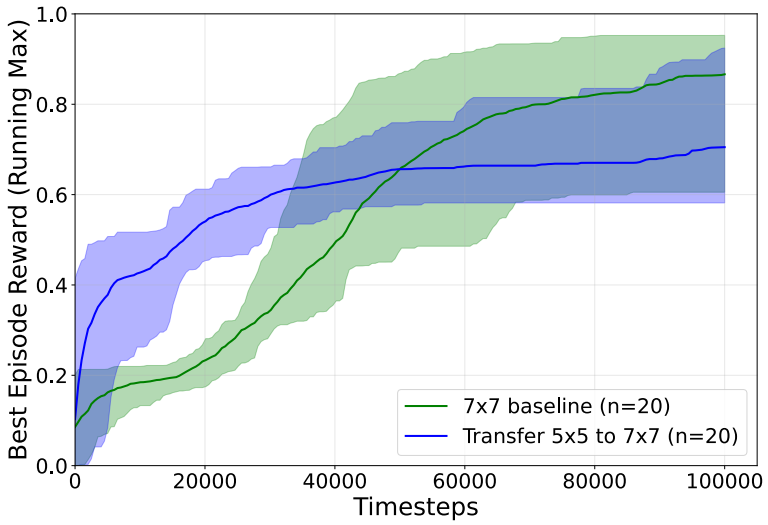

Figure 2: Transfer from 5×5 to 7×7 grid environment.

Compared to the 7×7 baseline trained from scratch, the transferred agent demonstrates a clear head start, achieving a significantly higher average reward from the beginning of training. This advantage is particularly evident until approximately 50,000 steps, during which the mean reward of the transferred agent consistently exceeds that of the baseline. Between 5,000 and 30,000 steps, the entire 95% confidence band of the transferred runs lies above the baseline's, indicating that the improvement is statistically significant during this phase.

After 50,000 steps, however, the baseline surpasses the transferred agent in average performance and maintains a higher reward level until the end of training. This pattern suggests that, while transfer learning offers a faster initial learning speed, the baseline ultimately achieves superior peak performance in this relatively small environment. One possible explanation is that the modest increase in environment size from 5×5 to 7×7 limits the benefit of transfer, as the target environment remains close in complexity to the source.

Table 1 summarizes key performance metrics of the experiment. The baseline reaches a higher average maximum reward of 0.87 at step 99,127, while the transferred agent achieves a lower average peak reward of 0.60 and does not reach the baseline's peak within the training horizon. Interestingly, the transferred agent has a higher mean area under the curve (AUC) of 59,156 compared to 55,352 for the baseline, reflecting its stronger early performance.

Together, these results demonstrate that transfer learning from 5×5 to 7×7 provides a valuable initial boost in learning speed and early reward accumulation but may not improve final peak performance when the target environment is only marginally larger than the source.

Model	**Max avg. reward**	**Mean AUC reward**	**Max avg. reward @ step**
Baseline 7x7	**0.87**	55352	0.87 @ step **99127**
Transfer (5x5 $\rightarrow$ 7x7)	0.60	**59156**	0.87 never reached

Table 1: Performance comparison of transfer from 5×5 to 7×7 grid environment.

4.2 Transfer and curriculum learning from 5×5 to 9×9 environment

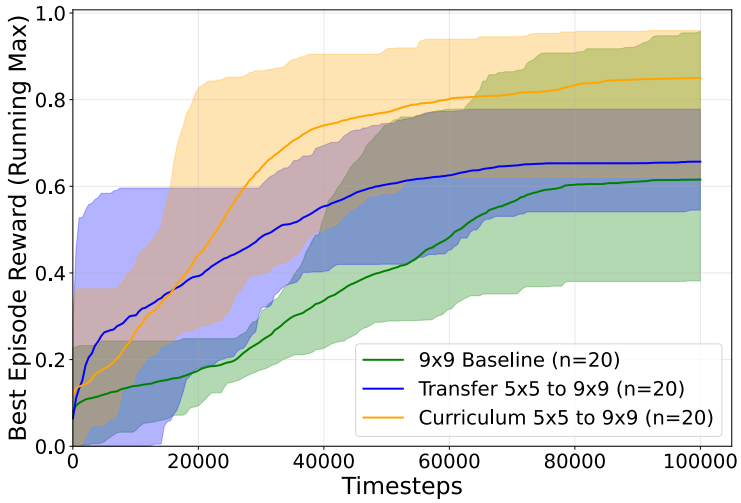

Figure 3: Transfer from 5×5 to 9×9 grid environment with both direct transfer and curriculum transfer with intermediate step to 7×7.

The experiments comparing direct transfer learning from the 5×5 to the 9×9 environment and curriculum transfer learning, where training progresses from 5×5 to 7×7 and then to 9×9, reveal clear advantages over training a baseline agent from scratch in the 9×9 environment. Figure 3 again

shows the average performance over 20 independent training runs, each with a different random seed, along with a 95% confidence bands that highlight statistical reliability.

Compared to the 9×9 baseline trained from scratch, both transfer variants exhibit a clear initial advantage, achieving higher mean rewards from the earliest stages of training on. This improvement is particularly evident in the curriculum transfer setting, which begins to pull ahead of direct transfer at approximately 18,000 steps and maintains the highest average performance throughout the training. Between 18,000 and 39,000 steps, the 95% confidence bands of the curriculum runs lies fully above that of the baseline, indicating a clear statistically significant advantage in this interval. While the confidence band of direct transfer does not remain consistently above the baseline, it still shows a marked head start, converging to higher performance more quickly than the baseline. Notably, both transfer and curriculum methods achieve higher maximum average rewards than the baseline, demonstrating the positive influence of prior knowledge on learning speed and eventual performance.

The statistical significance of these differences is further supported by a Mann–Whitney U-test conducted at a 95% significance level. For the comparison between the mean episode rewards of the baseline and those of the direct transfer we obtain $U = 64.00$, $p = 2 \times 10^{-4}$, indicating a statistically significant difference in performance distributions. For the comparison between the baseline and curriculum transfer we get $U = 15.00$, $p = 6 \times 10^{-7}$, again confirming a highly significant difference in favor of the curriculum approach.

These dynamics are further supported by the quantitative metrics summarized in Table 2. The curriculum transfer agent achieves the highest maximum average reward of 0.85, substantially outperforming both the baseline's peak of 0.61 and the direct transfer's 0.66. Additionally, the curriculum approach attains this peak at an earlier training step of 28,991, compared to 54,235 steps for the direct transfer and 96,641 steps for the baseline. The mean area under the reward curve (AUC) also reflects these trends, with curriculum transfer leading at 64,466, followed by direct transfer at 51,634, and baseline at 37,907. This demonstrates that curriculum transfer not only accelerates learning but also yields higher cumulative rewards throughout the 100,000 training steps.

Overall, these results indicate that when transferring to a substantially larger environment, prior training in a smaller environment accelerates early learning and improves peak performance. Furthermore, incorporating an intermediate curriculum stage amplifies these benefits, leading to faster performance improvements and higher final rewards.

Model	**Max avg. reward**	**Mean AUC reward**	**Max avg. reward @ step**
Baseline 9x9	0.61	37,907	0.61 @ step 96,641
Transfer (5x5 $\rightarrow$ 9x9)	0.66	51,634	0.61 @ step 54,235
Curriculum (5x5 $\rightarrow$ 9x9)	**0.85**	**64,466**	0.61 @ step **28,991**

Table 2: Performance comparison of direct transfer and curriculum transfer (with intermediate step on 7×7) from 5×5 to 9×9 grid environment.

5 Discussion

In this work, we investigated how transfer learning influences the convergence speed of Deep Q-Networks (DQNs) when fine-tuning across grid-world environments of varying sizes. Specifically, we pretrained DQNs on a small 5×5 grid environment and examined their performance when transferred and fine-tuned on larger grids of size 7×7 and 9×9. Additionally, we explored curriculum learning by introducing an intermediate training step on the medium-sized environment before final fine-tuning on the largest grid.

Our results show that transfer learning consistently speeds up early learning. For the 5×5 to 7×7 transfer, the pretrained agent started stronger but was eventually outperformed by the baseline, indicating that transfer offers faster learning but may limit peak performance when the target environment differs only slightly from the source. For the larger jump to the 9×9 environment, both direct transfer and curriculum learning improved convergence speed and final performance, with curriculum learning achieving the best results. This confirms that transfer learning is more beneficial when the

target environment is substantially more complex and that gradually increasing difficulty further boosts performance.

In summary, we answer our research question positively: transfer learning accelerates convergence speed when fine-tuning DQNs across grid sizes, especially with larger environment differences and when combined with curriculum learning.

Future work could explore other RL algorithms, more complex environments with advanced tasks or obstacles, and advanced transfer methods to enhance fine-tuning efficiency and generalization. Additionally, analyzing what representations are transferred and how freezing specific layers affects the transfer could deepen our understanding of transfer learning in DRL.

References

Maxime Chevalier-Boisvert, Bolun Dai, Mark Towers, Rodrigo Perez-Vicente, Lucas Willems, Salem Lahlou, Suman Pal, Pablo Samuel Castro, and Jordan Terry. Minigrid & miniworld: Modular & customizable reinforcement learning environments for goal-oriented tasks. In *Advances in Neural Information Processing Systems 36, New Orleans, LA, USA*, December 2023.

Maxime Chevalier-Boisvert, Bolun Dai, Mark Towers, Rodrigo De Lazcano Perez-Vicente, Lucas Willems, Salem Lahlou, Suman Pal, Pablo Samuel Castro, and J K Terry. Minigrid & miniworld: Modular & customizable reinforcement learning environments for goal-oriented tasks. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=PFfmfspm28.

Gabriel de la Cruz, Yunshu Du, James Irwin, and Matthew Taylor. Initial progress in transfer for deep reinforcement learning algorithms. 07 2016.

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021. URL http://jmlr.org/papers/v22/20-1364.html.

Zhuangdi Zhu, Kaixiang Lin, Anil K. Jain, and Jiayu Zhou. Transfer learning in deep reinforcement learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(11):13344–13362, November 2023. ISSN 0162-8828. doi: 10.1109/TPAMI.2023.3292075. URL https://doi.org/10.1109/TPAMI.2023.3292075.