Self-Supervised Attention-Aware Reinforcement Learning

Haiping Wu **Khimya Khetarpal** **Doina Precup**
McGill University Mila
{haiping.wu2, khimya.khetarpal}@mail.mcgill.ca dprecup@cs.mcgill.ca

Abstract

Visual saliency has emerged as a key visualization technique for interpreting deep reinforcement learning (RL) agents. Contrary to much of the existing research that uses saliency as an analyzing tool, in this work, we use visual attention as an inductive bias for decision making in RL agents. We propose a novel self-supervised attention learning approach which can 1. learn to select regions of interest without explicit annotations, and 2. act as a plug for existing deep RL methods to improve the learning performance. We empirically show that the self-supervised attention-aware deep RL methods outperform the baselines in the context of both the rate of convergence and performance. Furthermore, the proposed self-supervised attention is not tied to specific policies, nor restricted to a specific scene. We posit that our approach is a universal self-supervised attention module for multi-task learning and transfer learning, and empirically validate the generalization ability of the learned attention. Finally, we can also extract object keypoints as a byproduct of the learned attention. We demonstrate qualitatively that the extracted object keypoints are superior to existing methods.

Introduction

In recent years, deep reinforcement learning (RL) methods [13, 14, 12] have achieved great success in large part driven by the revolution in convolution neural networks (CNN) and feed-forward networks as function approximators. However, it is yet unknown how these deep RL agents understand scenes and make decisions. We are interested in building RL agents which learn representations guided by an understanding of what is important in a scene for sequential decision making.

Object-oriented representation is a long-standing approach to understanding and simplifying a scene [3, 6, 8, 5, 10]. Recent works [16, 7, 9, 11, 4] try to obtain object keypoints in an unsupervised manner. However, current unsupervised keypoints detection methods including the Transporter [9] are limited in that they do not deal with variable number of objects, scale, and classes of objects. Furthermore, the use of object-oriented representation for deep RL has not been highly explored.

We here argue that attention masks are a better technique to help the learning of policies given current tools. Attention masks aim to find salient areas in a scene and account for any number of regions in that they are not restricted to specific object categories or count. More importantly, since it has the same form as that of the original image (i.e. a map of the attention values v.s. a map of pixel values), it is straightforward to plug in any existing deep RL methods for decision making. Inspired by Transporter [9], we propose a self-supervised attention module which is designed for learning attention masks instead of object keypoints. The module is an auto-encoder like architecture with a bottleneck in attention masks that it needs to correctly identify the regions of interest to perform the image reconstruction. The overall pipeline is shown in Figure 1. The learned attention masks are class-agnostic. Moreover, they account for various shapes, number of objects, as opposed to pre-defined number of objects during training [9]. Our **contributions** are as follows: 1) We design a

NeurIPS 2020 Workshop on Object Representations for Learning and Reasoning, Vancouver, Canada.

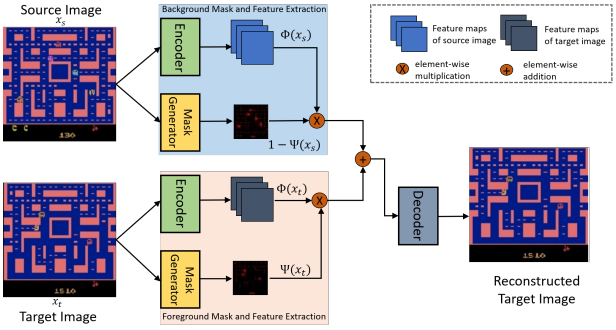

Figure 1: **Proposed self-supervised attention module pipeline.** The core idea is to employ a self-supervised loss through an auto-encoder architecture with a bottleneck. The module tries to reconstruct the target image x_t by using minimal information (features of foreground regions) from the target image x_t, and other needed information from source image x_s.

self-supervised attention mask module that learns general-purpose attention masks through a novel self-supervised loss. 2) We incorporate the self-supervised attention mask module with deep RL methods, and empirically show a gains in both the convergence speed and final scores in single task learning. 3) We empirically demonstrate that the attention masks learned via our self-supervised approach results in better generalization capabilities in both transfer and multi-task learning settings. 4) We show that the extracted object keypoints from our masks performs qualitatively better than the Transporter model which further highlights the flexibility of our method.

Self-Supervised Attention for Reinforcement Learning

Method: Self-Supervised Attention Module

Our aim is to learn a mask that indicates the potential of each location in the visual input being the region of interest. Hereafter, we refer to the region of interest as the foreground, and background otherwise. Inspired by the Transporter [9] model, we design a bottleneck architecture to reconstruct images, which could ideally differentiate between the interested foreground and background, in a self-supervised manner. The overall architecture is shown in Figure 1.

Given a source frame x_s and a target frame x_t, randomly sampled from one game-play, we design the self-supervised learning task as reconstructing the target frame x_t from the source frame x_s. We use auto-encoder with bottleneck to construct x_t. First, the encoder extracts features of x_s and x_t as $\Phi(x_s), \Phi(x_t) \in \mathbb{R}^{H' \times W' \times D}$ respectively. The mask generator outputs the mask maps of x_s and x_t as $\Psi(x_s), \Psi(x_t) \in [0,1]^{H' \times W'}$, indicating the probability of being interested for the corresponding feature map location. The feature used for reconstructing x_t is then calculated as follows: $\hat{\Phi}\left(\boldsymbol{x}_s, \boldsymbol{x}_t\right) \triangleq \left(1 - \Psi\left(\boldsymbol{x}_s\right)\right) \cdot \left(1 - \Psi\left(\boldsymbol{x}_t\right)\right) \cdot \Phi\left(x_s\right) + \Psi\left(\boldsymbol{x}_t\right) \cdot \Phi\left(x_t\right)$. Finally, besides the original auto-encoder pipeline that the decoder reconstructs $\hat{x_t}^{auto}$ from features $\Psi(x_t)$, the decoder also takes in the features $\hat{\Phi}\left(\boldsymbol{x}_s, \boldsymbol{x}_t\right)$ and outputs the reconstructed $\hat{x_t}$.

Ideally, we want the decoder to use the features that combine the background features from x_s and foreground features from x_t to reconstruct x_t. However, directly optimizing the reconstruction loss between x_t and $\hat{x_t}$ would give a trivial solution for masks that $\Psi(x_t) = 1$, which is not in our interest. Therefore, we propose to add a penalty term for the masks that leads to minimize the locations that are identified as regions of interest. We can also interpret this penalty term acts as a sparsity regularizer. The overall loss for training the self-supervised attention mask is defined as follows: $\mathcal{L}_{mask} = \|\hat{x_t} - x_t\|_{2*}^2 + \|\hat{x_t}^{\texttt{auto}} - x_t\|_2^2 + \lambda_m \|\Psi(x_t)\|_1$. where $\|\cdot\|_{2*}^2$ is squared-ℓ_2 norm with threshold δ, that ignores terms that have a squared value less than δ. It is defined as $\|\boldsymbol{y}\|_{2*}^2 = \sum_k y_k^2, \quad \forall k \text{ if } y_k^2 \geqslant \delta$. We ignore the error when the squared-ℓ_2 distance of a pixel location between the reconstruct $\hat{x_t}$ and target x_t is below δ. This allows the model to have the capability to ignore small changes that might occur in the background, focusing on salient parts of the reconstruction. δ is a hyper-parameter to be determined. The second term is the original auto-encoder loss, which is used for regulating the feature space to be meaningful. λ_m is a hyper-parameter that balances the total number of regions of interest. Since there is a penalty for positions that are identified as regions of interest, the loss would force the model to select relatively more important (necessary) parts from x_t and leaving the background in x_s with less penalty.

Attention-Aware Reinforcement Learning

We now discuss the utilization of the self-supervised attention module as a plug for existing deep RL methods. The idea is that for any deep RL methods that uses a CNN, we would exploit the intermediate features extracted by the CNN. Specifically, we multiply the features learned via the attention mask, and leave everything else unchanged for the policy learning.

For the baseline RL algorithm, we use A2C [12]. For a visual observation x, a CNN extracts intermediate feature maps as $f(x) \in \mathbb{R}^{H' \times W' \times C}$. The policy and state value function is then predicted using the feed-forward networks $\pi(a_t|f(x))$, $V(f(x))$ as function approximators. Policy gradient is used to train the networks, and we refer to the loss as $\mathcal{L}_{RL}$, as used in [12]. For the attention-aware RL learning, in addition to the original CNN extracting intermediate feature maps as $f(x)$, an additional self-supervised attention module is used, which takes the visual state x and produces the attention mask $\Psi(x) \in \mathbb{R}^{H' \times W'}$ through the mask generator. The original feature $f(x)$ is multiplied by the attention mask, obtaining the new feature as $\Psi(x)f(x)$. Thus, the policy and state value functions for A2C method are predicted as $\pi(a_t|\Psi(x)f(x))$, $V(\Psi(x)f(x))$.

The self-supervised attention module could be trained offline or jointly trained in an online fashion with the RL agent. For offline training, we sample source and target image pairs $\{(x_s, x_t)\}$ from a pre-collected image set or offline trajectories, and minimize the loss $\mathcal{L}_{mask}$. For joint training with RL agent, the source and target image pairs $\{(x_s, x_t)\}$ are sampled from the online trajectory of the current agent (as in single task learning experiments). The total training loss is $\mathcal{L} = \mathcal{L}_{RL} + \lambda\mathcal{L}_{mask}$, where λ is a mixing coefficient for the two losses. The plugged attention module tries to simplify the original features by suppressing the response of background regions, which helps the abstraction of the observation, and thus improves policy learning.

Experiments

Single-task Learning. Experiments are performed on Atari [1, 2] Environment. In the single-task setting, the self-supervised attention module and the RL agent are jointly trained in an online fashion for each game. Pairs of source and target frames are randomly sampled from the agent's trajectory to train the attention module. The results are shown in Figure 2. We could see that by masking the features using the attention, the agents learn faster and perform better than the baseline A2C method on Atari games shown in Figure 2. The qualitative performance indicates that the attention mask learned is indeed helpful for the understanding of the scene. By only seeing regions of interest, the scene is greatly simplified for understanding and reasoning.

Multi-task Learning - One mask module across different tasks. Unlike the keypoints representation in Transporter [9], where the keypoints are linked to specific objects, or the top-down attention masks that are related to specific RL objectives, the self-supervised attention masks are not semantically restricted in specific scenes or RL objectives. Consequentially, we could potentially train the mask module across a range of tasks, having a *universal* attention mask for many tasks.

To show the generalization ability of the self-supervised attention module, we train the attention module in a multi-task setting. More specifically, we train the self-supervised mask module on frames jointly collected from three different games (Asteroids, Assault, Ms.Pacman) using a random policy. For each training iteration, image pairs (x_s, x_t) are randomly sampled from the three games, and the networks are trained using $\mathcal{L}_{mask}$. We then apply the *universal* trained self-supervised mask module to RL learning of these different games by multiplying the intermediate features $f(x)$ to get $\Psi(x)f(x)$. The agents learn to play each game separately from scratch using $\mathcal{L}_{RL}$ and the attention module parameters are fixed. The results are shown in Figure 3. We see that this *universal* attention module facilitates learning policies on different games, achieving nearly the same performance compared to using the self-supervised attention module specifically trained on one game as in single-task setting (as shown in Figure 2) We conjure that as training data covering more range of tasks, the attention module could have more generalization ability.

Transfer mask across tasks. To further validate the transfer ability of the self-supervised attention module, we design a pipeline that shows the learned attention mask could generalize to related scenes which it has never seen before. First, the self-supervised mask module is trained on frames from the source domain Atari game, *JourneyEscape* using the loss $\mathcal{L}_{mask}$. Then, we fix the parameters of the attention module, and apply it to the RL learning of the target domain game *Asteroids* and

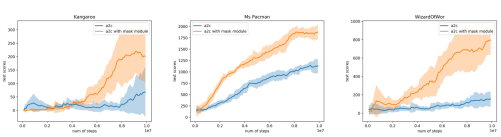

Figure 2: **Single-task Learning**. Average (over 5 random seeds) test scores during learning of A2C with/without the our self-supervised attention mask.

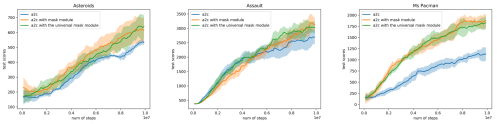

Figure 3: **Multi-task Learning**. Comparison between the baseline method A2C, A2C with the self-supervised attention module, A2C with the the universal attention module jointly trained on three games.

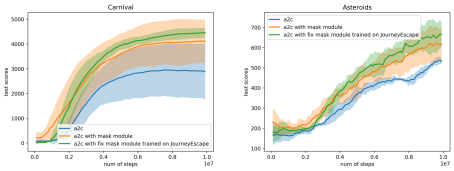

Figure 4: **Transfer Learning**. Comparison between the baseline method A2C and A2C with the fix attention module trained on JourneyEscape, showing that the attention module has the ability to transfer across games.

Carnival in another instance, using the self-supervised attention-aware RL. The results are shown in Figure 4. Notably, even when the attention module has never seen any frames from the target games, the attention masks are still beneficial for the learning of agents as it provide significant gains in the performance. This further highlights that the proposed self-supervised module can generalize to unseen scenarios that have similar visual components, indicating the transfer ability.

Bottom-up Object Extraction. We show preliminary results that our self-supervised attention module could also be used to extract object keypoints to potentially facilitate object-centric reinforcement learning. In particular, we extract object locations from the self-supervised attention masks. Each cell in the attention mask map is considered as a candidate for the center of one object. Non-maximum suppression (NMS) [15] is applied upon the learned attention mask to get the object center proposals. We end up with k object keypoints by taking the k max object proposals with the attention mask value. We compare with Transporter [9] as shown in Figure 5.

Discussion

We designed a self-supervised attention module that could identify salient regions of interest. Our approach is flexible in that the attention mask is not related to particular object semantics or restricted to specific downstream tasks. It is straightforward to plug-and-play the proposed method in existing deep RL approaches with CNNs as feature extractor. Extensive experiments show that the self-supervised attention module not only improves policy learning in the single-task setting, but also, in transfer and multi-task settings. To this end, we presented a universal attention module for multiple scenes allowing the transfer of attention to related unseen scenes. Additionally, we show preliminary results for extracting object keypoints from the self-supervised attention mask. The extracted keypoints reasonably focus on interested objects and are comparable to previous methods specially designed for object keypoints detection. In the future, this ability to extract task-agnostic object keypoints could be potentially used to build symbolic high level representations.

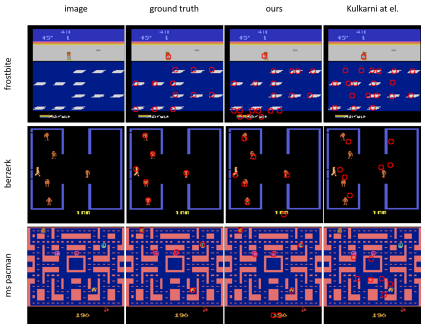

Figure 5: **Qualitative Analysis** Comparison of object keypoints extracted form the self-supervised attention masks, Transporter [9] and the ground truth. The number of object keypoints k are set to the same as Transporter. Our method successfully focuses on important objects and is visually better than Transporter.

References

[1] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

[2] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[3] SM Ali Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, David Szepesvari, Geoffrey E Hinton, et al. Attend, infer, repeat: Fast scene understanding with generative models. In *Advances in Neural Information Processing Systems*, pages 3225–3233, 2016.

[4] Anand Gopalakrishnan, Sjoerd van Steenkiste, and J ürgen Schmidhuber. Unsupervised object keypoint learning using local spatial predictability.

[5] Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Chris Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. *arXiv preprint arXiv:1903.00450*, 2019.

[6] Klaus Greff, Sjoerd Van Steenkiste, and Jürgen Schmidhuber. Neural expectation maximization. In *Advances in Neural Information Processing Systems*, pages 6691–6701, 2017.

[7] Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks through conditional image generation. In *Advances in neural information processing systems*, pages 4016–4027, 2018.

[8] Adam Kosiorek, Hyunjik Kim, Yee Whye Teh, and Ingmar Posner. Sequential attend, infer, repeat: Generative modelling of moving objects. In *Advances in Neural Information Processing Systems*, pages 8606–8616, 2018.

[9] Tejas D Kulkarni, Ankush Gupta, Catalin Ionescu, Sebastian Borgeaud, Malcolm Reynolds, Andrew Zisserman, and Volodymyr Mnih. Unsupervised learning of object keypoints for perception and control. In *Advances in Neural Information Processing Systems*, pages 10723–10733, 2019.

[10] Zhixuan Lin, Yi-Fu Wu, Skand Vishwanath Peri, Weihao Sun, Gautam Singh, Fei Deng, Jindong Jiang, and Sungjin Ahn. Space: Unsupervised object-oriented scene representation via spatial attention and decomposition. *arXiv preprint arXiv:2001.02407*, 2020.

[11] Matthias Minderer, Chen Sun, Ruben Villegas, Forrester Cole, Kevin P Murphy, and Honglak Lee. Unsupervised learning of object structure and dynamics from videos. In *Advances in Neural Information Processing Systems*, pages 92–102, 2019.

[12] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937, 2016.

[13] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. In *NIPS Deep Learning Workshop*. 2013.

[14] 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):529–533, 02 2015.

[15] Azriel Rosenfeld and Mark Thurston. Edge and curve detection for visual scene analysis. *IEEE Transactions on computers*, 100(5):562–569, 1971.

[16] Yuting Zhang, Yijie Guo, Yixin Jin, Yijun Luo, Zhiyuan He, and Honglak Lee. Unsupervised discovery of object landmarks as structural representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2694–2703, 2018.