LEARNING TO PREDICT WHERE TO LOOK IN INTERACTIVE ENVIRONMENTS USING DEEP RECURRENT Q-LEARNING

Sajad Mousavi
School of Computer Engineering
Iran university of Science and Technology (IUST)
Tehran, Iran
smousavi71@gmail.com

Ali Borji
Department of Computer Science
University of Central Florida (UCF)
Orlando, Florida 32816-2365, USA
aborji@crcv.ucf.edu

Nasser Mozayani
School of Computer Engineering
Iran university of Science and Technology (IUST)
Tehran, Iran
mozayani@iust.ac.ir

ABSTRACT

Bottom-Up (BU) saliency models do not perform well in complex interactive environments where humans are actively engaged in tasks (e.g., sandwich making and playing the video games). In this paper, we leverage Reinforcement Learning (RL) to highlight task-relevant locations of input frames. We propose a soft attention mechanism combined with the Deep Q-Network (DQN) model to teach an RL agent how to play a game and where to look by focusing on the most pertinent parts of its visual input. Our evaluations on several Atari 2600 games show that the soft attention based model could predict fixation locations significantly better than bottom-up models such as Itti-Kochs saliency and Graph-Based Visual Saliency (GBVS) models.

1 INTRODUCTION

Human visual attention is attracted either by salient stimuli reflected from the environment or task demands where an observer is attracted to objects of interest (Connor et al., 2004). Researchers have noticed that human visual system does not perceive and process the whole visual information provided at once. Instead, humans selectively pay attention to different parts of the visual input to gather relevant information sequentially and try to combine information from each step over time to build an abstract representation of the entire input (Connor et al., 2004; Rensink, 2000). Works done in neuroscience and cognitive science literature have revealed selection of regions of interest is highly dependent on two modes of visual attention: bottom-up and top-down. The bottom-up mode attends to low-level features of potential importance and attention is represented in the form of a saliency map (Itti et al., 1998; Harel et al., 2006). The top-down mode deals with task demands and goals which strongly influence scene parts to which humans should fixate (Torralba et al., 2006; Hayhoe & Ballard, 2005). Attention models inspired by the human perception have shown good results in a variety of applications of computer vision such as object recognition and detection (Navalpakkam & Itti, 2005; Walther & Koch, 2006), video compression (Itti, 2004; Guo & Zhang, 2010) and virtual reality (Seif El-Nasr et al., 2009). Moreover, with the increased interest in deep learning paradigm in recent years, visual attention mechanisms incorporated in these methods have also shown astonishing results in a wide range of applications including, image captioning (Lin et al., 2014; Xu et al., 2015), machine translation (Bahdanau et al., 2014), speech recognition and object recognition (Gonzalez-Garcia et al., 2015).

Reinforcement learning (Sutton & Barto, 1998) is one of the most powerful frameworks in solving sequential decision making problems, where decision making is based on a sequence of observations

of the tasks environment. Until recently, applying reinforcement learning to some real world applications where the input data is high dimensional (e.g., vision and speech) was a major challenge. Recent advances in deep learning has resulted in powerful tools for automatic feature extraction from raw data, e.g. raw pixels of an image. Research has shown that deep neural networks can be combined with reinforcement learning in order to learn useful representations. For example, Deep Q-Network (DQN) algorithm (Mnih et al., 2013; 2015), which is a combination of Q-learning with a deep neural network, has achieved good performances on several games in the Atari 2600 domain and in some games it can gain even higher scores than the human player. This combination, deep neural networks and reinforcement learning framework, has also been shown to achieve promising results in the computer vision domain (Ba et al., 2014; Sermanet et al., 2014), specifically in visual attention based models. The main challenge in sequential attention models is leaning where to look. Reinforcement learning techniques such as policy gradients are good choices to address this challenge (Mnih et al., 2014).

Overall, there are two types of attention models, the soft attention models and the hard attention models. The soft attention models are end-to-end approaches and differentiable deterministic mechanisms that can be learned by gradient based methods. However, the hard attention models are stochastic processes and not differentiable. Thus, they can be trained by using the REINFORCE algorithm (Williams, 1992) in the reinforcement learning framework. In this paper, a soft attention mechanism is integrated into the deep Q-network (as shown in Figure 1) to enable a deep Q-learning agent to learn to play Atari 2600 games by focusing on the most pertinent parts of its visual input. In fact, the model tries to learn the control actions and the attention locations simultaneously.

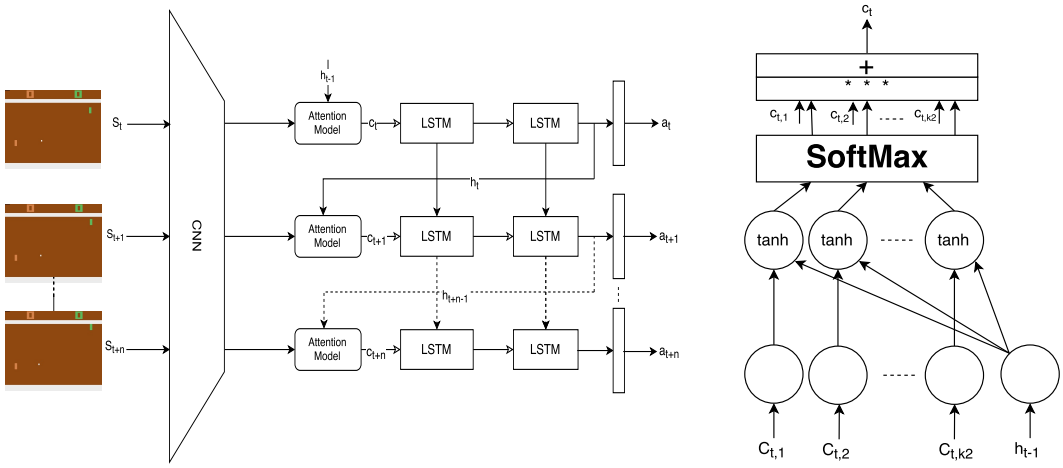
To test our model, we compare predicted fixation locations with explicit attention judgements of people with a specific scenario as explained in section 4.2.2. The results give better fixation prediction accuracy compared to two popular bottom-up (BU) saliency models: Itti-Kochs saliency model (Itti et al., 1998) and Graph-Based Visual Saliency (GBVS) model (Harel et al., 2006). We also demonstrate that the proposed model can learn how to play Atari 2600 games (i.e. leaning the control actions of the game) and where to look (i.e. learning the fixation locations) on video game frames effectively.

2 Related Work

Experiments studying eye movements in natural behaviour have proved that the task plays a strong role in learning where and when human fixate, and that the eyes are guided to the points that are sometimes non-salient (Torralba et al., 2006; Hayhoe & Ballard, 2005; Gonzalez-Garcia et al., 2015). For instance, some works on daily activities such as driving (Land & Lee, 1994), tea making (Land & Hayhoe, 2001) and making a sandwich (Hayhoe & Ballard, 2005) have demonstrated that almost all fixations fall on task-relevant objects and very few irrelevant regions are fixated. Recognizing the influence of reward signals (Schultz, 2000) on eye movements and in directing eye gaze, some researchers (e.g., Sprague & Ballard (2003)) have shown that reinforcement learning is useful for modelling eye movement. They suggested an RL-based method to learn visio-motor behaviours by considering uncertainty costs when an eye movement action is made in a sidewalk navigation task of a virtual urban environment (Sprague et al., 2007), where the goal is maximizing discounted sum of future rewards (or at least cost) over time.

Combining deep learning and reinforcement learning has achieved impressive results in learning to play various video games (Mnih et al., 2013; 2015; Van Hasselt et al., 2015; Guo et al., 2014; Silver et al., 2016) as well as in different problems of computer vision field. For instance, combining deep Convolutional Neural Networks (CNNs) with multi-layered Recurrent Neural Networks (RNNs) in particular, with Long Short-Term Memory (LSTM) components that use reinforcement learning, have led to end-to-end systems for deciding where to look. Such techniques could perform well in classification and recognition tasks, e.g. recurrent neural models of attention for fine-grained categorization on the Stanford Dogs data set (Sermanet et al., 2014), cluttered digit classification on the MNIST date set and playing a toy visual control problem (Mnih et al., 2014), and the recognition of house number sequences from Google Street View images (Ba et al., 2014).

Both soft attention and hard attention mechanisms have gained remarkable performance over different challenging problems. Recent research done by Xu et al. (2015) has applied both mechanisms to generate image captions. The attention mechanism in this work aims to generate each word of the



(a) (b)

Figure 1: (1a) Architecture of our model (a deep Q-network with possibility of visual attention). The CNN part of the model takes as input a sequence of states of the game and extracts feature maps. It then computes vertical feature slices, C_t with dimension D. (1b) The attention model. at each time step t, the attention model uses C_t along with the previous internal state of the LSTM part, h_{t-1} to produce an expected value, c_t of vertical feature slices, $C_{t,i}$ regarding $\alpha_{t,i}$, the importance of each part of input frame.

caption by focusing on the relevant parts of the image. Another approach using an attention mechanism is called spatial transformer networks (Jaderberg et al., 2015). Spatial transformer networks have used affine transformations as a soft attention mechanism to focusing on the relevant part of an image. More recently, Sharma et al. (2015) used a recurrent soft attention based model for action recognition in videos and showed that their model performs better than some baselines (without the attention mechanism). In this paper, inspired by the work of Sharma et al. (2015), we propose a soft attention mechanism incorporated in a Deep Q-Network (Mnih et al., 2013; 2015) to learn where to look and which actions to select. Wang et al. (2015) have proposed a new neural network architecture to play Atari 2600 games, which is an extension of the Double Deep Q-Network (Double DQN) in (Van Hasselt et al., 2015). Applying their method to several Atari 2600 games, they obtained state-of-the-art results. They also visualized saliency maps, indicating where the agent looks at, while taking an action. To this end, they generated the saliencies by computing the Jacobians of the state-value and action-value functions, inspired by the way introduced in (Simonyan et al., 2013). In contrast, here, we use multi-layered LSTMs and the soft attention mechanism to predict fixation locations and compare our models accuracy with the bottom-up saliency models.

3 Model Description

3.1 Model Architecture: ConvNet + Attention Model + LSTM

The idea behind the proposed model is similar to the work done in (Sharma et al., 2015), but with differences in terms of the way the model is trained (i.e. reinforcement learning) and the application domain. Here we are interested in learning to play Atari 2600 games by selecting the appropriate control actions provided for each game while paying attention to the most important locations.

Our model is an aggregation of CNN, soft attention mechanism and RNN (i.e. LSTM layers), as shown in Figure 1. In order to extract convolutional features, we used the first three layers of the Deep Q-Network (DQN) (Mnih et al., 2015) as the CNN part of the model, three stacks of convolutions plus rectifier nonlinearity layers with filters 32 8×8, 64 4×4 and 64 3×3, and strides $4, 2$ and 1, respectively. At each time-step t, a visual frame is fed into the system and the last convolutional layer of the CNN part outputs D feature maps $K \times K$ (for example, here 64 feature maps 7×7). Then, feature maps are converted to K^2 vectors in which each vector has D dimension

as follows:

$$C_t = [C_{t,1}, C_{t,2}, \ldots, C_{t,K^2}], \qquad C_{t,i} \in \mathbb{R}^D.$$

Next, they are fed to the attention model to compute the probabilities corresponding to the importance of each part of the input frame. In other words, the input frame is divided into K^2 regions and the attention mechanism tries to attend to the most relevant region.

We utilize the soft attention mechanism of Xu et al. (2015) (See Bahdanau et al. (2014); Cho et al. (2015) for a detailed discussion). Figure 1b shows the structure of the attention mechanism. The attention model takes K^2 vectors, $C_{t,1}, C_{t,2}, \ldots, C_{t,K^2}$ and a hidden state h_{t-1}. It then produces a vector c_t which is a linear weighted combination of the values of $C_{t,i}$. The h_{t-1} is the internal state of the LSTM at the previous time step. Each vector $C_{t,i}$ is a representation of different regions of the input frame. More formally, the attention module tries to select the vectors using a linear combination of h_{t-1} and $C_{t,i}$, i.e:

$$c_t = E[C_t] = \sum_{i=1}^{k \times k} \alpha_{t,i} C_{t,i}, \tag{1}$$

$$\alpha_{t,i} = \frac{\exp(f_{att}(C_{t,i}, h_{t-1}))}{\sum_{j=1}^{k \times k} \exp(f_{att}(C_{t,i}, h_{t-1}))} \qquad i \in 1, 2, \ldots, k \times k, \tag{2}$$

$$f_{att}(C_{t,i}, h_{t-1}) = \tanh({W_{hatt}}^T h_{t-1} + {W_{catt}}^T C_{t,i}), \tag{3}$$

At each time step t, attention module computes f_{att}, a $\tanh$ layer which is a composition of the values of $C_{t,i}$ and h_{t-1}. Then uses it to calculate $\alpha_{t,i}$, a softmax over $k \times k$ regions. They can be considered as the amount of the importance of the corresponding vector $C_{t,i}$ among of $k \times k$ vectors in the input image. Once $\alpha_{t,i}$ values are ready, the attention model calculates c_t, a weighted sum of all vectors $C_{t,i}$ based on given $\alpha_{t,i}$s. Thus, in this way, the RL agent can learn to emphasize on the interesting part of the input frame based on the given state. Note that the soft attention model is fully differentiable which allows training the systems in an end-to-end manner.

The RNN part of the network, which is a stack of two LSTM layers with the LSTM sizes 64, uses the previous hidden state h_{t-1} and the output of the attention module c_t to calculate the next hidden state h_t. The h_t is used as input of the output layer, which is a fully-connected linear layer with the output neurons for each legal action of the game. It is also utilized as input of the attention module in order to calculate the value c_{t+1} at the next time-step.

3.2 Model Training

The goal of the RL agent is to learn an optimal policy π, the probability of selecting action a in state s by focusing on the most relevant part of the given state, such that by following the underlying policy the sum of the discounted rewards over time is maximized. Since the proposed model is an end-to-end learning system, all of the network parameters (i.e. three parts of the model: ConvNet, the attention network and the LSTMs), θ_t can be learned by trying to minimize the following loss function of mean-squared error in Q values:

$$L(\theta) = E[(r + \gamma max_{a'} Q(s', a'; \theta_{t-1}) - Q(s, a; \theta_t))^2], \tag{4}$$

where $r + \gamma max_{a'} Q(s', a'; \theta_{t-1})$ is the target value, r is a scalar reward which the agent receives after taking action a in state s. Parameter $0 \leqslant \gamma \leqslant 1$ is called the discount factor. For optimizing the above loss function, we utilize the stochastic gradient descent method. Thus, in the Q-learning algorithm, the parameters are updated as follows:

$$\theta_i = \theta_{i-1} + \alpha(y_i - Q(s, a; \theta_i)) \frac{\partial Q(s, a; \theta_i)}{\partial \theta_i}, \tag{5}$$

where it is implicit that $y_i = r + \gamma max_{a'} Q(s', a'; \theta_{t-1})$ is the target value for iteration i, and α is a learning rate.

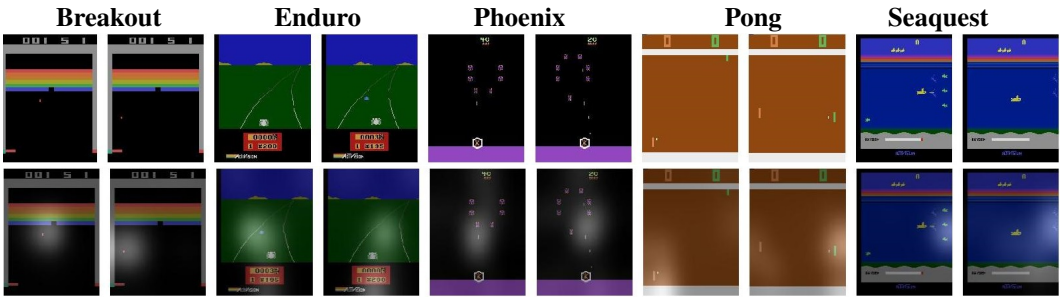


Figure 2: Visualization of attended locations by the proposed model over sample frames of the five Atari 2600 games. The white regions of frames indicate where the model is interested to look at. Higher brightness means more attention.

4 Experiment and Results

4.1 Evaluation metrics and Hyperparameters

To compare the power of the proposed model for predicting fixation locations against the other models, we used two metrics: Normalized Scanpath Saliency (NSS) (Peters et al., 2005) and Area Under Curve (AUC) (Bruce & Tsotsos, 2005).

NSS is used for evaluating the salience map values using fixated locations. It is a response value at the human eye position (x_h, y_h) of the saliency map (sm) suggested by a model, where saliency scores of the predicted map density have zero mean and unit standard deviation. NSS can be formulated as follows:

$$NSS = \frac{sm(x_h, y_h) - \mu_{sm}}{\sigma_{sm}}, \tag{6}$$

where ${\sigma^2}_{sm}$ and μ_{sm} are the variance and the mean of the saliency map (sm).

NSS equal 1 means that the subjects eye position fall in a region where the predicted map density is one standard deviation above average, while NSS $= 0$ suggests the model would not perform better than random. NSS < 0 indicates attention to non-salient locations predicted by the model.

AUC is the area under the Receiver Operating Characteristics (ROC) curve (Bruce & Tsotsos, 2005). It is one the most commonly used ways to evaluate the performance of a binary classifier. The ROC curve is created based on the true positive rate (TPR) against the false positive rate (FPR) for different cut-off points of a parameter (here, the salience value). Each pair (TPR, FPR) of the ROC curve is computed with a particular threshold. To use this metric in this work, human fixations on the image are considered as ground truth and the saliency map can be viewed as a binary classifier to classify each pixel of the image to fixated and non-fixated samples. By drawing the ROC curve and calculating the area under the ROC curve, we can evaluate how well the saliency map predicts actual human eye fixations. The perfect predication has a score of 1.0, while a score of 0.5 indicates that the model is no better chance.

In all of our experiments, we trained the system with a 2-layer LSTM with 64 hidden/cell units and a value of 4 for the number of sequence steps in backpropagation through time. In order to ensure stable learning, LSTM gradients were clipped to a value of 10 (Hausknecht & Stone, 2015). At each step of training the network, the initial LSTM hidden and cell states were set to zero. Exploration strategy of the agent was ϵ-greedy policy with ϵ decreasing linearly from 1 to 0.1 over the first million steps. The discount factor was set to $\gamma = 0.99$. The networks were trained for 2 million steps and the size of the replay memory was 500, 000. All network weights were updated by the RMSProp optimizer (Yu, 2010) with mini batches of size 32, a momentum of 0.95, and a learning rate of $\alpha = 0.00025$. Training for all the games was done with the same network architecture and hyperparameters.

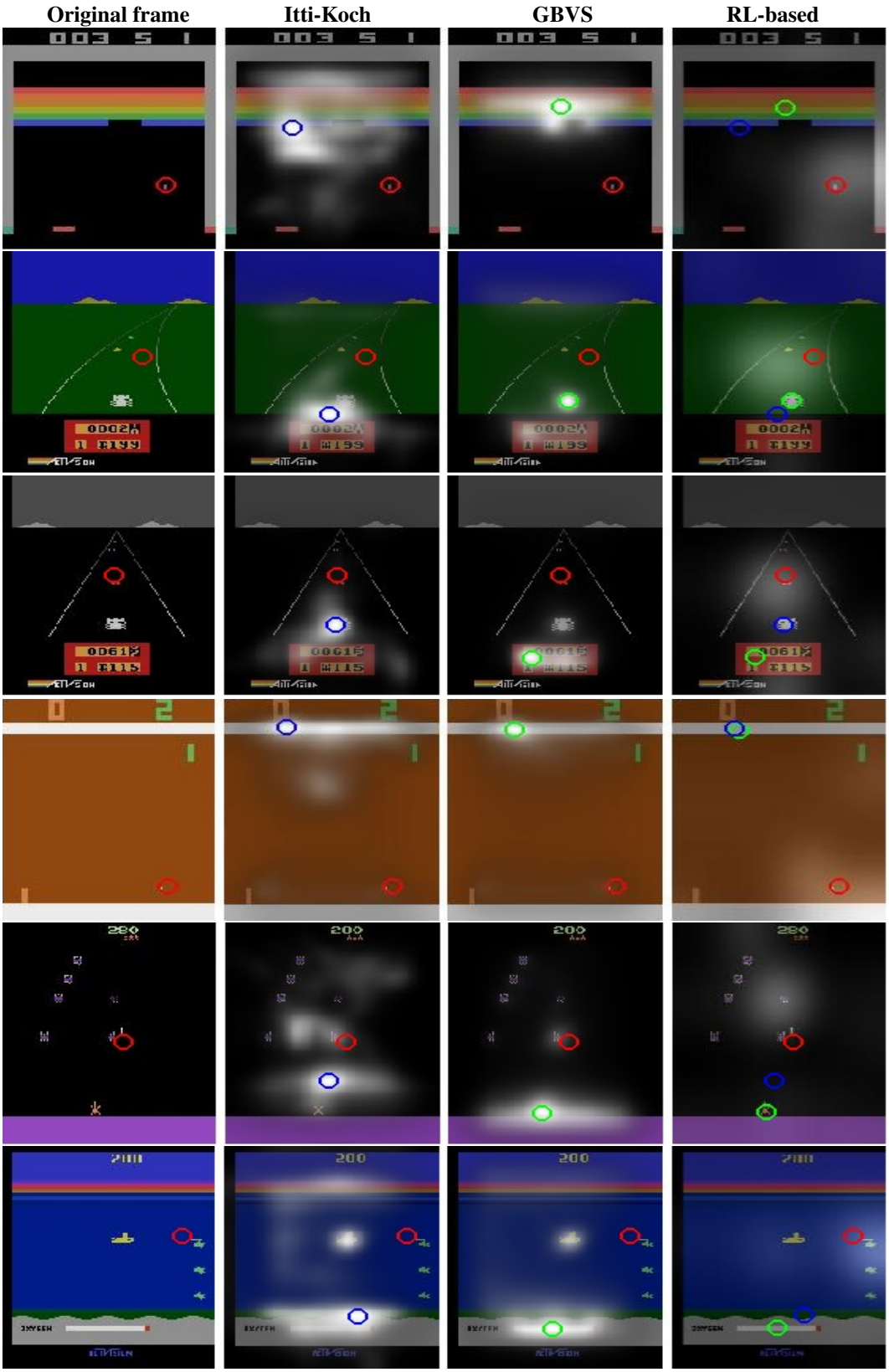


Figure 3: Each row shows a sample frame of an Atari 2600 game and corresponding saliency maps overlaid on the frame from different models. The red circles indicate the human fixation (i.e., the clicked positions by the subjects). The green and blue circles indicate the location of the maximum point in each map for GBVS and Itti-Koch algorithms, respectively. The white parts indicate attended regions by the model. Higher brightness means more attention.

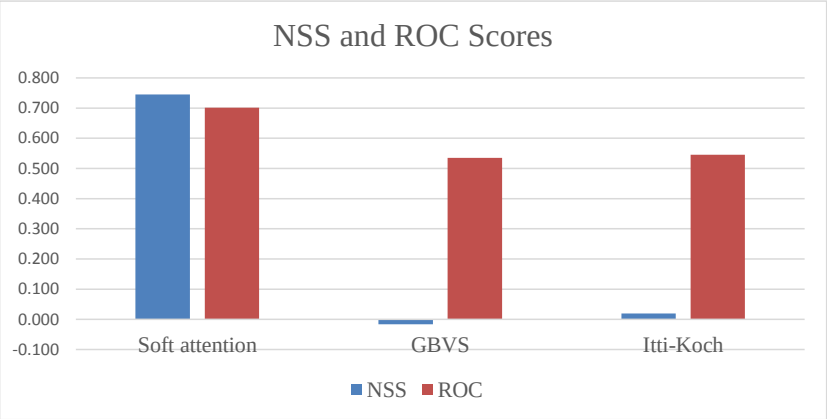


Figure 4: NSS and ROC scores of predicted fixation locations via the different models.

4.2 Experiment Setup

4.2.1 The Arcade Learning Environment

We have used Atari 2600 games in the Arcade Learning Environment (ALE) (Bellemare et al., 2013). The ALE provides an environment that emulates the Atari 2600 games. It presents a very challenging environment for the reinforcement learning and other approaches that have a high dimensional visual input (210×160 RGB video at 60Hz), which might also be partially observable. It presents a wide range of interesting games that are useful to be a standard testbed for evaluating the novel algorithms. In our experiments, we used 5 Atari games: Pong, Phoenix, Enduro, Breakout and Seaquest. Our goal is training a RL agent to learn a certain optimal policy to play each of the games and attention to task-relevant parts of the input frame. In fact, learning to choose the optimal action based on the region selected from among provided candidates (vertical feature slices resulting from the last convolution layer of the CNN part of the model, see C_t in section 3.1). Figure 2 shows some sample frames of the utilized games as well as the learned attended regions.

4.2.2 Psychophysics and Collecting Data

To test the accuracy of the model, we recruited three subjects (two males and one female). They were not experts in playing Atari 2600 games. So, at first we provided them five games (Pong, Phoenix, Enduro, Breakout and Seaquest) of the set of Atari 2600 games. We asked subjects to play the games to learn how to play and familiarize themselves with the environment. Second, we showed the recorded videos played by the RL agent to subjects (the frame rate: 5fps; higher rate was a little fast for the subjects to concentrate). Subjects were asked to click at frame locations where they thought should be looked at while actually playing the games. Then, we collected the clicked positions and corresponding frames as the ground truths (surrogates for actual eye positions of the subjects) in our analyses. For each game, we considered one episode, which is a trajectory from initial to a terminal state.

4.2.3 Bottom-up saliency models

In order to evaluate the suggested model, we compared the performance of our model against two popular bottom-up saliency (BU) models: Itti-Kochs saliency model (Itti et al., 1998) and Graph-Based Visual Saliency (GBVS) introduced by Harel et al. (2006). We used the freely available implementations of the saliency models provided by Harel[1]. Itti-Kochs bottom up model uses twelve features extracted from the image, including color contrast (red/green and blue/yellow), temporal luminance flicker, luminance contrast, four orientations ($0°$, $45°$, $90°$, $135°$), and four oriented motion energies (up, down, left, right). Next, it computes conspicuity maps of individual features

[1]J. Harel, A saliency implementation in MATLAB: `http://www.klab.caltech.edu/~harel/share/gbvs.php`

through center surround architecture inspired by biological receptive fields. It then linearly combines saliency from different features to generate a unique master saliency map representing the conspicuity of each location in the visual input. The GBVS algorithm attempts to create feature maps in the way done by Itti-Kochs method, but normalizes them using a graph based approach to reach a better combination of conspicuous maps.

Table 1: Comparison of performance of the proposed model against bottom-up models on several frames of the five Atari 2600 games

	Breakout (724 frames)		**Enduro (2914 frames)**		**Phoenix (575 frames)**	
Model	NSS	ROC	NSS	ROC	NSS	ROC
Soft attention	1.326	0.787	0.699	0.689	0.187	0.529
GBVS	-0.074	0.489	-0.083	0.578	-0.143	0.485
Itti-Koch	-0.112	0.453	-0.048	0.593	0.267	0.599

	Pong (1346 frames)		**Seaquest (577 frames)**	
Model	NSS	ROC	NSS	ROC
Soft attention	0.846	0.76	0.571	0.694
GBVS	0.179	0.478	0.058	0.564
Itti-Koch	0.129	0.472	0.023	0.54

4.3 RESULTS

We ran the two bottom-up saliency models (i.e. Itti-Koch saliency model and the GBVS algorithm) over five Atari 2600 games and compared their NSS and AUC scores to scores achieved by our suggested model. Table 1 reports a comparison of the proposed model with those bottom-up models. The results on Table 1 demonstrate that the soft attention model significantly performs better than the Itti-Koch and the GBVS methods using both evaluation metrics over all five games. We observe that NSS and ROC scores for some games like Breakout, Enduro and Pong are better than others. Further, movements of objects (especially, enemies) in the games Phoenix and Seaquest are high compared to the other games which makes learning an appropriate control policy difficult. As a result to better predict fixation locations, the algorithm needs more than 2 million time steps. For fair comparison, similar to the other games, we stopped training the networks of these games after 2 million steps.

Figure 3 shows several sample frames of five Atari games and their corresponding saliency maps overlaid on the frames from the investigated models. The distance between interesting places of the frame for the subject (shown in red circles in the figure) and predicted fixation locations (i.e. the location of the maximum point in the saliency map) of the models (shown in blue and green circles) illustrates how well a models predictions match subjects click positions. We see that the soft attention model (RL-based model) is able to suggest significantly better attention predictions compared to the GBVS and Itti-Koch methods.

Figure 4 shows the NSS and ROC scores of the model across all frames of five games. The plot illustrates a big performance difference between bottom-up saliency models, and the proposed model. For instance, the soft attention model obtains mean NSS and ROC scores of 0.74 and 0.70 while bottom-up saliency models do not perform better than random. From these results, we learn that bottom-up saliency models performs poorly in interactive tasks where humans are actively engaged in tasks (e.g. playing video games). The interested readers are referred to Borji et al. (2014), where they propose several top-down attention models and exhaustively compare them with bottom-up saliency models.

5 CONCLUSION

In this paper, we utilized a combination of Deep Q-network algorithm and recurrent soft attention mechanism to decide where to look as well as learn action selection in complex interactive environments. We applied the proposed model to predict fixation locations, while a reinforcement learning agent is playing Atari 2600 games. To show fixation prediction accuracy of our model, we set up a

scenario to collect click locations (as measure of explicit attention) from a number of subjects while they are watching the videos of played games. We ran the two most commonly used bottom-up saliency models over five recorded video game video clips which have been played by a RL agent. We compared the model to those bottom up models. Our experiments show that saliency maps suggested by our model obtain significantly better AUC and NSS scores. Despite the remarkable results, we are already feeding the entire frame into the system, which is computationally expensive. Using only focused parts of the given frame as input of the system (i.e., foveated representation) would be an interesting future research direction.

REFERENCES

Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Marc G. Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: an evaluation platform for general agents. *J. Artif. Int. Res.*, 47(1):253–279, 2013. ISSN 1076-9757.

Ali Borji, Dicky N Sihite, and Laurent Itti. What/where to look next? modeling top-down visual attention in complex interactive environments. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(5):523–538, 2014. ISSN 2168-2216.

Neil Bruce and John Tsotsos. Saliency based on information maximization. In *Advances in neural information processing systems*, pp. 155–162, 2005.

Kyunghyun Cho, Aaron Courville, and Yoshua Bengio. Describing multimedia content using attention-based encoder-decoder networks. *IEEE Transactions on Multimedia*, 17(11):1875–1886, 2015. ISSN 1520-9210.

Charles E Connor, Howard E Egeth, and Steven Yantis. Visual attention: bottom-up versus top-down. *Current Biology*, 14(19):R850–R852, 2004. ISSN 0960-9822.

Abel Gonzalez-Garcia, Alexander Vezhnevets, and Vittorio Ferrari. An active search strategy for efficient object class detection. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3022–3031. IEEE, 2015. ISBN 1063-6919.

Chenlei Guo and Liming Zhang. A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression. *IEEE Transactions on Image Processing*, 19(1):185–198, 2010. ISSN 1057-7149.

Xiaoxiao Guo, Satinder Singh, Honglak Lee, Richard L Lewis, and Xiaoshi Wang. Deep learning for real-time atari game play using offline monte-carlo tree search planning. In *Advances in neural information processing systems*, pp. 3338–3346, 2014.

Jonathan Harel, Christof Koch, and Pietro Perona. Graph-based visual saliency. In *Advances in neural information processing systems*, pp. 545–552, 2006.

Matthew Hausknecht and Peter Stone. Deep recurrent q-learning for partially observable mdps. *arXiv preprint arXiv: 1507.06527v3*, 2015.

Mary Hayhoe and Dana Ballard. Eye movements in natural behavior. *Trends in cognitive sciences*, 9(4):188–194, 2005. ISSN 1364-6613.

Laurent Itti. Automatic foveation for video compression using a neurobiological model of visual attention. *IEEE Transactions on Image Processing*, 13(10):1304–1318, 2004. ISSN 1057-7149.

Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 20(11):1254–1259, 1998. ISSN 0162-8828.

Max Jaderberg, Karen Simonyan, and Andrew Zisserman. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pp. 2017–2025, 2015.

Michael F Land and Mary Hayhoe. In what ways do eye movements contribute to everyday activities? *Vision research*, 41(25):3559–3565, 2001. ISSN 0042-6989.

Michael F Land and David N Lee. Where do we look when we steer. *Nature*, 1994. ISSN 1476-4687.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollr, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pp. 740–755. Springer, 2014.

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

Volodymyr Mnih, Nicolas Heess, and Alex Graves. Recurrent models of visual attention. In *Advances in Neural Information Processing Systems*, pp. 2204–2212, 2014.

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, 2015. ISSN 0028-0836. doi: 10.1038/nature14236. URL `http://dx.doi.org/10.1038/nature14236`.

Vidhya Navalpakkam and Laurent Itti. Modeling the influence of task on attention. *Vision research*, 45(2):205–231, 2005. ISSN 0042-6989.

Robert J Peters, Asha Iyer, Laurent Itti, and Christof Koch. Components of bottom-up gaze allocation in natural images. *Vision research*, 45(18):2397–2416, 2005. ISSN 0042-6989.

Ronald A Rensink. The dynamic representation of scenes. *Visual cognition*, 7(1-3):17–42, 2000. ISSN 1350-6285.

Wolfram Schultz. Multiple reward signals in the brain. *Nature reviews neuroscience*, 1(3):199–207, 2000. ISSN 1471-003X.

Magy Seif El-Nasr, Thanos Vasilakos, Chinmay Rao, and Joseph Zupko. Dynamic intelligent lighting for directing visual attention in interactive 3d scenes. 2009.

Pierre Sermanet, Andrea Frome, and Esteban Real. Attention for fine-grained categorization. *arXiv preprint arXiv:1412.7054*, 2014.

Shikhar Sharma, Ryan Kiros, and Ruslan Salakhutdinov. Action recognition using visual attention. *arXiv preprint arXiv:1511.04119*, 2015.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, and Marc Lanctot. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016. ISSN 0028-0836.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Nathan Sprague and Dana Ballard. Eye movements for reward maximization. In *Advances in neural information processing systems*, pp. None, 2003.

Nathan Sprague, Dana Ballard, and Al Robinson. Modeling embodied visual behaviors. *ACM Transactions on Applied Perception (TAP)*, 4(2):11, 2007. ISSN 1544-3558.

Richard S. Sutton and Andrew G. Barto. *Introduction to Reinforcement Learning*. MIT Press, 1998. ISBN 0262193981.

Antonio Torralba, Aude Oliva, Monica S Castelhano, and John M Henderson. Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search. *Psychological review*, 113(4):766, 2006. ISSN 1939-1471.

Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. *CoRR, abs/1509.06461*, 2015.

Dirk Walther and Christof Koch. Modeling attention to salient proto-objects. *Neural networks*, 19 (9):1395–1407, 2006. ISSN 0893-6080.

Ziyu Wang, Nando de Freitas, and Marc Lanctot. Dueling network architectures for deep reinforcement learning. *arXiv preprint arXiv:1511.06581*, 2015.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992. ISSN 0885-6125.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2(3):5, 2015.

Huizhen Yu. Convergence of least squares temporal difference methods under general conditions. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 1207–1214, 2010.