Why the Agent Made that Decision: Contrastive Explanation Learning for Reinforcement Learning

Rui Zuo, *Syracuse University, Syracuse, NY, 13210, USA*

Zifan Wang, *Syracuse University, Syracuse, NY, 13210, USA*

Simon Khan, *Air Force Research Laboratory, Rome, NY, 13441, USA*

Garrett Ethan Katz, *Syracuse University, Syracuse, NY, 13210, USA*

Qinru Qiu, *Syracuse University, Syracuse, NY, 13210, USA*

Abstract—Reinforcement learning (RL) has demonstrated remarkable success in solving complex decision-making problems, yet its adoption in critical domains is hindered by the lack of interpretability in its decision-making processes. Existing explainable AI (xAI) approaches often fail to provide meaningful explanations for RL agents, particularly because they overlook the contrastive nature of human reasoning—answering "why this action instead of that one?" To address this gap, we propose a novel framework of contrastive learning to explain RL selected actions, named ***VisionMask****. VisionMask is trained to generate explanations by explicitly contrasting the agent's chosen action with alternative actions in a given state using a self-supervised manner. We demonstrate the efficacy of our method through experiments across diverse RL environments, evaluating it in terms of faithfulness, robustness and complexity. Our results show that VisionMask significantly improve human understanding of agent behavior while maintaining accuracy and fidelity. Furthermore, We present examples illustrating how VisionMask can be used for counterfactual analysis. This work bridges the gap between RL and xAI, paving the way for safer and more interpretable RL systems.*

Deep Reinforcement Learning (DRL) is a powerful technology in machine intelligence, widely used for many applications.[1] However, understanding a DRL agent's decision-making process is challenging, due to the inherent lack of explainability in the high-dimensional, non-linear structure of its underlying Deep Neural Network (DNN).[2] The lack of transparency undermines users' trust, driving the development of Explainable AI (xAI). Various methods in computer vision have been proposed to enhance the transparency of AI systems.[3–5] At the core, they share a common foundation: **attributing** the classifier's outputs to more interpretable features and using a saliency map to visualize these attributions. Their only differences are how these attributions are calculated. A high-quality attribution-based explanation should meet several key criteria. First, it should demonstrate *faithfulness*, meaning that including features with high attribution should lead the model to the target output, and excluding them should prevent it. Second, it should exhibit *specificity*, ensuring that only critical features receive high attribution. Sometime this is also referred to as *sparseness*. Finally, it should be *robust*, meaning the explanation should remain consistent and not change significantly with minor variations in the input. Attribution-based explanation has been studied for DRL models. Exp-Atari[6] and SARFA[7] utilized policy distributional shifts as the basis for attribution in RL. Specifically, they calculate attribution of a feature as the difference in Q/V values or action distributions between the original and perturbed states. For example, given agent policy π, the attribution of a feature is proportional to $E_{s'}(|\pi(s) - \pi(s')|_2)$ where s stands for the original state and s' represents perturbed states generated for this feature. By calculating the attributions for all features, a saliency map m can be created. However, the perturbation-based explanations lack faithfulness. Since each perturbation focuses only on local features while ignoring the joint impact of feature combinations, overlaying the saliency map with the original state, $(m \odot s)$, does not result in a feature combination that leads the agent to the

target action distribution $\pi(s)$. A better approach to enhance faithfulness is to **learn** a model to predict the saliency map m that minimizes the difference between $\pi(m \odot s)$ and $\pi(s)$. *Explainer*[8] leveraged this idea by training an explanation model for an image classifier. However, Explainer categorize class labels into target and non-target for each training sample and focus on learning saliency map (or mask) only for the target label while treat all non-target labels as a single group. Unlike a (well trained) image classifier, where predictions for non-target labels are typically close to 0, DRL agent in many scenarios, does not exhibit a clear preference for the actions. Non-target actions may sometimes have probabilities only slightly lower than those of target actions. Analyzing how masking the feature may affect the non-target action probability provides additional information that can be used to train the explanation model more effectively. The above analysis motivates us to design a trainable saliency map generator for attribution-based explanations and train it using two channels of contrastive information:**(i) Action-wise contrast:** We believe that environment states contain features that motivate the DRL agent to select both target action and non-target actions. However, the target action is ultimately chosen because it corresponds to higher reward or has a stronger presence. For each action, a saliency map can be generated as an explanation. Choosing features according to the saliency map for a non-target action should push the agent away from the target action, and vice versa. This inspired us to treat the saliency map of the non-target action $m_{a'}$ and the target action m_a as a negative pair, which can be leveraged for contrastive learning.[9–10] **(ii) Feature-wise contrast:** To exclude irrelevant features (e.g. background) from the saliency map, explanations also need to be discriminative in filtering out such information. When only irrelevant features are accessible to the agent, the resulting action distribution should be as uniform as possible. Therefore, the target action's saliency map (m) and its inverted counterpart ($\tilde{m} = 1 - m$) form another negative pair for contrastive learning.

In this work, we present VisionMask as an RL explainer that is contrastively trained to generate saliency maps to explain agent's actions. We specifically focus on agents that maps images to actions and consider each pixel value as the interpretable input feature, although the similar technique could be extended to other type of features. We carefully design the objective function to enable self-supervised contrastive learning of explanations from both action-wise and feature-wise perspectives, fostering the generation of more faithful explanations. We conduct evaluation on six RL environments with five baselines based on faithfulness, robustness, and sparseness. Quantitatively, VisionMask outperforms the baselines in terms of faithfulness, while exhibiting strong robustness and high sparseness. Qualitatively, we compared VisionMask with the baselines in two settings: visual comparison and human studies. In the visual comparison, VisionMask provides sharper explanations that align more closely with human interpretations, as demonstrated by counterfactual examples. In the human studies, VisionMask's explanations help users better understand the agent's decisions and calibrate appropriate trust.

VISIONMASK

In this section, we present our VisionMask[1] architecture. The primary goal is to generate action-wise saliency maps that attribute the most relevant features in the state to each action. For agents that map images to actions, the features and states correspond to pixels and images.

Problem Formulation Formally, we define the environment as a Markov Decision Process (MDP) $\{\mathcal{S}, \mathcal{A}, P, R, \gamma\}$, where $\mathcal{S}$ represents the state space; $\mathcal{A}$ denotes the action space with $|\mathcal{A}| = K$; the state transition function $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ depicts the transition between states based on actions, where $\Delta(\mathcal{S})$ represents the set of probability distributions over states; the reward function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ provides the immediate reward for state-action pairs; and $\gamma \in [0, 1]$ and $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ represent the discount factor and policy. Return G is defined as $G = \sum_{k=0}^{\infty} \gamma^k R_{k+1}$, and the expected cumulative reward of a policy π is $\mathbb{E}_\pi[G] = \mathbb{E}_\pi[\sum_{k=0}^{\infty} \gamma^k R_{k+1}]$, where the expectation is taken with respect to the initial state distribution, transition probabilities, and action probabilities determined by π. VisionMask operates on a given trained expert policy π_E such that $\pi_E \approx \pi^* = \arg\max_\pi \mathbb{E}_\pi[G]$, where π^* is the optimal policy. We can obtain a dataset of expert demonstrations $\mathcal{D}_E = \{(s_i, \pi_E(s_i))\}_{i=1}^N$ consisting of N state-action pairs, from trajectories sampled while executing π_E in the environment. Our goal is to learn an explainer f_{θ^*} that minimizes the loss $\theta^* = \arg\min_\theta \sum_{(s,a)\in\mathcal{D}_E} \mathcal{L}(a, s, \theta)$ where $\mathcal{L}$ is the training loss function to be discussed in next section. The explainer function $f_\theta : \mathcal{S} \to [0, 1]^{K \times d_s}$, wher d_s represents the feature size of the state $s \in \mathcal{S}$ and K denotes the number of candidate actions, predicts the attributions of each action to each feature in the state

[1] A preprint version is available: https://arxiv.org/abs/2411.16120.

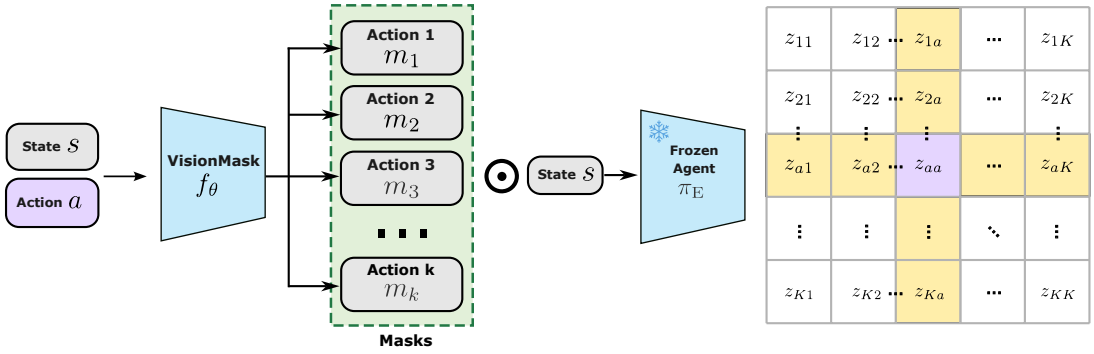

FIGURE 1: **Architecture of VisionMask.**

s. The value of the output is bounded within the range $[0, 1]$, with the (i, j)th element indicates the jth feature's attribution to the ith action.

Architecture As shown in Figure 1, we first collect the expert dataset $\mathcal{D}_E$ using the expert policy π_E. From this dataset $\mathcal{D}_E$, state-action pairs (s_i, a_i) are sampled and fed to VisionMas f_θ to generate the set of saliency maps M. Generating the saliency map from given visual input is a dense prediction task that shares similarities with image segmentation, where each pixel is assigned a value to indicate whether it belongs to an object or background. Hence, we structure the explainer f_θ akin to the widely used image segmentation model, DeepLabv3[14], however, retrain it using self-supervised contrastive learning. To make sure that the output saliency value are bounded to the range $[0, 1]$, a sigmoid function is applied at the output of f_θ. For each $m_i \in M$, we also calculate a complement map $\tilde{m}_i = \mathbf{1} - m_i$ highlighting the irrelevant regions for the action i. Then the masks m_i and $\tilde{m}_i$ are overlaid onto the original state s to generate two masked states s_i and $\tilde{s}_i$ using the following overlay function: $s_i = s \odot m_i + r \odot (\tilde{m}_i)$, $\tilde{s}_i = s \odot \tilde{m}_i + r \odot (m_i)$ where $\odot$ is Hadamard Product and r is a reference value. Numerous options exist for the reference value, such as setting the pixel to zero, assigning a constant value, blurring the pixel, or cropping it. Empirical study shows that setting the reference to the background gives the best results. To generate self-supervised contrastive loss to train the model, we query the agent to obtain the corresponding logits $z_i = \pi_E(s_i)$ and $\tilde{z}_i = \pi_E(\tilde{s}_i)$, and the action probabilities $p_i = \text{softmax}(z_i)$ and $\tilde{p}_i = \text{softmax}(\tilde{z}_i)$, where $p, p_i \in [0, 1]^K$, $0 \leq i < K$. By concatenating each p_i and $\tilde{p}_i$, we have the the action probabilities of each mask $\mathbf{p}, \tilde{\mathbf{p}} \in \mathbb{R}^{K \times K}$,

$$\mathbf{p} = [p_1, p_2, \dots, p_k]^T \quad \tilde{\mathbf{p}} = [\tilde{p}_1, \tilde{p}_2, \dots, \tilde{p}_k]^T$$

Training Loss

To enable the agent contrastively learn the saliency map m_a, we carefully designed the training loss function $\mathcal{L}$ as follows:

$$\mathcal{L}(\mathbf{s}, \mathbf{a}, \theta) = \mathcal{L}_a(\mathbf{s}, \mathbf{a}) + \lambda_{ne}\mathcal{L}_{ne}(\mathbf{s}) + \lambda_{area}\mathcal{L}_{area}(\mathbf{s}, \mathbf{a})$$

where $\mathcal{L}_a(\mathbf{s}, \mathbf{a})$ is action-wise contrastive action loss, $\mathcal{L}_{ne}(\mathbf{s}, \mathbf{a})$ is feature-wise loss, $\mathcal{L}_{area}(\mathbf{s}, \mathbf{a}, \mathbf{n})$ is the area size loss and $\lambda_{ne}, \lambda_{area}$ are regularization hyper-parameters.

Action-wise contrastive loss $\mathcal{L}_a$. Let a denote the target action chosen by the agent. Our primary goal is to learn explanations faithful to the a, making (a, p_a) the only positive pair. Furthermore, the explanation must be discriminative, meaning it should clearly distinguish the target action a from all other possible actions. As a result, every other pair of $(a, p_{i \neq a})$ is treated as a negative pair. Then, we compute the cross-entropy loss between these pairs,

$$\mathcal{L}_{pos}(\mathbf{s}, \mathbf{a}) = -\frac{1}{K}\sum_{k=1}^{K}[\![k = \mathbf{a}]\!] \log\left(\frac{\exp(z_{aa})}{\sum_{k=1}^{K}\exp(z_{ak})}\right) \quad (1)$$

$$\mathcal{L}_{neg}(\mathbf{s}, \mathbf{a}) = -\frac{1}{K}\sum_{k=1}^{K}[\![k = \mathbf{a}]\!] \log\left(\frac{\exp(z_{aa})}{\sum_{k=1}^{K}\exp(z_{ka})}\right) \quad (2)$$

The contrastive action loss $\mathcal{L}_a = \mathcal{L}_{pos} + \mathcal{L}_{neg}$. Here z_{ij} denotes the j-th element in the logits vector z_i, $[\![k = \mathbf{a}]\!]$ denotes the indicator function which returns 1 if k is the same as label $\mathbf{a}$, and 0 otherwise. Note that we do not compute the loss with $\mathbf{p}_{k, i \neq a}$, as ensuring the faithfulness of the target action a is our primary objective here.

Feature-wise loss $\mathcal{L}_{ne}$. To ensure that the visual input regions selected by m_i is sufficient and necessary for the agent to make decisions, we also need to make sure that the unselected region, i.e., $s_{\tilde{m}_i}$, does not provide useful information for action selection, hence the action distribution $\tilde{\mathbf{p}}$ should follow a uniform distribution. Motivated by this rationale, we define negative

Method	*SMB* Acc.	Del. ↓	Ins. ↑	LLE ↓	Sp. ↑	*Enduro* Acc.	Del. ↓	Ins. ↑	LLE ↓	Sp. ↑	*Seaquest* Acc.	Del. ↓	Ins. ↑	LLE ↓	Sp. ↑
LIME	78.7	27.0	30.2	69.4	**98.3**	88.1	36.3	38.0	47.9	**90.2**	90.1	14.3	21.4	17.3	**97.7**
RISE	80.5	**18.9**	38.0	**1.9**	2.0	90.3	37.6	39.1	**0.01**	0.5	92.0	8.1	29.0	**0.02**	1.4
Greydanus	16.9	56.4	20.2	20.8	63.3	27.4	34.6	33.3	0.25	82.9	44.6	14.4	7.88	0.12	75.1
SARFA	16.9	55.6	21.4	68.7	65.5	27.9	33.3	33.4	0.26	77.0	44.8	7.6	8.7	0.13	75.2
Explainer	92.2	23.6	49.0	38.1	81.2	90.2	34.2	33.1	0.22	81.2	95.3	13.4	30.0	0.9	97.5
VisionMask	**95.9**	20.4	**67.6**	38.0	82.3	**98.7**	**32.9**	**41.2**	0.5	80.0	**99.6**	**6.4**	**34.3**	0.9	97.6

TABLE 1: **Quantitative results** on *SMB*, *Enduro* and *Seaquest* of VisionMask against 5 baselines. Five metrics are compared. The faithfulness is measured by Deletion(Del.) and Insertion(Ins.) metrics(%); Robustness is measured by Local Lipschitz Estimate (LLE)(%); And Complexity is measured by Sparseness(Sp.)(%). Blue represents second best results.

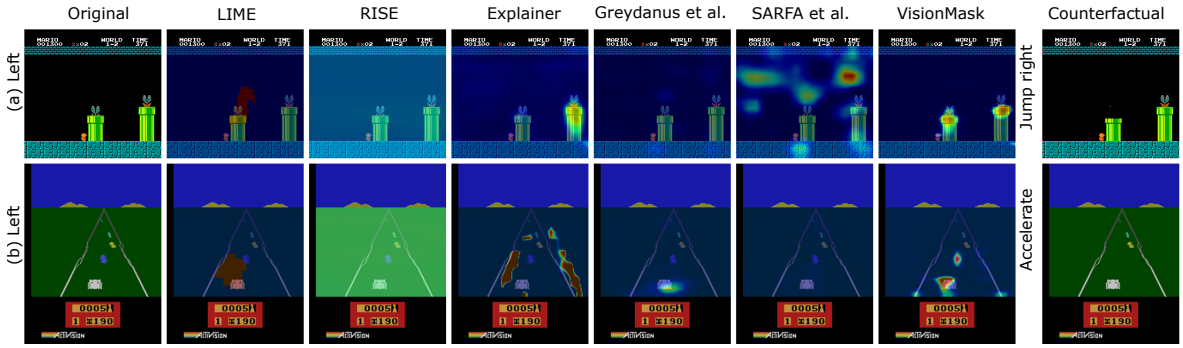

FIGURE 2: **Qualitative examples** of VisionMask and five baselines in three environments. (a) ∼ (b) show the saliency map overlaid on input image and the counterfactual examples where regions in original input are removed based on the saliency map generated by VisionMask. (a) Human explanation: "Mario moves to the left to avoid the Piranha Plant emerging on the pole." VisionMask correctly identified the Piranha Plant. Counterfactual analysis shows that removing the Piranha Plant at the top of the first pipe changes the action from 'Left' to 'Jump right'.

entropy loss regarding $\tilde{\mathbf{m}}$ as the following:

$$\mathcal{L}_{ne}(\mathbf{s}) = \frac{1}{K^2} \sum_{ij} \tilde{\mathbf{p}}_{ij} \log \tilde{\mathbf{p}}_{ij}.$$

Area size loss $\mathcal{L}_{area}$. A low effort way to minimize $\mathcal{L}_a$ and $\mathcal{L}_{ne}$ is to include all pixels in the mask, m_i, and no pixel in the complement mask, $\tilde{m}_i$, which obviously is not a valid solution. We need to ensure that each importance mask only consists a small number of crucial pixels. Thus, we define $\mathcal{L}_{area}$ using L1 norm as follows:

$$\mathcal{L}_{area}(\mathbf{s}) = \frac{1}{K} \sum_{k} (|\frac{1}{Z} \sum_{i,j} m_k[i,j] - a_{max}|)$$

where Z it the number of pixels in state.

Experiments

In this section, we begin by outlining the experimental setup. We then present quantitative and qualitative analyses to evaluate our approach. Additionally, we provide counterfactual explanations to demo VisionMask's faithfulness and sensitivity. **Environment Selection**. We conduct experiments across three types of environments: *Super Mario Bros (SMB)*[11], *Enduro* and *Seaquest*[12]. We mainly compare our model with perturbation-based baselines for black-box RL such as Greydanus [6] and *SARFA*[7]. We presents the quantitative results in Table 1, VisionMask achieves the best faithfulness and the best balance between the robustness and sparseness. In addition, we also compared with three techniques originally designed to explain image classifiers, including a learning-based method, *Explainer*[8], and two perturbation-based methods, LIME[3] and RISE[13]. We present example explanations from three environments, *SMB* and *Enduro* in Figure 2, along with some counterfactual analysis generated from the explanations provided by VisionMask. VisionMask accurately highlights the relevant regions, providing sharp explanations that are both accurate and interpretable.

Conclusion

We presented *VisionMask*, an agent-agnostic DRL explanation model trained in self-supervised contrastive learning. *VisionMask* generates explanations with higher fidelity and better effectiveness compared to existing attribute-based methods.

ACKNOWLEDGMENTS

This research is partially supported by the Air Force Office of Scientific Research (AFOSR), under contract FA9550-24-1-0078, and NSF award CNS-2148253.

The paper was received and approved for public release by AFRL on May 28th 2024, case number AFRL-2024-2908 . Any Opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of AFRL or its contractors.

REFERENCES

1. R. Sutton, A. Barto, undefined. others. "Reinforcement learning," in Journal of Cognitive Neuroscience, vol. 11, no. 1, pp. 126–134, 1999.
2. Hickling, T., et al. "Explainability in Deep Reinforcement Learning: A Review into Current Methods and Applications," in ACM Comput. Surv., vol. 56, no. 5, 2023.
3. M. Ribeiro, S. Singh, C. Guestrin, "" Why should i trust you?" Explaining the predictions of any classifier," in Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.
4. Selvaraju, R., et al, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in Proceedings of the IEEE international conference on computer vision, 2017, pp. 618–626.
5. Bach, S., et al. "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," in PloS one, vol. 10, no. 7, pp. e0130140, 2015.
6. Samuel Greydanus, undefined., et al, "Visualizing and Understanding Atari Agents," in Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, 2018, pp. 1787–1796.
7. Nikaash Puri, undefined., et al, "Explain Your Move: Understanding Agent Actions Using Specific and Relevant Feature Attribution," in 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020, 2020.
8. Stalder, S., et al. "What you see is what you classify: Black box attributions," in Advances in Neural Information Processing Systems, vol. 35, pp. 84–94, 2022.
9. S. Chopra, R. Hadsell, Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005, pp. 539-546 vol. 1.
10. F. Schroff, D. Kalenichenko, J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 815–823.
11. Christian Kauten, . "Super Mario Bros for OpenAI Gym." . (2018).
12. Bellemare, Marc G, Yavar, Naddaf, Joel, Veness, Michael, Bowling. "The arcade learning environment: An evaluation platform for general agents". Journal of Artificial Intelligence Research 47. (2013): 253–279.
13. Vitali Petsiuk, , Abir Das, Kate Saenko. "RISE: Randomized Input Sampling for Explanation of Black-box Models." British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018. BMVA Press, 2018.
14. Chen, L.C., Papandreou, G., Schroff, F., Adam, H. (2017). Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587.