2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Domain Adaptation in Visual Reinforcement Learning via Self-Expert Imitation with Purifying Latent Feature

Lin Chen[1,†], Jianan Huang[1,†], Zhen Zhou[1,†], Yaonan Wang[1], Yang Mo[1], Zhiqiang Miao[1], Kai Zeng[1], Mingtao Feng[2], and Danwei Wang[3], *IEEE Life Fellow*

***Abstract*— Generalizing visual reinforcement learning is fundamental to robot visual navigation, involving the acquisition of a policy from interactions with source environments to facilitate adaptation to analogous, yet unfamiliar target environments. Recent advancements capitalize on data augmentation techniques, self-supervised learning methods, and the generative adversarial network framework to train policy neural networks with enhanced generalizability. However, current methods, upon extracting domain-general latent features, further utilize these features to train the reinforcement learning policy, resulting in a decline in the performance of the learned policy guiding the agent to accomplish tasks. To tackle these challenges, a framework of self-expert imitation with purifying latent features was devised, empowering the policy to achieve robust and stable zero-shot generalization performance in visually similar domains previously unseen, without diminishing the performance of guiding the agent to accomplish tasks. The extraction method of domain-general latent features is proposed to enhance their quality based on the variational autoencoder. Extensive experiments have shown that our policy, compared with state-of-the-art counterparts, does not diminish the performance of the policy guiding the agent to accomplish tasks after generalization.**

I. INTRODUCTION

The application of deep reinforcement learning in vision-related tasks such as object goal navigation [1], [2], visual-language navigation [3], [4], and autonomous driving [5], [6] has attracted increasingly more researchers in recent years. In real-world complex scenarios, image semantic segmentation techniques actively enable accurate segmentation and recognition of the contents within images [7]. Building upon this foundation, extracting color-background-independent generalized features is critical for enhancing the generalization capability of reinforcement learning applications in domains like autonomous driving and visual navigation, while also ensuring optimal performance of the reinforcement learning process. Deep reinforcement learning (DRL) techniques demand extensive interaction with the environment to train policy neural networks. Consequently, a trained policy neural network may falter in a new environmental domain, despite similarities with the training domain. Moreover, minor alterations in the image state at the pixel level can render a trained policy ineffective [8]. These two issues pose significant obstacles to achieving poor generalization when implementing DRL for vision tasks.

Deep reinforcement learning (DRL) methods have incorporated various advanced image domain adaptation techniques to improve their generalization in visual tasks [9]–[11]. Nonetheless, these methods are not without their limitations. Domain randomization, for instance, struggles in one-to-many generalization scenarios and demands multiple similar source domains for effective training [12]–[14]. Image-to-image conversion methods, although effective, can be impractical for real-time applications such as robot visual navigation due to the significant computational overhead required by generator models [15], [16]. In response to these challenges, researchers have adopted encoder-decoder models for mapping image states and potential embeddings, aiming to extract internal representations and boost generalization capabilities [17]. Moreover, data augmentation techniques have been employed to enhance the generalizability of RL policies [18]–[20]. Incorporating data augmentation introduces instability into the generalization performance of RL strategies. Excessive transformations can result in unpredictability, showcasing strong performance in specific scenarios while underperforming in others. LUSR [5] introduces a two-phase reinforcement learning method. Initially, it employs an augmented recurrent consistency VAE approach to distinguish between generic domain and domain-specific feature representations. In the second phase of reinforcement learning training, domain-generic features are utilized to facilitate generalization from the source domain to the target domain without necessitating additional training. However, these methods, after extracting domain-general latent features, proceed to utilize them for training the reinforcement learning policy, leading to a decline in the performance of the learned policy guiding the agent to accomplish tasks.

To tackle these challenges, a framework of self-expert imitation with purifying latent features was devised, empowering the policy to achieve robust and stable zero-shot gener-

This work was supported by the National Key Research and Development Program of China (Grant No. 2022YFB3903804), the National Natural Science Foundation of China (Grant No. 62103141, 62293515, 62027810, 62273138, 62133005, 62373293), the Hunan Provincial Natural Science Foundation of China (2021JC0004, 2022JJ40095), and the Jiangxi Province 03 Special Project and 5g Project (20232ABC03A09).

[1]Lin Chen, Jianan Huang, Zhen Zhou, Yaonan Wang, Yang Mo, Zhiqiang Miao, and Kai Zeng are with the School of Electrical and Information Engineering, Hunan University, Changsha, 410082, China, and also with the National Engineering Research Center for Robot Visual Perception and Control Technology, Changsha 410082, China. (email: chenlin21@hnu.edu.cn; jiananhuang@hnu.edu.cn; zhenzhou@hnu.edu.cn; yaonan@hnu.edu.cn; moyanghnu@hnu.edu.cn; miaozhiqiang@hnu.edu.cn; zkwalt@hnu.edu.cn;) (Corresponding author: Yang Mo)

[2]Mingtao Feng is with the School of Computer Science and Technology, Xidian University, Xian, 710126, China. (email: mintfeng@hnu.edu.cn)

[3]Danwei Wang is with School of Electrical and Electrical Engineering, Nanyang Technological University, Nanyang Avenue, 639798, Singapore. (email: EDWWANG@ntu.edu.sg)

[†]These authors contributed equally.

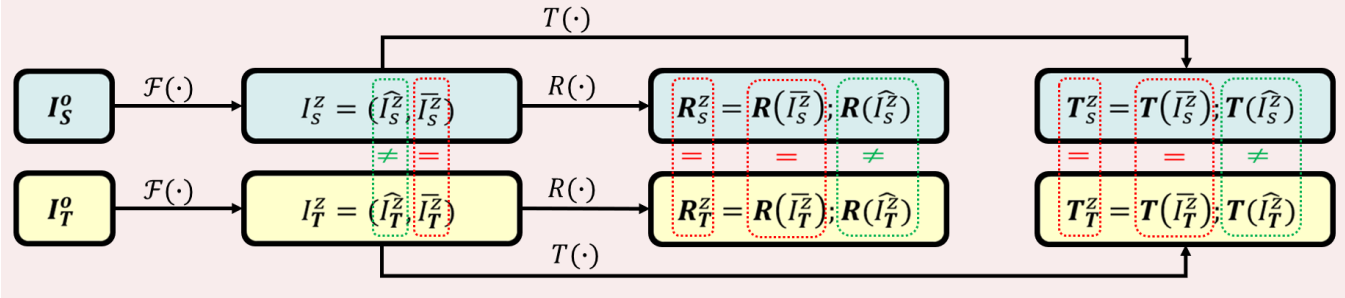


Fig. 1: The illustration of feature separation, as well as the relational diagrams depicting the relationships among various variables in Equations 1 and 2.

alization performance in visually similar domains previously unseen, without diminishing the performance of guiding the agent to accomplish tasks. Initially, we formulate a purified latent feature approach using the variational autoencoder (VAE) [21] to isolate domain-general embeddings, representing information shared across all domains. Concurrently, a reinforcement learning methodology is employed to optimize the performance of the expert strategy neural network in the source domain. Subsequently, we introduce a self-expert imitation framework designed to correlate expert actions with their respective generic domain embeddings.

The main contributions of this work are summarized as follows:

1) A self-expert imitation method with purifying domain-general latent features for robot visual navigation is presented, enabling the acquisition of a policy from interactions with source environments to facilitate adaptation to analogous, yet unfamiliar target environments.

2) A framework for separating domain-general and domain-specific features, based on variational autoencoder, is developed, enhancing the quality of domain-general latent features. After feature purification, image augmentation samples are no longer required in training, resulting in enhanced stability of the generalization performance of the trained policy.

3) The experimental results demonstrate that our approach outperforms state-of-the-art performance in the CarRacing game and that the trained policy does not lead to a decline in the performance of guiding the agent to accomplish tasks.

II. PROPOSED METHOD

In this section, we present self-imitation with purified latent features, a learning framework designed to address the generalization challenges faced by reinforcement learning in vision tasks. Initially, we formally define the problem. Specifically, image state features are categorized into domain-specific and domain-general components. To purify the domain-general features, we employ a method based on a variational autoencoder. Furthermore, we elaborate on the self-expert imitation technique, which is developed to mitigate policy performance degradation following feature separation.

A. Problem Definition

In reinforcement learning vision tasks, we denote the raw observation image and internal latent state as I^o and I^z, respectively. The raw observation image state I^o comprises the RGB values for each pixel. A neural network is employed to model a non-linear mapping function, $\mathcal{F} : I^o \rightarrow I^z$, which transforms the observation image state into the corresponding internal latent state. The internal latent state I^z is further decomposed into domain-specific features $\widehat{I^z}$ and domain-general state $\overline{I^z}$, represented as $I^z = \left(\widehat{I^z}, \overline{I^z}\right)$. The relationship between the image states in the source and target domains can be summarized as follows:

$$\begin{gathered} I_S^o \neq I_T^o \\ I_S^z = \left(\widehat{I_S^z}, \overline{I_S^z}\right); \quad I_T^z = \left(\widehat{I_T^z}, \overline{I_T^z}\right) \\ \overline{I_S^z} = \overline{I_T^z}; \quad \widehat{I_S^z} \neq \widehat{I_T^z} \end{gathered} \tag{1}$$

where I_S^o and I_T^o represent the image states in the source and target domains, respectively. In this study, the transition function T and reward function R align with those defined in LUSR [5], both of which depend solely on the domain-general state $\overline{I^z}$. The reward and transition functions for the source and target domains, when considering the internal latent image state, are interrelated as follows:

$$\begin{gathered} R_S^z = R\left(\overline{I_S^z}\right) = R\left(\overline{I_T^z}\right) = R_T^z \\ T_S^z = T\left(\overline{I_S^z}\right) = T\left(\overline{I_T^z}\right) = T_T^z \\ T\left(\widehat{I_S^z}\right) \neq T\left(\widehat{I_T^z}\right); \quad R\left(\widehat{I_S^z}\right) \neq R\left(\widehat{I_T^z}\right), \end{gathered} \tag{2}$$

where R_S^z and R_T^z denote the reward values in the source and target internal latent image states, respectively. Inspired by [22], graphs have been employed to provide a more lucid depiction of the relationships among various variables, as illustrated in Figure 1. As shown in Equation (2), reinforcement learning and imitation learning methodologies are applied using the state $\overline{I^z}$ for training the policy. This approach facilitates the adaptation of the policy from the source to the target domain. The primary objective of this work is to enable policy adaptation across both source and target domains while maintaining general reward scores. The essence of this research lies in learning and acquiring the mapping function $\mathcal{F} : I^o \rightarrow \overline{I^z}$.

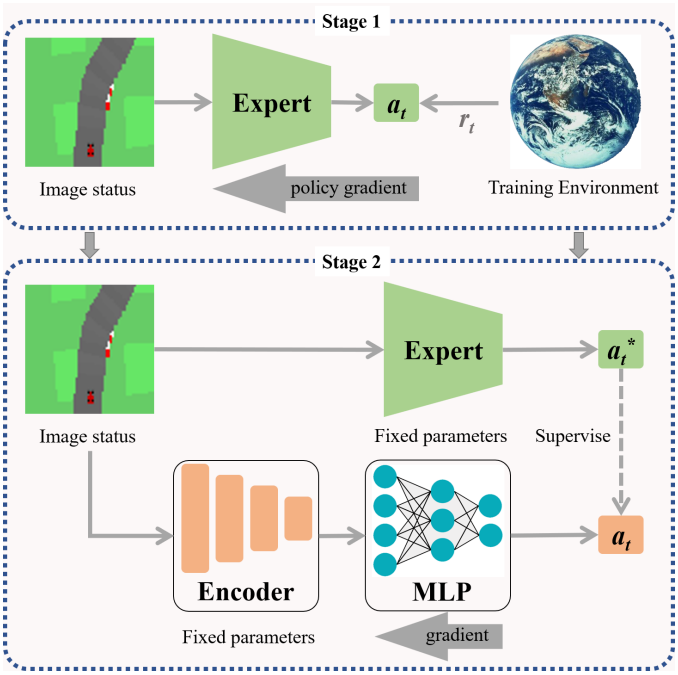


Fig. 2: The architecture diagram of self-expert imitation with reinforcement learning. The process comprises two distinct stages. Initially, a high-performance expert policy is derived through reinforcement learning training. Subsequently, self-expert imitation learning is leveraged to ensure policy generalization across various domains. MLP denotes a multi-layer perceptron policy neural network.

B. Self-expert Imitation with RL

The proposed self-expert imitation technique, incorporating a latent feature purification mechanism, seeks to develop a policy that can generalize from the source domain to various target domains without a loss in performance. However, a significant challenge lies in optimizing the dimensions of the domain-general vectors to ensure the policy's performance remains unaffected.

As depicted in Fig. 2, we introduce a two-stage self-expert imitation training methodology to address this challenge. In the first stage, a high-performance expert policy is trained within an interactive environment. At this juncture, the expert neural network is confined to training within the original domain, without regard for its generalization capabilities. The expert policy is optimized to achieve peak performance by minimizing the PPO objectives function through gradient descent. In the second stage, the expert action a_t^* is translated to a domain-general state $\overline{I_t^z}$, associated with the image state I_t^o, thereby facilitating policy cross-domain generalization. During this stage, the reward signal is excluded from the training process of the multilayer perceptron (MLP) policy network. Initially, the expert policy is employed to compile a training dataset $\mathcal{D}$ of trajectories. Subsequently, as illustrated in Fig. 2, the parameters of the expert policy neural network and the encoder network are held constant while the MLP is updated via gradient descent on a supervised regression loss: $|\pi_m(\mathcal{F}(I_t^o)) - \pi_e(I_t^o)|$. In summary, the proposed self-expert imitation with purifying latent feature approach not only facilitates cross-domain generalization of the policy but also enhances its overall performance.

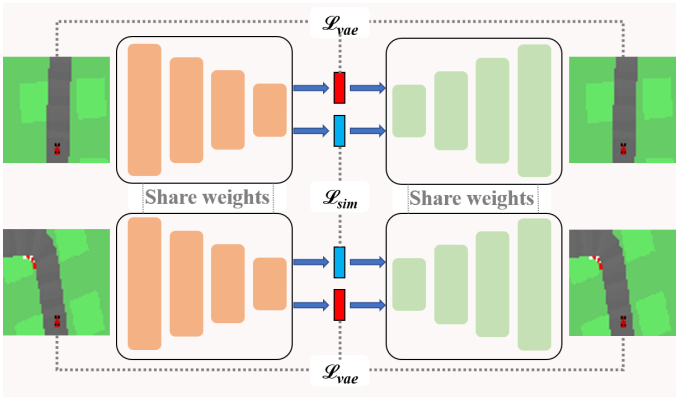


Fig. 3: The architecture of feature purification with VAE. The orange piece in the figure represents the encoder, and the green part is the decoder. The blue and red squares represent the domain-general and domain-specific vectors, respectively.

C. Feature Purification with VAE

We introduce a novel approach utilizing Variational Autoencoder (VAE) [21] to learn the mapping function $\mathcal{F}: I^o \rightarrow \overline{I^z}$. As discussed in Section II-A, a key objective is to delineate domain-specific states $\widehat{I^z}$ from domain-general features $\overline{I^z}$. Randomly selected observation states from predefined fields serve as VAE inputs. The VAE encodes the image observation states into a latent vector that preserves spatial continuity. Through the encoder Enc, the image information is transformed into a Gaussian distribution. Sampling within this feature space allows the decoder Dec to reconstruct an image of identical dimensions to the input. The latent vector z^z comprises both $\widehat{z^z}$ and $\overline{z^z}$ in our methodology. Upon training completion, the encoder functions as the mapping $\mathcal{F}$, exclusively retaining the domain-general feature $\overline{z^z}$.

In this study, we define the forward calculation as $Dec\,(Enc\,(z^o)) = z^o\prime$, which leverages the characteristics of the VAE's encoder and decoder. Here, z^o represents the raw observation image state. Upon encoding, we obtain the latent vector $\left(\widehat{z^z}, \overline{z^z}\right)$, expressed as $Enc\,(z^o) = \left(\widehat{z^z}, \overline{z^z}\right)$. When this latent vector is decoded, it transforms into a reconstructed image $z^o\prime$, denoted as $Dec\left(\widehat{z^z}, \overline{z^z}\right) = z^o\prime$. As illustrated in Fig.3, two image states, z_1^o and z_2^o, from the same domain are encoded into $\left(\widehat{z_1^z}, \overline{z_1^z}\right)$ and $\left(\widehat{z_2^z}, \overline{z_2^z}\right)$, respectively, which can be formulated as $Enc\,(z_1^o) = \left(\widehat{z_1^z}, \overline{z_1^z}\right)$ and $Enc\,(z_2^o) = \left(\widehat{z_2^z}, \overline{z_2^z}\right)$. Since the input observation image states originate from the same domain, we can infer $\widehat{z_1^z} \approx \widehat{z_2^z}$. Furthermore, swapping $\widehat{z_1^z}$ and $\widehat{z_2^z}$ should not affect the reconstruction outcome, leading to $Dec\left(\widehat{z_2^z}, \overline{z_1^z}\right) \approx z_1^o$ and $Dec\left(\widehat{z_1^z}, \overline{z_2^z}\right) \approx z_2^o$. Given that the image states belong to

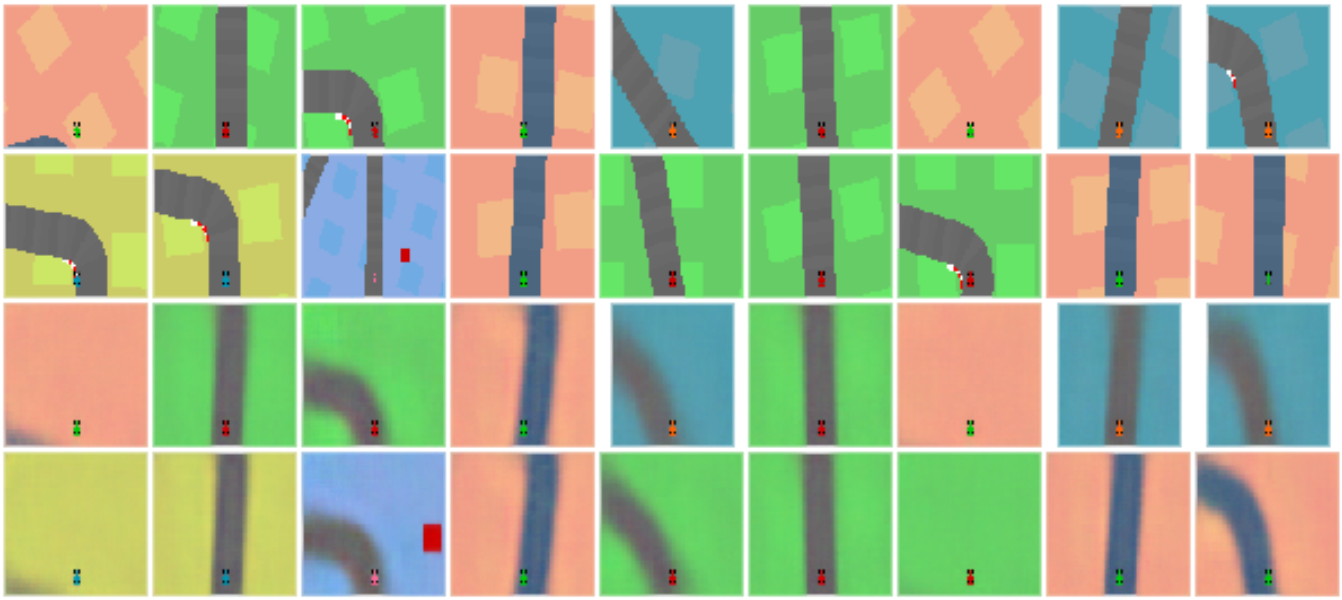


Fig. 4: The resulting image of domain-general and domain-specific features is purified and reconstructed. The trained encoder extracts the domain-specific and domain-general features of the first and second row images , which are $\widehat{Z_1^z}$, $\overline{Z_1^z}$, $\widehat{Z_2^z}$ and $\overline{Z_2^z}$, respectively. The image shown in the third row is reconstructed using the learned decoder for the domain-specific features $\widehat{Z_1^z}$ and the domain-general embeddings $\overline{Z_1^z}$ of the first row. The fourth row of images is reconstructed based on the first row's domain-general features $\overline{Z_1^z}$ and the second row's domain-specific embeddings $\widehat{Z_2^z}$.

the same domain, we consider the domain-specific vectors $\widehat{z_1^z}$ and $\widehat{z_2^z}$ to be equivalent in this work. Consequently, the similarity between these vectors is harnessed to design the following loss function:

$$\mathcal{L}_{sim} = \frac{-1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} 1_{i \neq j} \cdot \frac{\widehat{z_i^z}}{\left\| \widehat{z_i^z} \right\|} \cdot \frac{\widehat{z_j^z}}{\left\| \widehat{z_j^z} \right\|}, \tag{3}$$

where N denotes the number of samples within a given batch. The term $1_{i \neq j}$ evaluates to 1 if i and j are distinct, and 0 otherwise.

In the context of this study, images are reconstructed utilizing a Variational Autoencoder (VAE). The VAE loss function comprises two distinct components. Initially, the mean squared error (MSE) quantifies the image reconstruction error. Subsequently, the Kullback-Leibler (KL) divergence [23] assesses the disparity between the latent variable distribution and the standard Gaussian distribution. The reconstruction error loss for the image state is mathematically represented as follows:

$$\mathcal{L}_{MSE} = \frac{1}{N} \cdot \sum_{i=1}^{N} \left(z_i^o - z_i^o\prime \right)^2, \tag{4}$$

where z_i^o and $z_i^o\prime$ denote the input state and the reconstructed image, respectively. The loss function for domain-general vectors is expressed as

$$D_{KL} = -\frac{1}{2N} \cdot \sum_{i=1}^{N} \cdot \left(1 + \log \left(\sigma^2_{\overline{z_i^z}} \right) - \mu^2_{\overline{z_i^z}} - \sigma^2_{\overline{z_i^z}} \right), \tag{5}$$

where $\mu_{\overline{z_i^z}}$ and $\sigma_{\overline{z_i^z}}$ denote the mean and variance of the domain-general vectors $\overline{z_i^z}$, respectively. Consequently, the weighted sum of the individual component losses constitutes the final loss function. The ultimate objective is to minimize this function,

$$\mathcal{L} = \alpha \cdot \mathcal{L}_{sim} + \beta \cdot D_{KL} + \mathcal{L}_{MSE}, \tag{6}$$

where α $(0 < \alpha < 1)$ and β $(0 < \beta < 1)$ are weight parameters. Upon completion of the learning process, the mapping function $\mathcal{F} : I^o \rightarrow \overline{I^z}$ can be obtained.

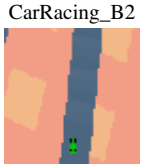
III. EXPERIMENTS

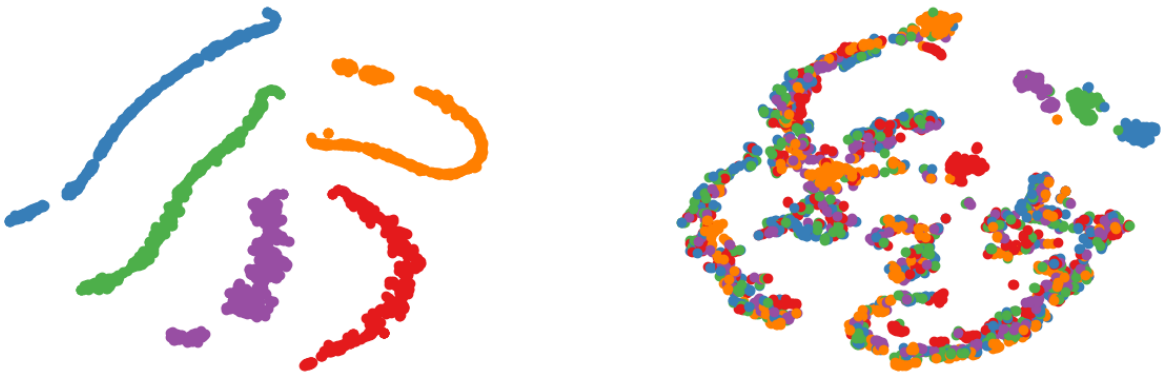
In this section, we present a comprehensive evaluation of our approach using a series of CarRacing game variants, benchmarked against existing methodologies. Initially, we outline the experimental setup to illustrate the effectiveness of our proposed technique. Subsequently, we detail the performance outcomes of the algorithm. Finally, we construct a dataset employing the AdaIN method [24] to assess the robustness of the extracted domain-general features.

A. Experiment Setting

The objective of CarRacing games is to maintain the car's movement on the track by controlling physical quantities, which include direction, throttle, and brake. The feedback from the environment is a 64×64×3 top-down view of the car. The PPO approach is utilized to update the policy through continuous interaction with the CarRacing game environment. The learned policy directs the car along the road, with the final score serving as a performance indicator for the policy. All environments are classified into the source domain, seen target domains, and unseen target domains, similar to LUSR [5], to implement the proposed method.

TABLE I: Our proposed method applies reinforcement learning to adapt performance in other domains in CarRacing games.

Approach	Source Domain	Seen Target Domains				Unseen Target Domains	
	CarRacing_A1	CarRacing_B1	CarRacing_B2	CarRacing_B3	CarRacing_B4	CarRacing_C1	CarRacing_C2
							
	Score	Score(Ratio)	Score(Ratio)	Score(Ratio)	Score(Ratio)	Score(Ratio)	Score(Ratio)
Ours	940.39	929.38 (0.99)	901.53 (0.96)	959.94 (1.02)	917.32 (0.97)	956.33 (1.01)	935.89 (0.99)
CDCBA [11]	862.88	885.74 (1.03)	859.14 (1.00)	882.56 (1.02)	854.38 (0.99)	857.99 (0.99)	814.38 (0.94)
Ad-DISR [9]	853.99	871.58 (1.02)	870.41 (1.02)	854.52 (1.00)	874.98 (1.02)	863.45 (1.01)	872.20 (1.02)
LUSR [5]	805.13	803.52 (1.00)	807.37 (1.00)	803.11 (1.00)	781.94 (0.97)	678.7 (0.84)	800.56 (0.99)



(a) (b)

Fig. 5: The t-SNE plot of domain-general and domain-specific features from five CarRacing games with different backgrounds. Subgraphs (a) and (b) are domain-specific and domain-general embeddings t-SNE diagrams, respectively.

As indicated in Table I, CarRacing_A1 represents the source domain. The seen target domains include CarRacing_B1, CarRacing_B2, CarRacing_B3, and CarRacing_B4, while the unseen target domains comprise CarRacing_C1 and CarRacing_C2. The image states generated by environments CarRacing_B3 and CarRacing_C1 display narrower characteristics compared to those from other environments. A total of 500,000 images were collected from the source and seen target domains, with each domain contributing 100,000 images. During this experimental process, we used the same environment as LUSR to train an expert policy and gather offline data for imitation learning. The relevant experiments were conducted using LUSR's publicly available code and offline dataset[1]. Approximately 6000 epochs were dedicated to the reinforcement learning training process. The gamma value was set at 0.99, the learning rate was configured to 0.0002, the clip parameter was established at 0.4, and the number of training iterations was set to 10.

Initially, we employ the method delineated in Section II-C to achieve feature separation from the collected data. In this approach, the trained encoder serves as the mapping function, establishing a correspondence between each domain's image and the domain-general vector. Concurrently, the PPO algorithm is utilized to learn an expert policy within the source domain environment. Subsequently, our proposed self-expert imitation learning framework is applied to map the expert action to the domain-general state in the source domain. Lastly, we execute the trained policy across both seen and unseen target domains to evaluate its generalization performance.

B. Results and Discussion

To assess the efficacy of the domain-general and domain-specific vectors produced by the mapping function $\mathcal{F}$, we randomly selected samples from both the source domain and the visible target domain to compute these vectors. Subsequently, these vectors underwent dimensionality reduction using t-SNE. As depicted in fig.5, the trained encoder effectively distinguishes between the domain-general and domain-specific features that characterize the image state.

[1]https://github.com/KarlXing/LUSR

TABLE II: Our proposed method is compared with the method without domain-general feature extraction.

Domain	Ours	DARLA (β = 10)	DARLA(β = 30)	DARLA(β = 100)	VAE-Embedding	CURL	CycleGAN
CarRacing_A1	940.39	845.87	851.48	778.78	816.74	748.58	709.12
CarRacing_B1	929.38	645.81	834.99	704.27	616.89	560.23	707.64
CarRacing_B2	901.53	250.85	819.05	207.90	282.71	-44.24	704.33
CarRacing_B3	959.94	-72.99	-60.76	451.81	484.57	-55.29	713.86
CarRacing_B4	917.32	-62.96	-73.18	27.77	223.88	-32.45	711.85
CarRacing_C1	956.33	-65.72	-72.55	182.35	332.58	-113.13	715.43
CarRacing_C2	935.89	631.09	806.76	539.63	595.42	-69.23	671.67

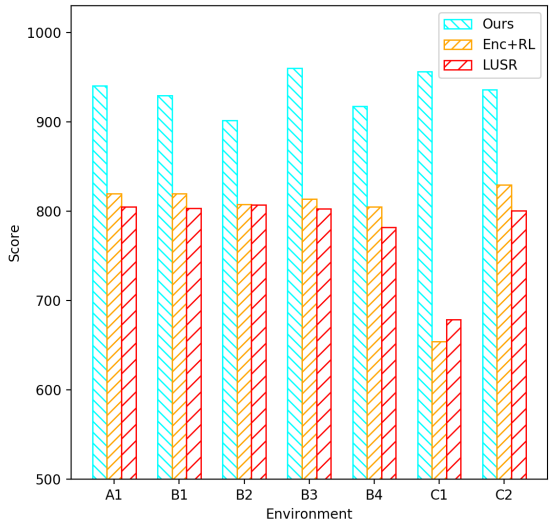


Fig. 6: Histogram of final scores for policies trained with different methods. The horizontal axis represents seven different CarRacing game domains.

Furthermore, as illustrated in fig.4, the encoder, trained via our proposed method, adeptly isolates domain-general embeddings from the images of each domain.

We present the generalization performance of a policy trained using the self-expert imitation technique proposed in this paper across seven distinct domains. Our approach is compared with previous methods, including Ad-DISR [9], LUSR [5], DARLA [17], and CURL [25]. As indicated in Table I, our method achieves superior scores and generalizes effectively to both seen and unseen domains. For CDCBA [11], LUSR [5], and Ad-DISR [9], these methods primarily focus on extracting domain-general features to facilitate policy generalization. However, determining the optimal feature size remains challenging, leading to lower policy scores compared to the current optimal reinforcement learning scores in the source domain. As depicted in Table II, the DARLA method struggles to ascertain the optimal value of the parameter β, resulting in a relatively weak policy generalization capability. Additionally, the VAE-Embedding method exhibits significant performance drops and instability in both seen and unseen domains. The training data for the DARLA and VAE-Embedding methods are analogous to images obtained via data augmentation techniques applied to Source Domain images. Experimental results highlight that the stability of generalization performance in these two methods is compromised. This instability stems from the different features learned by the RL policy across various training batches, influenced by data augmentation. Excessive transformations induced by data augmentation can lead the RL policy to excel in certain scenarios while underperforming in others, making it challenging to predict the policy's generalization performance. Although CycleGAN demonstrates commendable generalization performance, its policy scores are lower than those achieved by our method. This discrepancy may be attributed to the loss of some domain-general information in its latent embeddings.

In this work, we introduced a domain-general feature extraction approach leveraging Variational Autoencoders (VAE) within the self-expert imitation learning framework. To evaluate its efficacy, we implemented the PPO algorithm to learn domain-general embeddings directly in the source domain. The outcomes, illustrated in fig.6, reveal that our feature extraction method consistently surpasses LUSR across both seen and unseen domains. Following domain-general feature extraction, LUSR was also trained using the PPO algorithm. As depicted in fig.6, our method demonstrates superior performance compared to LUSR in a majority of domains. Experimental findings indicate that the self-expert imitation learning framework significantly enhances the final policy's performance in analogous unseen domains.

IV. Conclusion

In this paper, we propose a self-expert imitation method with purifying domain-general latent features for robot visual navigation, enabling the acquisition of a policy from interactions with source environments to facilitate adaptation to analogous, yet unfamiliar target environments. Furthermore, a framework for separating domain-general and domain-specific features, based on the variational autoencoder, is proposed to enhance the quality of domain-general features. Extensive experiments were conducted in the CarRacing game environment, demonstrating that our policy, after training, did not diminish the performance of guiding the agent to

accomplish tasks and outperformed state-of-the-art counterparts. In our forthcoming research, the focus is on crafting a highly versatile approach that eliminates the necessity of data collection during the initial training phase. Furthermore, the integration of semantic segmentation is being investigated to facilitate the application of our method in the domains of robot visual navigation and autonomous driving.

REFERENCES

[1] H. Du, X. Yu, and L. Zheng, "Vtnet: Visual transformer network for object goal navigation," *arXiv preprint arXiv:2105.09447*, 2021.

[2] Z. Rao, Y. Wu, Z. Yang, W. Zhang, S. Lu, W. Lu, and Z. Zha, "Visual navigation with multiple goals based on deep reinforcement learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 12, pp. 5445–5455, 2021.

[3] B. Lin, Y. Zhu, Y. Long, X. Liang, Q. Ye, and L. Lin, "Adversarial reinforced instruction attacker for robust vision-language navigation," *arXiv preprint arXiv:2107.11252*, 2021.

[4] T. Wang, Z. Wu, and D. Wang, "Visual perception generalization for vision-and-language navigation via meta-learning," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–7, 2021.

[5] J. Xing, T. Nagata, K. Chen, X. Zou, E. Neftci, and J. L. Krichmar, "Domain adaptation in reinforcement learning via latent unified state representation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 12, 2021, pp. 10 452–10 459.

[6] Y. Tang, C. Zhao, J. Wang, C. Zhang, Q. Sun, W. X. Zheng, W. Du, F. Qian, and J. Kurths, "Perception and navigation in autonomous systems in the era of learning: A survey," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21, 2022.

[7] S. K. Panda, Y. Lee, and M. K. Jawed, "Agronav: Autonomous navigation framework for agricultural robots and vehicles using semantic segmentation and semantic line detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 6271–6280.

[8] S. Gamrian and Y. Goldberg, "Transfer learning for related reinforcement learning tasks via image-to-image translation," in *International conference on machine learning*. PMLR, 2019, pp. 2063–2072.

[9] D. Li, L. Meng, J. Li, K. Lu, and Y. Yang, "Domain adaptive state representation alignment for reinforcement learning," *Information Sciences*, vol. 609, pp. 1353–1368, 2022.

[10] L. Fan, G. Wang, D.-A. Huang, Z. Yu, L. Fei-Fei, Y. Zhu, and A. Anandkumar, "Secant: Self-expert cloning for zero-shot generalization of visual policies," *arXiv preprint arXiv:2106.09678*, 2021.

[11] L. Meng, J. Li, and K. Lu, "Cross-domain communications between agents via adversarial-based domain adaptation in reinforcement learning," in *ICC 2022-IEEE International Conference on Communications*. IEEE, 2022, pp. 413–418.

[12] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, pp. 23–30.

[13] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray *et al.*, "Learning dexterous in-hand manipulation," *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 3–20, 2020.

[14] R. B. Slaoui, W. R. Clements, J. N. Foerster, and S. Toth, "Robust domain randomization for reinforcement learning," 2019.

[15] X. Pan, Y. You, Z. Wang, and C. Lu, "Virtual to real reinforcement learning for autonomous driving," *arXiv preprint arXiv:1704.03952*, 2017.

[16] E. Tzeng, C. Devin, J. Hoffman, C. Finn, P. Abbeel, S. Levine, K. Saenko, and T. Darrell, "Adapting deep visuomotor representations with weak pairwise constraints," in *Algorithmic Foundations of Robotics XII*. Springer, 2020, pp. 688–703.

[17] I. Higgins, A. Pal, A. Rusu, L. Matthey, C. Burgess, A. Pritzel, M. Botvinick, C. Blundell, and A. Lerchner, "Darla: Improving zero-shot transfer in reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1480–1490.

[18] A. Zhang, R. T. McAllister, R. Calandra, Y. Gal, and S. Levine, "Learning invariant representations for reinforcement learning without reconstruction," in *International Conference on Learning Representations*, 2020.

[19] R. Raileanu, M. Goldstein, D. Yarats, I. Kostrikov, and R. Fergus, "Automatic data augmentation for generalization in reinforcement learning," in *Advances in Neural Information Processing Systems*, 2021.

[20] M. Tomar, A. Zhang, R. Calandra, M. E. Taylor, and J. Pineau, "Model-invariant state abstractions for model-based reinforcement learning," *arXiv preprint arXiv:2102.09850*, 2021.

[21] J. Rocca, "Understanding variational autoencoders (vaes)," *Data Science*, 2019.

[22] I. Bica, D. Jarrett, and M. van der Schaar, "Invariant causal imitation learning for generalizable policies," *Advances in Neural Information Processing Systems*, vol. 34, pp. 3952–3964, 2021.

[23] S. Kullback, *Information theory and statistics*. Courier Corporation, 1997.

[24] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," *Advances in neural information processing systems*, vol. 31, 2018.

[25] M. Laskin, A. Srinivas, and P. Abbeel, "Curl: Contrastive unsupervised representations for reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 5639–5650.