2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Effective Representation Learning is More Effective in Reinforcement Learning than You Think

Jiawei Zheng[1] and Yonghong Song[2]

Abstract— **In reinforcement learning (RL), learning directly from pixels, is commonly known as vision-based RL. Effective state representations are crucial for high performance in vision-based RL. However, in order to learn effective state representations, most current vision-based RL methods based on contrastive unsupervised learning use auxiliary tasks similar to those in computer vision, which does not guarantee the effective information interaction between representation learning and RL. To learn more efficient states, we propose a simple and effective vision-based RL method. It leverages the representations acquired through contrastive learning by the Teacher Encoder and the Student Encoder to collaboratively estimate the Q-function. This cooperative process utilizes the TD error to steer updates to the Teacher Encoder, thereby ensuring effective information exchange between representation learning and RL. We refer to this approach as Reinforcement Learning with Teacher-Student Collaboration (RLTSC). RLTSC incorporates recent advancements in contrastive unsupervised learning, endowing it with potent representation learning capabilities. It provides a robust estimate of the Q-function with minimal variance and effectively guides the Teacher Ecoder to update and acquire a more efficient representation. RLTSC substantially enhances data efficiency in vision-based RL, surpassing state-of-the-art methods on various continuous and discrete control benchmarks. Remarkably, RLTSC even outperforms RL methods based on physical state features in terms of data efficiency for continuous control benchmarks. This may enlighten us: effective representation learning is more effective in reinforcement learning than you think!**

I. INTRODUCTION

Learning from visual observation is a fundamental problem in reinforcement learning (RL), such as in DeepMind Control Suite (DMControl) [1], Atari games [2], etc. Currently, by combining the expressive ability of deep learning networks with RL algorithms, it is possible to train an agent that obtains environmental information from high-dimensional observations and performs complex control tasks. However, it is generally accepted that learning policies from physical state-based features is significantly more effective than learning from pixels [1].

Therefore, many recent works have been devoted to obtaining a good state representation, which is believed to be one of the key solutions to improve the efficacy of vision-based RL [3]. Most of these efforts focus on two areas: (i) Some works have studied how vision-based RL can use data augmentation to improve the data efficiency of algorithms [4], [5], [6], [7]. (ii) Inspired by contrastive unsupervised learning in the field of computer vision, some works utilize unsupervised learning, using unsupervised contrastive loss and RL loss to update model parameters in a batch [8], [9], [10], [11], [12], [13], [14]. These works generally design some auxiliary tasks to add additional loss functions to enable the model to learn better representations.

For the second type of model, we found that most of the current RL methods based on unsupervised learning use auxiliary tasks similar to those in computer vision. This does not interact well with RL algorithms, and auxiliary tasks for high-dimensional pixels usually introduce additional hyperparameters and use a lot of hardware resources. Hence, we regard this approach as suboptimal.

Based on the above analysis, we investigate unsupervised auxiliary tasks that directly interact information with RL algorithms. We propose Reinforcement Learning with Teacher-Student Collaboration (RLTSC). It is a simple and effective contrastive unsupervised method, which leverages the representations acquired through contrastive learning by the Teacher Encoder and the Student Encoder to collaboratively estimate the Q-function, and uses TD error to guide the update of the Teacher. This method can well enable information interaction between contrastive learning and RL algorithms, and enhance the correlation between learned representations and RL objectives. We use CURL [11] as the baseline to implement the method. We demonstrate the effectiveness of our approach in multiple environments on the continuous control benchmark DMControl and the discrete control benchmark Atari.

Our contributions are summarized as follows:

- We introduce a novel idea to vision-based RL: Teacher-Student collaboration(TSC). The comparison between the Student Encoder and the Teacher encoder in contrastive learning is not only based on the similarity measure of image representation learning, but also involves the utilization of the representations they have acquired to jointly estimate the Q-function. This idea can effectively improve the sample efficiency of the algorithm and enable the RL algorithm to learn robustly on images.
- To enhance the representation learning of the vision-based RL algorithm, we have integrated recent advances in contrastive unsupervised learning into the baseline and introduced the TSC idea. This allows the algorithm to better extract high-level features from raw pixels using contrastive unsupervised learning while gaining

This work is supported by the National Natural Science Foundation of China under Grant 62350083 and the Key R&D Plan of Shaanxi Province (Program No.2023-YBGY-029).

[1]Jiawei Zheng, master student in the School of Software Engineering, Xi'an Jiaotong University, Shaanxi, China JaweiZheng@stu.xjtu.edu.cn

[2]Yonghong Song, professor in the School of Software Engineering, Xi'an Jiaotong University, Shaanxi, China songyh@xjtu.edu.cn

the advantages of the TSC idea. We refer to this approach as Reinforcement Learning with Teacher-Student Collaboration (RLTSC).

- Experimental confirmation studies investigate the strong performance of RLTSC, demonstrating the effectiveness of our enhancements in both the contrastive unsupervised learning component and the TSC idea. The RLTSC outperforms **state-of-the-art model-free methods** in multiple environments of the continuous control benchmark DMControl, and converges faster and **better than physical state-based RL**. The RLTSC outperforms **state-of-the-art model-free methods** in multiple environments of the discrete control benchmark Atari.

II. The Preamble

In this section, we give a brief overview of the classic actor-critic approach Soft Actor Critic(SAC) [15], the development of contrastive learning, and the application of representation learning to vision-based RL.

A. Soft Actor-Critic

SAC is a relatively stable off-policy reinforcement learning algorithm and a maximum entropy reinforcement learning algorithm. The idea of maximum entropy RL is not only to maximize the cumulative reward, but also to make the strategy more random. This keeps the agent from falling into sub-optimality by repeatedly choosing the same action. In this way, a regular term of entropy is added to the goal of RL.

SAC learns two critic networks Q_{ω_1}, Q_{ω_2} and a policy network π_θ. SAC updates the critic network by minimizing the Bellman error. Assuming that N tuples $\{(s_i,a_i,r_i,s_{i+1})\}_{i=1,\ldots,N}$ are sampled in the ReplayBuffer R, α is an entropy regularization coefficient, SAC uses two target networks $Q_{\omega_1^-}$, $Q_{\omega_2^-}$ to calculate the target $y_i = r_i + \gamma min_{j=1,2} Q_{\omega_j^-}(s_{i+1},\tilde{a}_{i+1}) - \alpha log\pi_\theta(a_{i+1}|s_{i+1})$, where $a_{i+1} \sim \pi_\theta(.|s_{i+1})$. Then calculate the loss function to update the two critic networks Q_{ω_1}, Q_{ω_2} : $L_j = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - Q_{\omega_j}(s_i,a_i)\right)^2$. Afterwards, SAC samples action $\tilde{a}_i$ with a reparameterization trick, and then the following loss function updates the current actor network: $L_\pi(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left(\alpha log\pi_\theta(\tilde{a}_i|s_i) - \min_{j=1,2} Q_{\omega_j}(s_i,\tilde{a}_i)\right)$.Finally, SAC automatically adjusts the entropy regularization coefficient α and updates the target network.

B. Contrastive Learning

Contrastive learning is a framework for training a model to distinguish between similar and dissimilar images in a set of datasets. This can be understood as the task of performing dictionary lookups, where positive and negative examples represent a set of keys related to the query [16]. CURL is a landmark work combining vision-based RL with contrastive learning. It combines the RL algorithm SAC with the contrastive learning method MoCo [16] to build an unsupervised representation framework that can be used for RL. Since then, the rapid development of BYOL [17]-style contrastive learning methods has also profoundly affected the application of contrastive learning in RL. In order to improve the representation learning ability of the model, some works [18], [8] have added BYOL-style improvements to the contrastive unsupervised representation framework proposed by CURL.

C. Representation Learning for RL

Vision-based RL has high use value in real-world applications such as robotics, autonomous driving, and game AI. However, high-dimensional visual information often contains noise or redundancy, which brings great difficulties to RL agents to learn effective representations [19].

In order for RL agents to learn effective representations, a great deal of research has been devoted to developing various methods. Some works jointly learn through RL loss and auxiliary tasks to provide additional representational supervision. These tasks include pixel reconstruction [8], [19], [20], contrastive learning of instance recognition [11], reward prediction [21], [19], etc. However, the ideas of these works mostly overlap with many existing ideas, including: reconstruction after mask [14], [8], prediction with dynamic model [19], [22], etc. Few works combine RL algorithms with auxiliary tasks of contrastive learning. The biggest innovation of this study is to let TD error guide the contrastive learning. We use this auxiliary task for contrastive learning and RL information interaction.

III. Method

In this section, we use continuous control benchmarks as an example to briefly describe the implementation details of the RLTSC. Firstly, we analyzed the effectiveness of TSC ideas. Subsequently, we further enhance the RLTSC algorithm by integrating the latest advances in contrastive unsupervised learning. An overview of the RLTSC's architecture is shown in Fig. 1.

A. Optimality Invariance for Q-function

Data augmentation, as a data-driven approach, has demonstrated significant potential for vision-based RL in terms of both sample efficiency and generalization ability. We define a general augmentation f : $\mathscr{O} \times \mathscr{V} \rightarrow \mathscr{O}^{aug}$ as a mapping from the original observation space $\mathscr{O}$ to the augmented observation space $\mathscr{O}^{aug}$:

$$o^{aug} \triangleq f(o;\nu) \quad \forall o \in \mathscr{O}, \nu \in \mathscr{V} \tag{1}$$

where $\nu \in \mathscr{V}$ is a set of random parameters and $f(.)$ is a function for observation o augmentation.

In supervised learning, data augmentation usually assumes that after some transformation of the input image, the corresponding label or output remains unchanged. In other words, when a certain transformation is applied to an image, the model is expected to give the same predictions for the image before and after the transformation.

Considering the particularity of RL, DrQ [4] defines the *optimality invariance* assumption as adding a constraint on the transformation f , so as to deduce that the state s is

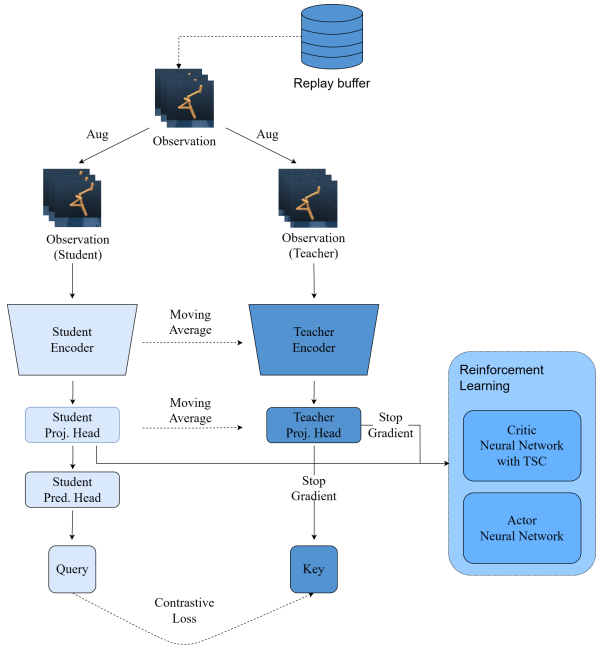

Fig. 1. RLTSC Architecture: A batch of transitions is sampled from the replay buffer. Observations are then image-augmented twice to form teacher and student observations, which are then encoded with the teacher encoder and student encoders, respectively. The representations obtained by the two encoders serve two purposes: (1) They are used as input for the Critic Neural Network with TSC to jointly estimate the Q-function. (2) They are used for similarity measurements and contrastive loss calculation.

equivalent to the observation o and the enhanced state s^{aug} is equivalent to the enhanced observation o^{aug}. Therefore, an optimality-invariant state transformation $f : \mathcal{O} \times \mathcal{V} \to \mathcal{O}$ can be defined as a mapping that preserves the Q-values:

$$Q(s,a) \triangleq Q(f(o;v),a) \quad \forall o \in \mathcal{O}, a \in \mathcal{A}, v \in \mathcal{V} \tag{2}$$

where v is the set of parameters of $f(.)$, drawn from the set of all possible parameters $\mathcal{V}$.

B. Teacher-Student Collaboration

Optimality-invariant caused us to think about how to make full use of this property to enhance the performance of unsupervised RL.

In unsupervised RL, most research has always been limited to using auxiliary tasks in computer vision to help contrastive learning. Instead, we propose a different idea: using the relationship between the Teacher Encoder and the Student Encoder in contrastive learning to construct a auxiliary task.

It is well known that momentum updates are commonly used in contrastive learning to update the Teacher Encoder. Formally, we denote the Student Encoder as $f_s(.)$, the parameters of $f_s(.)$ as v_s, the Teacher Encoder as $f_t(.)$, and the parameters of $f_t(.)$ as v_t, we update v_t by:

$$v_t \longleftarrow m v_t + (1-m) v_s \tag{3}$$

Here $m \in [0,1)$ is a momentum coefficient. It can be seen from Equation (3) that the new v_t is related to the old v_t and v_s. The parameter v_s is updated by back-propagation, but the loss function of back-propagation is affected by v_t.

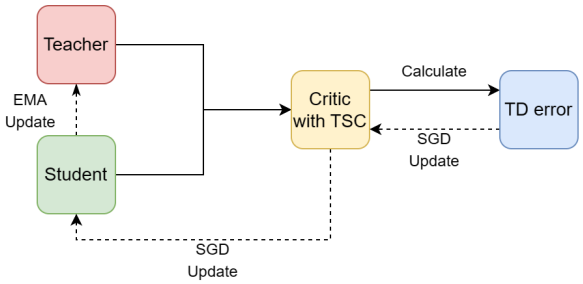

Fig. 2. The mind map of TSC: The Teacher Encoder's representation cooperates with the Student Encoder's representation to calculate the TD error, and the TD error guides the teacher coder's update through SGD and EMA.

According to Optimality Invariance for Q-function, we can obtain an estimate with low variance while taking into account the learning of the Teacher Encoder and the Student Encoder:

$$Q(s,a) \approx \frac{1}{2}(Q(f_t(o_t;v_t),a) + Q(f_s(o_s;v_s),a)) \tag{4}$$

where o_t and o_s are different transformations on the same observation, which guarantees low variance of the estimates. $v_t$and v_s are the parameters of f_s and f_t respectively, which take into account the update of the Teacher Encoder and the Student Encoder.

This suggests a scheme to update the Q-function. First, we calculate the values and the target values for every transition tuple (o_i, a_i, r_i, o_i') as:

$$q_i = \frac{1}{2}(Q_\theta(f_s(o_s, v_s), a_i) + Q_\theta(f_t(o_t, v_t), a_i) \tag{5}$$

$$y_i = r_i + \frac{1}{2}\gamma(Q_{\theta'}(f_s(o_s', v_s), a_i') + Q_{\theta'}(f_t(o_t', v_t), a_i')) \tag{6}$$

Then, the Q-function is updated using these targets through an SGD update using learning rate λ_θ:

$$\theta \leftarrow \theta - \lambda_\theta \nabla_\theta (q_i - y_i)^2 \tag{7}$$

We call this idea Teacher-Student Collaboration. A generic off-policy actor-critic algorithm with TSC is shown in Algorithm 1. The mind map of this idea is shown in Fig. 2. There are two main benefits of this approach: (i) the Teacher Ecoder and the Student Encoder generate agent states with the same Q-values, thus **providing a mechanism to reduce the variance of Q-function estimation**. (ii) The combination of SGD Update and EMA Update is equivalent to a **heuristic update for Teacher**.

C. Enhancements in Contrastive Unsupervised Learning

RLTSC employs instance discrimination similar to CURL [11], and we have retained most of CURL's settings in our implementation. However, considering that CURL's use of contrastive learning may be somewhat outdated and may not yield optimal representation learning, we have enhanced it by incorporating various advancements in contrastive unsupervised learning techniques. The following will introduce

Algorithm 1 A generic off-policy actor-critic algorithm with Teacher-Student Collaboration

Hyperparameters: mini-batch size N, learning rate λ_θ, Student Encoder f_s, Teacher Encoder f_t, target network update rate τ, random image augmentation aug.

for $i = 1$ to T **do**
  $a_t \sim \pi(.|o_t)$
  $o'_t \sim p(.|o_t, a_t)$
  $\mathscr{D} \leftarrow \mathscr{D} \cup (o_t, a_t, r(o_t, a_t), o'_t)$ ▷ Add a transition
  $UpdateCritic(\mathscr{D})$
  $UpdateActor(\mathscr{D})$
end for
procedure $UpdateCritic(\mathscr{D})$
  $\{(o_i, a_i, r_i, o'_i)\}_{i=1}^{N} \sim \mathscr{D}$ ▷ Sample a mini batch
  for $i = 1$ to N **do**
    $o_s, o_t, o'_s, o'_t = aug(o_i), aug(o_i), aug(o'_i), aug(o'_i)$
    $a'_i \sim \pi(.|f_s(o'_i, \nu'_s))$
    $\hat{Q}_i = \frac{1}{2}(Q_{\theta'}(f_s(o'_s, \nu_s), a'_i) + Q_{\theta'}(f_s(o'_t, \nu_t), a'_i))$
    $q_i \leftarrow \frac{1}{2}(Q_\theta(f_s(o_s, \nu_s), a_i) + Q_\theta(f_t(o_t, \nu_t), a_i)$
    $y_i \leftarrow r(s_i, a_i) + \gamma \hat{Q}_i$
  end for
  $J_Q(\theta) = \frac{1}{N}\sum_{i=1}^{N}(q_i - y_i)^2$
  $\theta \leftarrow \theta - \lambda_\theta \nabla J_Q(\theta)$ ▷ Update the critic
  $\theta' \leftarrow (1-\tau)\theta' + \tau\theta$ ▷ Update the critic target
end procedure

the key components of the RLTSC's contrastive unsupervised learning. These components collaborate to establish RLTSC's robust representation learning capabilities.

Image Augmentation. We use random displacement image augmentation with bilinear interpolation. In the settings of visual continuous control by DMControl, this augmentation can be instantiated by first padding each side of the 84×84 observation with 4 pixels, and then choosing a random 84×84 crop, resulting in an offset of ± 4 pixels from the original image, and finally replacing each pixel value with the average of the four closest pixel values. We apply the aforementioned data augmentation consistently across all frames in the stack, ensuring temporally consistent spatial jittering. This form of augmentation effectively combines the demands for both variability and low difficulty [23] in vision-based RL augmentation.

Image Encoder. Due to the simplicity of the DMControl observations, We use the Image Encoder with residual connections [24] in RLTSC. Compared to an encoder composed solely of convolution layers, the encoder with residual connections can more effectively extract pixel features. Otherwise, we use the same setup as most advanced contrastive learning methods [25], [17], [26] : add a projection head and a prediction head after the Student Encoder, and add a projection head after the Teacher Encoder. This practice frequently facilitates the model in acquiring improved representations through projection and prediction.

Contrastive Loss. Similar to CPC [27], we use the bilinear inner product to measure the similarity between q and k. This method can be expressed as $sim(q,k) = q^T W k$, where W is a learned parameter matrix. To learn the embeddings of these similarity relations, we use the InfoNCE loss:

$$\mathbb{L}_q = log \frac{exp(q^T W k_+)}{exp(q^T W k_+) + \sum_{i=0}^{K-1} exp(q^T W k_i)} \tag{8}$$

The sum is over one positive and K negative samples[16]. Intuitively, the Equation (8) can be interpreted as the log-loss of a (K+1)-way softmax classifier with the label k_+.

IV. Experiment

In this section, we evaluate the effectiveness of RLTSC in improving the data efficiency of vision-based RL algorithms. We conduct extensive experiments on the continuous control benchmark DMControl and the discrete control benchmark Atari.

A. Evaluation on DMControl

Setup. DMControl presents a fairly challenging and diverse set of tasks, including bipedal balance, locomotion, contact force, and goal-reaching with both sparse and dense reward signals. On DMControl, according to previous work[11], [6], [4], [8], [5], [12], we selected six commonly used environments from DMControl, *i.e.*, *Finger, spin; Cartpole, swingup; Reacher, easy; Cheetah, run; Walker, walk and Ball in cup, catch* for evaluation. We test the mean performance of the agent over 10 episodes of 100k and 500k environment steps. This setup is known in many works as DMControl-100k and DMControl-500k. The score for each environment ranges from 0 to 1000.

Baselines. Throughout the experimental evaluation process, we carefully chose a set of representative baselines for comparison with our experimental results, demonstrating the effectiveness of our method. These baselines are competitive methods for continuous control from pixels, including: (i) Static-SAC [15] which assumes access to physical state-based features and does not operate directly from pixels; (ii) DrQ [4] which uses an implicit regularization method to increase the data efficiency of vision-based RL; (iii) Dreamer [28] and (iv) PlayVirtual [29] both of which learn a latent space world model and plan explicitly through it; (v) CURL [11] which uses an unsupervised contrastive loss to learn representations of pixels to aid vision-based RL; (vi) CCLF [12] which uses a model-agnostic contrastive-curiosity-driven learning framework to make full use of sample importance and improve learning efficiency in a self-supervised manner. (vii) MLR [8] which uses mask-based reconstruction to facilitate state representation learning in RL.

Results. The data efficiency results of the DMControl experiments are shown in Table I. We can observe that: (i) On DMControl-100k, RLTSC is **the state-of-the-art vision-based RL algorithm** on 5 out of 6 DMControl environments. On DMControl-500k, RLTSC is **the state-of-the-art vision-based RL algorithm** on 4 out of 6 DMControl environments. Whether on DMControl-100k or DMControl-500k, the mean score of RLTSC is **the best**

TABLE I

COMPARISON RESULTS (MEAN ± STD DEVIATION FOR 10 SEEDS) ON THE DMCONTROL-100K AND DMCONTROL-500K BENCHMARKS.

100K Step Scores	*State-SAC*	*DrQ*	*Dreamer*	*PlayVirtual*	*CURL*	*CCLF*	*MLR*	*RLTSC(ours)*
Finger, spin	811±46	901±104	341±70	915±49	767±56	**944±42**	907±58	933±54
Cartpole, swingup	835±22	759±92	326±27	816±36	582±146	799±61	806±48	**856±23**
Reacher, easy	746±25	601±213	314±115	785±142	538±233	738±99	866±103	**924±128**
Cheetah, run	616±18	344±67	235±137	474±50	299±48	317±38	482±38	**577±43**
Walker, walk	891±82	612±164	277±12	460±173	403±24	648±110	643±114	**808±56**
Ball in cup, catch	746±91	913±53	246±174	926±31	769±43	914±20	933±16	**963±18**
Mean	774.2	688.3	289.8	729.3	559.7	726.7	772.8	**843.5**
500K Step Scores								
Finger, spin	923±21	938±103	796±183	963±40	926±45	974±6	973±31	**976±18**
Cartpole, swingup	848±15	868±10	762±27	865±11	841±45	869±9	**872±5**	857±9
Reacher, easy	923±24	942±71	793±164	942±66	929±44	941±48	957±41	**977±12**
Cheetah, run	795±30	660±96	570±253	719±51	518±28	588±22	674±37	**862±48**
Walker, walk	948±54	921±45	897±49	928±30	902±43	936±23	**939±10**	937±33
Ball in cup, catch	974±33	963±9	879±87	967±5	959±27	961±9	964±14	**972±11**
Mean	901.8	882.0	782.8	897.3	845.8	878.2	896.5	**930.2**

among all vision-based RL algorithms. (ii) On DMControl-100k and DMControl-500k, our proposed method improves the mean score by **50.7%** and **10.0%** compared with the baseline, respectively. This demonstrates the advantages of RLTSC in improving the data efficiency of vision-based RL algorithms. (iii) On DMControl-100k and DMControl-500k, our proposed method improves the mean scores by **9.0%** and **3.1%**, respectively, compared with State-SAC(which takes physical state-based features as input). Previous model-free vision-based RL methods have never surpassed State-SAC in data efficiency, and **RLTSC has achieved this for the first time**. This is an amazing finding, and **it also shows that good representation learning can allow vision-based RL methods to obtain better representations than physical representations**.

B. Evaluation on Atari

Implementation. In SectionIII, we have discussed the implementation details of RLTSC on the continuous control benchmark DMControl. We implemented our method using CURL as the baseline on the discrete control benchmark Atari. We only add the TSC idea to the baseline, and did not modify other parts of the algorithm such as the encoder and hyperparameters.

Setup. For discrete control, similar to many other works [11], [8], [4], [30], we test the agent on the Atari-100k benchmark, which contains 26 Atari games and allows the agent to train for 100k interaction steps (i.e., 400k environment steps with 4 action repetitions). We measure the agent's performance on each game using human-normalized performance, and use the Mean Human-Normalized Performance(Mean Human-Norm'd) of all games to evaluate the data efficiency of the algorithm. The human-normalized performance is calculated by $\frac{S_A - S_R}{S_H - S_R}$, where S_A, S_R and S_H are the scores of the agent, random play and the expert human, respectively.

Baselines. For benchmarking performance on Atari, we compare RLTSC to: (i) Human Performance [31]; (ii) Random Agent [30]; (iii) OTRainbow [32] which overtrains Rainbow to achieve data efficiency. (iv) Eff. Rainbow [31] which modifies Rainbow's hyperparameters to improve data efficiency. (v) SimPLe [30] which is the top performing model-based RL method for vision in terms of Atari's data efficiency. (vi) DrQ. (vii) CURL. (viii) CCLF.

Results. The data efficiency results of the Atari experiments are shown in Table II. We can observe that: (i) Our proposed RLTSC outperforms all other algorithms on **9** out of **26** games, and outperforms the baseline on **17** out of **26** games. (ii) The Mean Human-Norm'd of RLTSC in all games is higher than all other model-free vision-based RL methods, and the performance is close to that of the state-of-the-art model-based method SimPLe. Compared with Eff. Rainbow and CURL, RLTSC's Mean Human-Norm'd has increased by 51.9% and 13.6%. (iii) RLTSC **surpasses human performance** on three games: Jamesbond, Krull and Road Runner.

C. Ablation Study

To demonstrate the effectiveness of these two improvements, in this section, we conduct an ablation study.

Setup. We employ DMControl-500k for assessing the model's performance. Unless otherwise specified, we use 3 random seeds to run each model in these environments.

Baselines. Since we have added two improvements to CURL, in order to prove the effectiveness of the improvement, the ablation experimental models we selected are: (i) CURL. (ii) *RLTSC w.o. TSC* which only integrates CURL with advanced methods of contrastive unsupervised learning, without adding TSC idea. (iii) RLTSC which combines CURL with advanced methods of contrastive unsupervised learning and TSC idea.

Results. The data efficiency results of the DMControl experiments are shown in Table III and Fig. 3. We can observe that: Both improvements to CURL have some effect on it. On DMControl-500k, the mean performance of CURL is **1.06**× of the original after only the improvement of the

TABLE II

COMPARISON RESULT (MEAN DEVIATION FOR 5 SEEDS) ON THE ATARI-100K BENCHMARK.

Game	*Human*	*Random*	*OTRainbow*	*Eff. Rainbow*	*SimPLe*	*DrQ*	*CURL*	CCLF	*RLTSC(ours)*
Alien	7127.7	227.8	824.7	739.9	616.9	771.2	558.2	920.0	**1129.8**
Amidar	1719.5	5.8	82.8	**188.6**	88.0	102.8	142.1	154.7	169.0
Assault	742.0	222.4	351.9	431.2	527.2	452.4	600.6	**612.4**	473.6
Asterix	8503.3	210.0	628.5	470.8	**1128.3**	603.5	734.5	708.8	734.0
Bank Heist	753.1	14.2	**182.1**	51.0	34.2	168.9	131.6	36.0	67.6
Battle Zone	37187.5	2360.0	4060.6	10124.6	5184.4	12954.0	**14870.0**	5775.0	11700.0
Boxing	12.1	0.1	2.5	0.2	9.1	6.0	1.2	7.4	**11.9**
Breakout	30.5	1.7	9.84	1.9	**16.4**	16.1	4.9	2.7	5.7
Chopper Cmd	7387.8	811.0	1033.33	861.8	**1246.9**	780.3	1058.5	765.0	1120.6
Crazy Climber	35829.4	10780.5	21327.8	16185.3	**62583.6**	20516.5	12146.5	7845.0	12810.1
Demon Attack	1971.0	152.1	711.8	508.0	208.1	1113.4	817.6	**1360.9**	834.2
Freeway	29.6	0.0	25.0	27.9	20.3	9.8	26.7	22.6	**28.1**
Frostbite	4334.7	65.2	231.6	866.8	254.7	331.1	1181.3	1401.0	**1731.6**
Gopher	2412.5	257.6	778.0	349.5	771.0	636.3	669.3	**814.7**	682.4
Hero	30826.4	1027.0	6458.8	6857.0	2656.6	3736.3	6279.3	6944.5	**7394.4**
Jamesbond	302.8	29.0	112.3	301.6	125.3	236.0	**471.0**	308.8	327.0
Kangaroo	3035.0	52.0	605.4	779.3	323.1	940.6	872.5	650.0	**1402.8**
Krull	2665.5	1598.0	3277.9	2851.5	**4539.9**	4018.1	4229.6	3975.0	3665.7
Kung Fu Master	22736.3	258.5	5722.2	14346.1	**17257.2**	9111.0	14307.8	12605.0	12680.4
Ms Pacman	6951.6	307.3	941.9	1204.1	**1480.0**	960.5	1465.5	1397.5	1392.2
Pong	14.6	-20.7	1.3	-19.3	**12.8**	-8.5	-16.5	-17.3	-14.8
Private Eye	69571.3	24.9	100.0	97.8	58.3	-13.6	**218.4**	100.0	100.0
Qbert	13455.0	163.9	509.3	1152.9	1288.8	854.4	1042.4	953.8	**2170.8**
Road Runner	7845.0	11.5	2696.7	9600.0	5640.6	8895.1	5661.0	11730.0	**15040.5**
Seaquest	42054.7	68.4	286.92	354.1	**683.3**	301.2	384.5	550.5	396.8
Up N Down	11693.2	533.4	2847.6	2877.4	3350.3	3180.8	2955.2	3376.3	**4072.0**
Mean Human-Norm'd	1.000	0.000	0.264	0.285	**0.443**	0.357	0.381	0.382	0.433

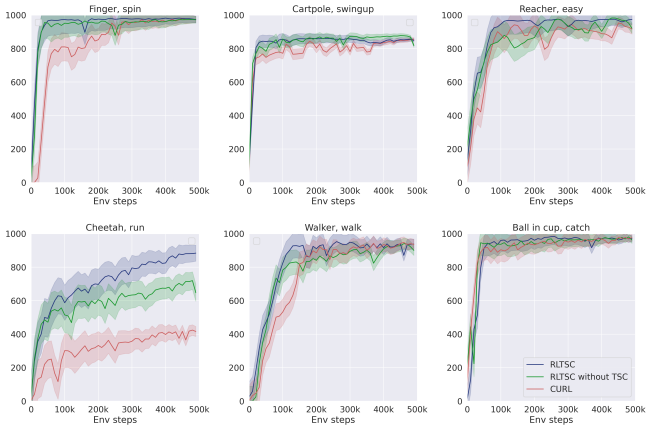

Fig. 3. Test performance during training (500k environment steps). Lines represent mean scores for random seeds, shading represents corresponding standard deviations.

TABLE III

ABLATION EXPERIMENT RESULTS (MEAN ± STD) ON THE DMCONTROL-500K BENCHMARKS.

500K Step Scores	*CURL*	*RLTSC w.o. TSC*	*RLTSC*
Finger, spin	926±45	953±56	976±18
Cartpole, swingup	841±45	876±23	857±9
Reacher, easy	929±44	972±35	977±12
Cheetah, run	518±28	702±51	862±48
Walker, walk	902±43	934±42	937±33
Ball in cup, catch	959±27	960±17	972±11
Mean	845.8	899.5	930.2
Mean CURL'd	1.0	1.06	1.10

unsupervised learning part; the mean performance of CURL is **1.10**× of the original after two kinds of improvements.

V. CONCLUSIONS

In this work, we present two notable enhancements to the CURL: (i) Introduction of the TSC idea: This innovative approach serves the dual purpose of alleviating the variance of the Q-function estimation within the algorithm and ensuring effective information exchange between representation learning and RL. (ii) Incorporation of Advances in contrastive unsupervised learning: We integrate recent developments in contrastive unsupervised learning into the field of vision-based RL. This innovative aims to enhance the quality of representations used in vision-based RL tasks. The combination of the two improvements contributes to the extraordinary performance of RLTSC. Extensive experiments demonstrate the data efficiency of RLTSC and show that RLTSC achieves state-of-the-art performance on the DMControl benchmark and outperforms other model-free vision-based RL methods on the Atari benchmark. We also observe in experiments that RLTSC outperforms State-SAC (with physical state-based features as input) on the DMControl benchmark. And this may suggests that visual embedding can encode additional information that is not present in the state. This research serves as a source of inspiration for the field of robotic learning, suggesting that visual information may, at times, yield superior performance in the realm of robotic control.

REFERENCES

[1] Y. Tassa, Y. Doron, A. Muldal, T. Erez, Y. Li, D. d. L. Casas, D. Budden, A. Abdolmaleki, J. Merel, A. Lefrancq, *et al.*, "Deepmind control suite," *arXiv preprint arXiv:1801.00690*, 2018.

[2] M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling, "The arcade learning environment: An evaluation platform for general agents," *Journal of Artificial Intelligence Research*, vol. 47, pp. 253–279, 2013.

[3] T. He, Y. Zhang, K. Ren, M. Liu, C. Wang, W. Zhang, Y. Yang, and D. Li, "Reinforcement learning with automated auxiliary loss search," *Advances in Neural Information Processing Systems*, vol. 35, pp. 1820–1834, 2022.

[4] I. Kostrikov, D. Yarats, and R. Fergus, "Image augmentation is all you need: Regularizing deep reinforcement learning from pixels," *arXiv preprint arXiv:2004.13649*, 2020.

[5] D. Yarats, R. Fergus, A. Lazaric, and L. Pinto, "Mastering visual continuous control: Improved data-augmented reinforcement learning," *arXiv preprint arXiv:2107.09645*, 2021.

[6] M. Laskin, K. Lee, A. Stooke, L. Pinto, P. Abbeel, and A. Srinivas, "Reinforcement learning with augmented data," *Advances in neural information processing systems*, vol. 33, pp. 19 884–19 895, 2020.

[7] D. Bertoin and E. Rachelson, "Local feature swapping for generalization in reinforcement learning," *arXiv preprint arXiv:2204.06355*, 2022.

[8] T. Yu, Z. Zhang, C. Lan, Y. Lu, and Z. Chen, "Mask-based latent reconstruction for reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 25 117–25 131, 2022.

[9] M. Kim, K. Rho, Y.-d. Kim, and K. Jung, "Action-driven contrastive representation for reinforcement learning," *Plos one*, vol. 17, no. 3, p. e0265456, 2022.

[10] A. Anand, E. Racah, S. Ozair, Y. Bengio, M.-A. Côté, and R. D. Hjelm, "Unsupervised state representation learning in atari," *Advances in neural information processing systems*, vol. 32, 2019.

[11] 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.

[12] C. Sun, H. Qian, and C. Miao, "Cclf: A contrastive-curiosity-driven learning framework for sample-efficient reinforcement learning," *arXiv preprint arXiv:2205.00943*, 2022.

[13] D. Jain, A. Majumder, S. Dutta, and S. Kumar, "Crc-rl: A novel visual feature representation architecture for unsupervised reinforcement learning," *arXiv preprint arXiv:2301.13473*, 2023.

[14] J. Zhu, Y. Xia, L. Wu, J. Deng, W. Zhou, T. Qin, T.-Y. Liu, and H. Li, "Masked contrastive representation learning for reinforcement learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[15] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International conference on machine learning*. PMLR, 2018, pp. 1861–1870.

[16] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9729–9738.

[17] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, *et al.*, "Bootstrap your own latent-a new approach to self-supervised learning," *Advances in neural information processing systems*, vol. 33, pp. 21 271–21 284, 2020.

[18] T. He, Y. Zhang, K. Ren, M. Liu, C. Wang, W. Zhang, Y. Yang, and D. Li, "Reinforcement learning with automated auxiliary loss search," *arXiv preprint arXiv:2210.06041*, 2022.

[19] E. Shelhamer, P. Mahmoudieh, M. Argus, and T. Darrell, "Loss is its own reward: Self-supervision for reinforcement learning," *arXiv preprint arXiv:1612.07307*, 2016.

[20] D. Yarats, A. Zhang, I. Kostrikov, B. Amos, J. Pineau, and R. Fergus, "Improving sample efficiency in model-free reinforcement learning from images," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 12, 2021, pp. 10 674–10 681.

[21] M. Jaderberg, V. Mnih, W. M. Czarnecki, T. Schaul, J. Z. Leibo, D. Silver, and K. Kavukcuoglu, "Reinforcement learning with unsupervised auxiliary tasks," *arXiv preprint arXiv:1611.05397*, 2016.

[22] Z. D. Guo, B. A. Pires, B. Piot, J.-B. Grill, F. Altché, R. Munos, and M. G. Azar, "Bootstrap latent-predictive representations for multitask reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 3875–3886.

[23] G. Ma, L. Zhang, H. Wang, L. Li, Z. Wang, L. Shen, X. Wang, and D. Tao, "Learning better with less: Effective augmentation for sample-efficient visual reinforcement learning," *arXiv preprint arXiv:2305.16379*, 2023.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[25] C. Xinlei, X. Saining, and H. Kaiming, "An empirical study of training self-supervised visual transformers," *arXiv preprint arXiv:2104.02057*, vol. 8, 2021.

[26] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 9650–9660.

[27] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[28] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, "Dream to control: Learning behaviors by latent imagination," *arXiv preprint arXiv:1912.01603*, 2019.

[29] T. Yu, C. Lan, W. Zeng, M. Feng, Z. Zhang, and Z. Chen, "Playvirtual: Augmenting cycle-consistent virtual trajectories for reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 5276–5289, 2021.

[30] L. Kaiser, M. Babaeizadeh, P. Milos, B. Osinski, R. H. Campbell, K. Czechowski, D. Erhan, C. Finn, P. Kozakowski, S. Levine, *et al.*, "Model-based reinforcement learning for atari," *arXiv preprint arXiv:1903.00374*, 2019.

[31] H. P. Van Hasselt, M. Hessel, and J. Aslanides, "When to use parametric models in reinforcement learning?" *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[32] K. Kielak, "Importance of using appropriate baselines for evaluation of data-efficiency in deep reinforcement learning for atari," *arXiv preprint arXiv:2003.10181*, 2020.