2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Automated Action Evaluation for Robotic Imitation Learning via Siamese Neural Networks*

Xiang Chang[1], Fei Chao[2,1,*], Changjing Shang[1], Qiang Shen[1]

***Abstract*— Despite recent advances in video-guided robotic imitation learning, many methods still rely on human experts to provide sparse rewards that indicate whether robots have successfully completed tasks. The challenge of enabling robots to autonomously evaluate whether their actions can complete complex, multi-stage tasks remains unresolved. In this work, we propose an efficient few-shot robotic learning algorithm that centres around learning and evaluating from a third-person perspective to address the aforementioned challenge. We develop a novel Siamese neural network-based robotic action-state evaluation system, named "Behavior-Outcome Dual Assessment" (BODA), in our robotic imitation learning system, so as to replace artificial evaluations from human experts in multi-stage imitation learning processes and to improve learning efficiency. In this way, one video demonstration of a target task is divided into several stages. For each stage, we design two Siamese neural network-based evaluation modules in BODA: One module focuses on action changes, and the other handles working environment changes. The two modules work together to provide a comprehensive assessment of the robot's completion of each stage from the view of both the action and working environment changes. Then, BODA is integrated within a model-based reinforcement learning framework to enable the completion of our imitation learning cycle. Extensive experiments demonstrate that the evaluation processes of BODA can automatically and accurately evaluate task completion status without human intervention. In contrast to conventional methods, BODA is able to keep the accumulation of errors within acceptable limits through self-assessment in stages.**

I. Introduction

Autonomous robots that can assist humans in situations of daily life have been a long-standing vision of robotics. The main step toward this goal is to create robots that can learn new tasks autonomously or through simple demonstrations, such as video clips, in unconstrained environments. Utilizing demonstrations is an effective means of convey complex tasks and circumvent exploration challenges [1], especially in the case of service-level robots intended for public use. Demonstrating and imitating tasks can enhance a robot's usability and reduces user learning costs. To accommodate multi-stage tasks, we partition input video demonstrations into several instructional images that represent distinct stages of the task. For example, a video demonstration of moving a wooden block is divided into four moving stages (Catch, pick up, move, and put down) with a length-fixed time step, and the end frame of each stage is collected as the introduction image. Other studies used state representation learning methods to enable robots can then encode introduction images, with extra neural networks, into latent states rather than into image pixels [2]. This kind of method is widely applied in robotic imitation tasks.

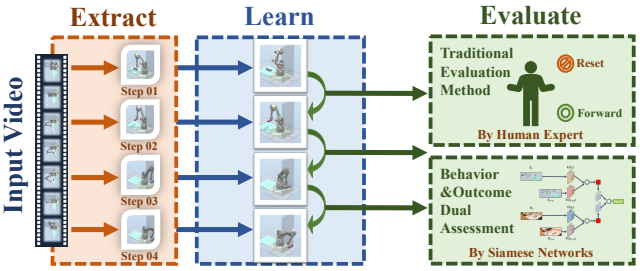

Fig. 1. Comparison between conventional methods and ours. Left: Extract instruction images from the video demonstration. Center: Learn tasks step-by-step by Model-based Reinforcement Learning. Right: Assessment and feedback by human vs. by BODA at the end of each stage.

However, this type of method also leads to a critical challenge: Stage classifiers are unable to accurately assess whether the expected action and objective have been fully accomplished. This challenge causes error accumulations within each stage and even affects sub-sequential learning. Thus, existing studies must introduce human experts for the stage completion status assessments [1]–[5], shown in Fig. 1. Although human involvement can support accurate status feedback of the current working environment, this method leads to massive human work and slow learning speed.

Therefore, this study designs an automated robotic action-state evaluation system, named "Behavior-Outcome Dual Assessment" (BODA), which consists of two Siamese neural network implemented modules [6]–[8] to model human experts' judgment process in the robotic action states evaluation. BODA refers to a method of evaluating performance by simultaneously considering both the actions taken and their resulting consequences, as that the quality of behavior cannot be determined solely by looking at the actions taken, but must also take into account the outcomes that those actions produce. By considering both aspects together, a more comprehensive and accurate assessment of performance can be achieved. In BODA, one module focuses on robotic action changes, and the other handles working environment changes. The two modules work together to provide a comprehensive assessment of the robot's completion of each stage from the view of the action and working environment

*This work was supported by the Natural Science Foundation of Fujian Province of China (No. 2021J01002)

[1]X. Chang, F. Chao, C. Shang and Q. Shen are with Department of Computer Science, Institute of Mathematics, Physics and Computer Science, Aberystwyth University, SY23 3DB, UK

[2]F. Chao is also with Department of Artificial Intelligence, School of Informatics, Xiamen University, 361005, China, *Corresponding author

changes, which can provide a more diverse range of rewards, to make the robot imitation learning process autonomous and efficient.

By embedding BODA into a reinforcement learning framework, our robot system can solve complex robotic imitation tasks without any human involvement. We evaluate BODA on multi-stage complex tasks using two 7-DoF robotic arms. BODA outperforms current methods regarding data efficiency and task success. To a certain extent, BODA can completely replace the manual assessment, automating the entire simulation learning process.

The main contributions of this paper include 1) Eliminating the requirement for manual intervention in the multi-stage task imitation learning process and automating the entire process. 2) Modelling the human judgment processes to automate the action state evaluation in both the action and working environment dimensions. 3) Introducing stage self-assessments for multi-stage imitation learning, and further avoiding the accumulation of errors in the learning process.

II. Related Work

A. Robotic Learning for Multi-stage Tasks

Multi-stage tasks in robotics are significantly more complex than those single-stage tasks. Many recent studies have enabled robots to learn complex multi-stage tasks by extracting video demonstrations into step-by-step instruction images in a time-stepped way [1], [2]. To learn multi-stage complex tasks, many researchers have proposed different solutions such as Long-horizon visual planning [3], [4], chaining dynamic movement primitives [9], one-shot visual imitation learning [10], domain-agnostic video discriminator [11], and dynamics cycle-consistency [12]. However, these studies often require the introduction of human interventions in their learning processes.

Furthermore, Reinforcement Learning (RL) is a ideal method that empowers robots to learn a diverse range of real-world skills autonomously [13]–[19]. In the realm of robotics, achieving data efficiency is often a significant challenge. To address this issue, several methods have been proposed to enhance the data efficiency of RL algorithms. One commonly used technique is state representation learning, which involves reducing the dimensionality of the data used for RL [20]. Another approach to enhance data efficiency is model-based RL, which has demonstrated greater efficacy in practice compared to model-free RL methods [9]

B. Siamese Neural Networks

Fig. 2 shows a typical Siamese neural network with a twin-network structure, the twin networks share identical weights. Siamese neural networks are often used to compare two images and determine whether the two images contain the same object [21]. The feature vectors of two images are generated by feeding them into identical convolutional networks. These feature vectors are then used to perform a binary classification that indicates the similarity or dissimilarity between the two domains [22], [23].

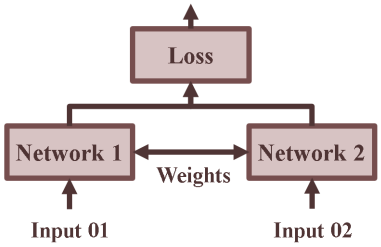

Fig. 2. Architecture of Siamese Neural Network.

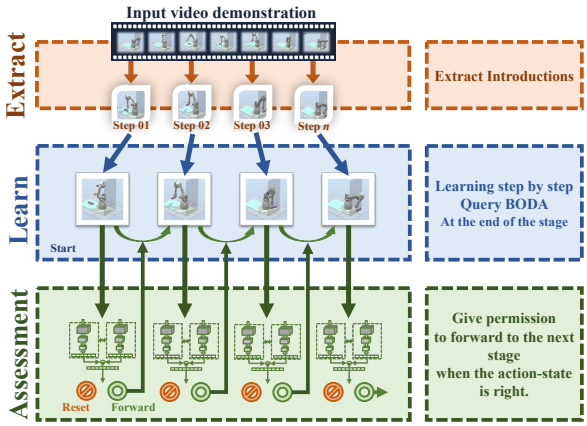

Fig. 3. Architecture of Proposed Approach.

Siamese neural networks have been extensively used in various fields, as demonstrated by several studies [6]–[8]. However, the utilization of Siamese neural networks in robotics remains limited, with only a few researchers exploring this avenue. Such as robotic anomaly detection system [24], facial emotion recognition in human-robot interaction [25], and making robots capable of grasping and identifying both known and new objects in obstructed working environments [26].

III. Methods

In the proposed approach, a robotic learning task is specified by a set of input video demonstrations, each of which contains an observed trajectory depicting the robot performing the task from a third-person perspective. To learn directly from input video demonstrations, the proposed method shown in Fig. 3 has three key steps:

1. Extract: extracting introduction images from videos.

2. Learn: modeling robot states and actions, and learning tasks step-by-step by a model-based reinforcement learning algorithm.

3. Assessment: evaluating the action state at the end of every stage.

This section details the whole workflow of our system.

A. Extract

Inspired by current work on robotic imitation learning [1], we adopt a similar approach by extracting and utilizing instructional images from video demonstrations, which define key stages of a robotic task. These instruction images

segment the overall task into stages (see Fig. 3). In this way, the robotic learning process is less affected by irrelevant information from the video demonstrations to reduce learning difficulties.

The images used for instructional purposes during the i-th stage are derived from the t_i-th frame of the input video. The time steps, denoted as $\{t_1, \cdots, t_S\}$, are defined over a fixed time span and correspond, intuitively, to the completion of natural stages in the task.

B. Learn

To control learning in image-based domains, numerous studies have shown that state representation learning is an effective tool for improving data efficiency [1], [2], [27]–[30]. Inspired by them, we use a representation learning method to lead stage-wise action learning in our work. Our approach draws inspiration from the work of Lee et al. [28], which proposes a framework for learning a latent state representation of observations through a probabilistic temporally-structured latent variable model.

Our approach is based on the assumption that the underlying state of the system, denoted as $\mathbf{S}_{t+1}$, is unobserved, but evolves as a function of the previous state, denoted as $\mathbf{S}_t$, and action, denoted as $\mathbf{a}_t$. We consider the robot images, captured from a third-person perspective, as observations, denoted as $\mathbf{O}_{t+1}$, of this state. The generative model for $\mathbf{S}_t$ can be summarized by an initial state distribution and a learned neural network dynamics model, then, we have:

$$p(\mathbf{s}_1) = \mathcal{N}(\mathbf{s}_1; 0, \mathbf{I}) \tag{1}$$

$$p(\mathbf{s}_{t+1} \mid \mathbf{s}_t, \mathbf{a}_t) = \mathcal{N}(\mathbf{s}_{t+1}; \mu(\mathbf{s}_t, \mathbf{a}_t), \Sigma(\mathbf{s}_t, \mathbf{a}_t)) \tag{2}$$

where μ and Σ are parameterized by neural networks; and $p(\mathbf{o}_t \mid \mathbf{s}_t)$ is a decoder represented as a convolutional neural network to complete the generative model.

A variational distribution, $q(\mathbf{s}_{1:T}; \mathbf{o}_{1:T})$, is introduced to learn this model. The distribution approximates the posterior $p(\mathbf{s}_{1:T} \mid \mathbf{o}_{1:T}, \mathbf{a}_{1:T})$, here the $1:T$ notation denotes entire trajectories. Thus, the approach uses a mean field variational approximation, which is defined as:

$$q(\mathbf{s}_{1:T}; \mathbf{o}_{1:T}) = \prod_t q(\mathbf{s}_t; \mathbf{o}_t) \tag{3}$$

Additionally, the encoder, denoted as $q(\mathbf{s}_t; \mathbf{o}_t)$, is also a convolutional network. The parameters of both p and q are jointly learned by maximizing the variational lower bound (ELBO), defined as:

$$\begin{aligned} \mathrm{ELBO}[p, q] = \mathbb{E}_q\left[p(\mathbf{o}_t \mid \mathbf{s}_t)\right] - D_{\mathrm{KL}}\left(q_{\mathbf{s}_1}(\cdot; \mathbf{o}_1) \,\|\, p_{\mathbf{s}_1}\right) \\ - \mathbb{E}_q\left[\sum_{t=1}^{T-1} D_{\mathrm{KL}}\left(q_{\mathbf{s}_{t+1}}(\cdot; \mathbf{o}_{t+1}) \,\|\, p_{\mathbf{s}_{t+1}}(\cdot \mid \mathbf{s}_t, \mathbf{a}_t)\right)\right]. \end{aligned} \tag{4}$$

The entire structured latent variable model is depicted in Fig. 4. This model offers an encoder and dynamics model that we utilize for encoding instruction images and for stage-wise model-based planning in the learned latent space. As

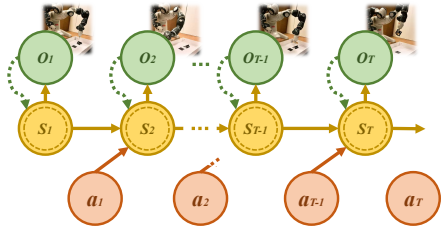

Fig. 4. A latent variable model is employed for the purpose of representing both images and actions of the robot [1].

RL methods have typically exhibited higher data efficiency than model-free methods, the model-based RL approach further reduces the amount of human supervision required during learning. In this work, we apply a RL procedure called MPC-CEM [29], [30]. This is a latent space model-predictive control method that iteratively searches for an optimal sequence of actions using the cross-entropy method.

During Stage s of the robot's operation, the RL procedure utilizes the log probability of stage classifier, $\mathcal{C}_s$, as the reward function and strives to surpass a classifier threshold $\alpha \in [0, 1]$ (a hyperparameter). If the threshold is not met, the planner automatically switches to $\mathcal{C}_{s-1}$, attempting to reset the robot to the beginning of the stage.

If the threshold is met during planning, the robot then queries BODA, which generates signals indicating either success or failure through evaluations of the action and working environment changes. If the signal indicates a failure, the robot switches to the reset behavior. Conversely, if the signal indicates success, the robot proceeds to the next stage and repeats this process.

This stage-wise learning avoids the compounding errors of trying to learn the entire task all at once. The full procedure is summarized in Fig. 3.

C. Assessment

To verify the performance of our learning model at the end of each task stage, we design an evaluation method, Behavior-Outcome Dual Assessment (BODA), to evaluate the robot's task performance in terms of both action and working environment impact.

Referring to the human judgment method for the task completion state, BODA simulates human judgment processes from two aspects. Generally, human experts determine whether a stage object has been completed based on two aspects:

- Whether expected objectives are achieved (e.g., the cup has been placed in the target position; switch has been successfully turned on)
- Whether the active state at the end of the stage is consistent with the instruction images. (to facilitate the start of the next stage).

Only when both objectives have been met, i.e., only when a target action has been completed and had a desired impact on the working environment, we can conclude that the robot has completed this stage (see Fig. 5).

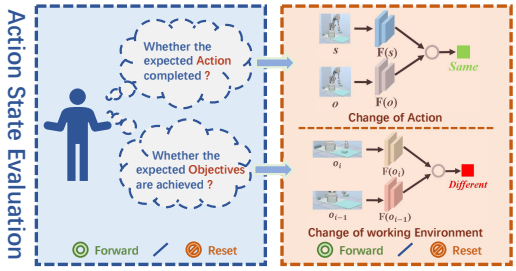

Fig. 5. By modeling the human judgment process, we designed two Siamese network frameworks to replace humans in the robot action states evaluation process: One focuses on action changes, the other on working environment changes.

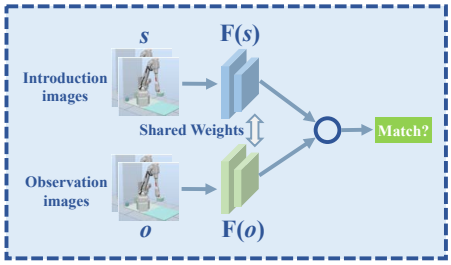

Fig. 6. Assessing Action Changes: A Siamese neural network framework comparing observations (o) and introductions (s).

To this end, we design BODA based on the different characteristics of action and working environment changes. BODA is implemented by two Siamese neural network-based modules to replace human experts in the robotic action-state evaluation process. One module focuses on action changes, and the other handles working environment changes. The two modules work together to provide a comprehensive assessment of the robot's completion of each stage from the view of the action and working environment changes.

1) Assessing Action Changes: For each Siamese network module, we create a two-channel convolutional neural network, where one channel calculates feature vectors for introduction images, and the other channel calculates feature vectors for observation images (see Fig. 6).

During training, the distances of the feature vectors within the same stage images are minimized; while the distances of the feature vectors within the different stage images are maximized. At the end of each stage, we compare the observation image with the introduction image by this framework to determine whether the action at that stage is successfully completed. If an observed image's feature is matched to the same stage introduction image's feature, the robot executes the same action as the one shown in the introduction image.

2) Assessing working environment Changes: We believe that after completing a stage of action, the real working environment must have the same change as the introduction image.

As shown in Fig. 7, the second module focuses on the changes in the working environment between two adjacent stages, the observed or introduction images are used for

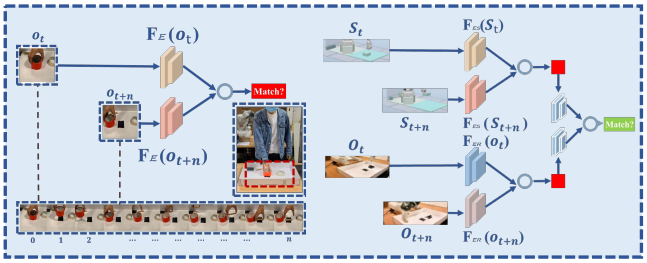

Fig. 7. Assessing working environment Changes: A Siamese neural network framework focus on the changes in the working environment between different stages.

different stages and these changes are measured in terms of the vector distance in the latent space.

To detect changes in the working environment between adjacent stages, we use three dual-channel Siamese neural networks. The first network NI_i detects changes in the working environment between two adjacent stages' introduction images. The changes in the working environment are represented by the vector distances. The second network NO_i handles working environment changes between two adjacent stages' observation images. The last network determines the vector distances between the outputs of NI_i and NO_i. If the vector distance is observed to be the same on both occasions, the robot can be considered to have the same change to the working environment in both introduction and observation and successfully achieved the task goal, as instructed.

3) Usages of BODA: Fig. 8 shows the overall structure of this evaluation component. In particular, if the classifier threshold is met during learning, the robot then queries BODA, and BODA starts to evaluate the robotic action state of the stage:

Fig. 9 shows the observation image and the current stage's introduction image are captured as inputs to the first module of BODA. In this phase, the module focuses on the action state of the robot itself, if the robot produces action, which is similar to that within both the observation and introduction images, BODA generates a successful signal to the robot.

For the task objectives, the second module receives four images, two introduction images (one at the beginning and the other at the end of the task stage), and two observation images (one at the beginning and the other at the end of the task). The four images only illustrate the working environment changes in the task workspace. If the model detects that the feature transformations of the two instruction images match those of the observation images, the second module produces a successful signal.

Only when both two modules give successful signals, the robot has successfully produced the specified action and completed the task objective. This is regarded as the basis for success in that stage, allowing the robot to move on to the next learning stage.

IV. Experimentation

Current important work on robotic learning from video demonstrations includes: **Behavioral cloning from observation** (BC/BCO) [31], **Full-video ablation** [32], [33], **Pixel-**

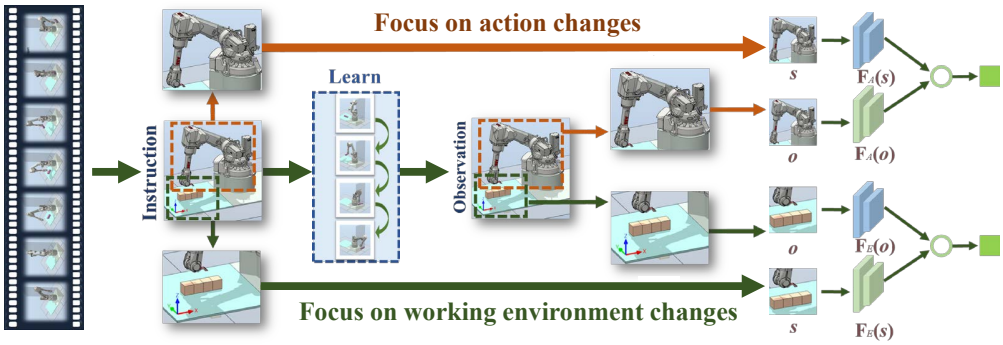

Fig. 8. Illustration of the BODA architecture.

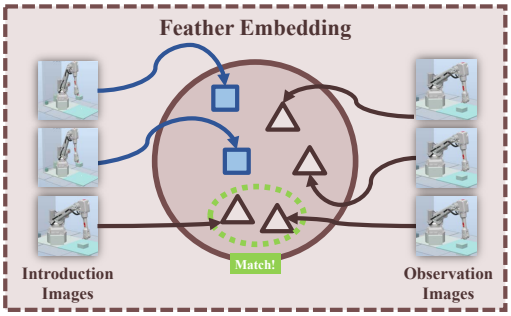

Fig. 9. During training, the distance between the feature vectors corresponding to the same stage images is minimized while that between different stage images is maximized. At the end of each stage, we compare the observation image with the introduction image by this framework to determine whether the action at that stage was successfully completed.

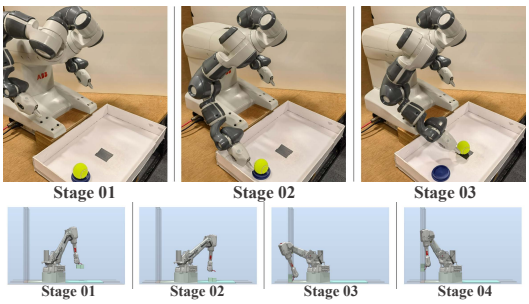

Fig. 10. Example sequence of instructions for our multi-stage tasks, Pushing (top) and Tetris (bottom).

space ablation [34], and **Time-contrastive networks** (TCN) [35]. However, previous comparison experiments have shown that, compared to the method we used in the "learning" module, some of the important work cannot work well under stage-wise learning processes [1], [2]. One of the main reasons for this is that these methods do not allow for staged self-assessment. Therefore, in this section, we mainly focus on the performance of our proposed "evaluation" component, BODA, and the gains that BODA brings to the overall imitation learning process; we thus compared BC, BCO, and TCN in the comparison.

A. Experimental Setup

We evaluated our method on three multi-stage tasks in simulated and real working environments: Moving Teris, Kit assembly, and pushing a ball to the target spot, which are specified as:

- **Tetris** (Tes.): The Tetris task consists of four stages, which are illustrated in Figure 10. The first step involves picking up a block from a desk, followed by placing the block in the target location and putting it down. To train our model for this task, we utilized ten video demonstrations consisting of 400 images. Additionally, negative samples were collected by recording the robot's trajectories while executing random actions that may lead to mistakes during the task.
- **Kit Assembly** (K.A.): The task comprises five stages, which are as follows: grasping the kit, dragging it to the target position, moving the arm, picking up the nut, and placing the nut in the kit.
- **Pushing a ball to the target spot** (P.aB.): The task depicted in Figure 10 involves three stages. Beginning with the ball placed on the table, the robot must find the correct location to initiate an action, push the ball towards the designated target location, and then retract the arm.

Our experiments demonstrate that our method is capable of learning to sequentially integrate multiple skills by following a set of instructions extracted from video demonstrations. To conduct these experiments, we employed two different types of 7-DoF ABB robots. Data for our experiments were collected by filming videos of robot actions and random trajectories. For real working environments, demonstrations were recorded by a camera, post-processed, and extracted as an image of size 256×256 pixels. In addition, for the simulated working environment, demonstrations were generated by screen recording. We extracted the introduction images from the video several times using different time spans to enrich the dataset.

B. Performance and Comparison

Table I presents a quantitative comparison of our representation learning method with other approaches for imitation learning from video. The table indicates whether human intervention is required (H.I.) for each method. We evaluated

TABLE I

PERFORMANCE AND COMPARISON

Meth.	Task	Stage Number					S. R	H. I.
		St. 1	St. 2	St. 3	S. 4	St. 5		
Ours	Tes.	**100.0%**	**85.0%**	**90.0%**	**75.0%**	-	**87.5%**	
	K.A.	100.0%	90.0%	80.0%	75.0%	70.0%	83.0%	
	PaB	100.0%	85.0%	75.0%	-	-	86.7%	
TCN	Tes.	60.0%	45.0%	30.0%	0.0%	-	33.8%	✓
BC	Tes.	90.0%	**85.0%**	60.0%	55.0%	-	72.5%	✓
BCO	Tes.	80.0%	30.0%	0.0%	0.0%	-	27.5%	✓

the performance of these methods on 20 trials for each of the three tasks demonstrated in our experiments, using success rate as the primary metric.

As expected, in all three tasks, the BODA achieved a success rate of over 80% in the evaluations. In particular, BODA clearly achieved the best performance in the Tetris task, compering with TCN, BC, and BCO. The possible reason is that these methods are not good at facing complex multi-stage tasks; thus, they cannot assess their own action status by stages, leading to the accumulation of errors.

The comparative results also suggest that BODA still performed well without any human intervention at all. Compared to traditional methods, BODA is able to keep the accumulation of errors within acceptable limits through staged self-assessment.

TABLE II

ABLATION STUDY OF EACH COMPONENT IN OUR METHOD.

Task	Vars.	Stage Number					S.R
		St.1	St.2	St.3	St.4	St.5	
Tes.	-A	90.0%	70.0%	65.0%	60.0%	-	71.3%
	-B	**100.0%**	75.0%	60.0%	**100.0%**	-	83.7%
	ALL	**100.0%**	**85.0%**	**90.0%**	75.0%	-	**87.5%**
K.A.	-A	95.0%	90.0%	60.0%	55.0%	**70.0%**	74.0%
	-B	90.0%	**95.0%**	55.0%	**75.0%**	60.0%	75.0%
	ALL	**100.0%**	90.0%	**80.0%**	**75.0%**	**70.0%**	**83.0%**
P.aB.	-A	90.0%	**85.0%**	65.0%	-	-	80.0%
	-B	**100.0%**	70.0%	**75.0%**	-	-	81.7%
	ALL	**100.0%**	**85.0%**	**75.0%**	-	-	**86.7%**

C. Ablation Study

To show the benefits of simultaneous assessment of movement and working environment for the proposed BODA, we carried out an ablation for the three tasks. To study the impacts of different components of our system, we considered two variants of our BODA:

- 1. BODA(-A): Siamese Network II is not established; i.e., the evaluation of working environment change is removed.
- 2. BODA(-B): Siamese Network I is not established; i.e., the evaluation of action is removed.

Table II illustrates the results of the two variants. BODA outperformed both two variants. The difference between the results of BODA-A and BODA-B is marginal, but BODA is significantly better.

Fig. 11 illustrates two possible scenarios that can occur at the end of task stages (using task Pushing as an example): 1) the wrong action was performed but the goal was achieved; 2) the appropriate action was performed but the goal was not achieved.

Fig. 11. Two possible scenarios that can occur at the end of task stages (using task Pushing as an example): 1) the wrong action was performed but the goal was achieved; 2) the appropriate action was performed but the goal was not achieved.

From a single global perspective, these two situations are not very different from the correct action state and are more likely to be overlooked, leading to errors carrying over to the next stage and thus ruining the whole learning process. In particular, in multi-stage tasks, the state at the end of the previous stage is exactly the starting state of the next stage, and if the state is misjudged, it will make the next stage of learning from a mistake. Simultaneous assessment of both action and working environment changes prevents this from happening and greatly reduces the possibility of misjudgment.

V. CONCLUSION

In this work, we presented a method for learning robotic multi-stage tasks directly from input video demonstrations without any human involvement. The system extracted the introduction images from the video several times using different time spans, which enabled a stage-wise model-based reinforcement learning (RL) algorithm to learn from visual inputs.

By simulating and optimizing the human judgment processes, BODA evaluates the performance by simulating and optimizing human judgment processes, taking both actions and consequences into account for a comprehensive and accurate assessment. The proposed approach achieved excellent assessment results without the intervention of human experts. This makes the entire imitation learning process fully automatic. However, there is still significant room for improvement in this work. Currently, the two points of interest, action and environmental changes, are given equal weights in the evaluation process, but in the real world, the ratio of importance in each step is sometimes different and there are cases in which judging them dynamically is better.

Thus, in our future work, we will attempt to explore different weights for the various points of interest targeted in the evaluation phase, in order to enhance the transferability and generalization capabilities of the assessment process. Additionally, we plan to investigate automatic stage extraction for more complex tasks using video semantic segmentation, which should improve the accuracy and efficiency of stage task extraction.

REFERENCES

[1] L. Smith, N. Dhawan, M. Zhang, P. Abbeel, and S. Levine, "Avid: Learning multi-stage tasks via pixel-level translation of human videos," *arXiv preprint arXiv:1912.04443*, 2019.

[2] M. Zhang, S. Vikram, L. Smith, P. Abbeel, M. Johnson, and S. Levine, "Solar: Deep structured representations for model-based reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2019, pp. 7444–7453.

[3] A. Mandlekar, D. Xu, R. Martín-Martín, S. Savarese, and L. Fei-Fei, "Learning to generalize across long-horizon tasks from human demonstrations," *arXiv preprint arXiv:2003.06085*, 2020.

[4] K. Pertsch, O. Rybkin, F. Ebert, S. Zhou, D. Jayaraman, C. Finn, and S. Levine, "Long-horizon visual planning with goal-conditioned hierarchical predictors," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17 321–17 333, 2020.

[5] K. Schmeckpeper, O. Rybkin, K. Daniilidis, S. Levine, and C. Finn, "Reinforcement learning with videos: Combining offline observations with interaction," *arXiv preprint arXiv:2011.06507*, 2020.

[6] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a" siamese" time delay neural network," *Advances in neural information processing systems*, vol. 6, 1993.

[7] S. Chopra, R. Hadsell, and 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)*, vol. 1. IEEE, 2005, pp. 539–546.

[8] M. Norouzi, D. J. Fleet, and R. R. Salakhutdinov, "Hamming distance metric learning," *Advances in neural information processing systems*, vol. 25, 2012.

[9] K. Chua, R. Calandra, R. McAllister, and S. Levine, "Deep reinforcement learning in a handful of trials using probabilistic dynamics models," *Advances in neural information processing systems*, vol. 31, 2018.

[10] S. Dasari and A. Gupta, "Transformers for one-shot visual imitation," *arXiv preprint arXiv:2011.05970*, 2020.

[11] A. S. Chen, S. Nair, and C. Finn, "Learning generalizable robotic reward functions from" in-the-wild" human videos," *arXiv preprint arXiv:2103.16817*, 2021.

[12] Q. Zhang, T. Xiao, A. A. Efros, L. Pinto, and X. Wang, "Learning cross-domain correspondence for control with dynamics cycle-consistency," *arXiv preprint arXiv:2012.09811*, 2020.

[13] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1238–1274, 2013.

[14] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke *et al.*, "Scalable deep reinforcement learning for vision-based robotic manipulation," in *Conference on Robot Learning*. PMLR, 2018, pp. 651–673.

[15] X. Gao, F. Chao, C. Zhou, Z. Ge, L. Yang, X. Chang, C. Shang, and Q. Shen, "Error controlled actor-critic," *Information Sciences*, vol. 612, pp. 62–74, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025522009896

[16] P. Sharma, D. Pathak, and A. Gupta, "Third-person visual imitation learning via decoupled hierarchical controller," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[17] A. Gupta, J. Yu, T. Z. Zhao, V. Kumar, A. Rovinsky, K. Xu, T. Devlin, and S. Levine, "Reset-free reinforcement learning via multi-task learning: Learning dexterous manipulation behaviors without human intervention," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6664–6671.

[18] X. Chang and F. Chao, "Robotic action-state evaluation via siamese neural network," in *Robotic Action-state Evaluation via Siamese Neural Network. UKRAS-22 Conference*, 2022.

[19] F. Chao, J. Lv, D. Zhou, L. Yang, C.-M. Lin, C. Shang, and C. Zhou, "Generative adversarial nets in robotic chinese calligraphy," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 1104–1110.

[20] T. Lesort, N. Díaz-Rodríguez, J.-F. Goudou, and D. Filliat, "State representation learning for control: An overview," *Neural Networks*, vol. 108, pp. 379–392, 2018.

[21] S. G. Venkatesh and B. Amrutur, "One-shot object localization using learnt visual cues via siamese networks," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 6700–6705.

[22] X. Zhou, W. Liang, S. Shimizu, J. Ma, and Q. Jin, "Siamese neural network based few-shot learning for anomaly detection in industrial cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5790–5798, 2020.

[23] D. Chicco, "Siamese neural networks: An overview," *Artificial Neural Networks*, pp. 73–94, 2021.

[24] M. Z. Zaheer, A. Mahmood, M. H. Khan, M. Astrid, and S.-I. Lee, "An anomaly detection system via moving surveillance robots with human collaboration," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 2595–2601.

[25] Y. Li, T. Zhang, and C. L. P. Chen, "Enhanced broad siamese network for facial emotion recognition in human–robot interaction," *IEEE Transactions on Artificial Intelligence*, vol. 2, no. 5, pp. 413–423, 2021.

[26] A. Zeng, S. Song, K.-T. Yu, E. Donlon, F. R. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo *et al.*, "Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching," *The International Journal of Robotics Research*, vol. 41, no. 7, pp. 690–705, 2022.

[27] T. Lesort, N. Díaz-Rodríguez, J.-F. Goudou, and D. Filliat, "State representation learning for control: An overview," *Neural Networks*, vol. 108, pp. 379–392, 2018.

[28] A. X. Lee, A. Nagabandi, P. Abbeel, and S. Levine, "Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model," *Advances in Neural Information Processing Systems*, vol. 33, pp. 741–752, 2020.

[29] M. Zhang, S. Vikram, L. Smith, P. Abbeel, M. Johnson, and S. Levine, "Solar: Deep structured representations for model-based reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2019, pp. 7444–7453.

[30] D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson, "Learning latent dynamics for planning from pixels," in *International conference on machine learning*. PMLR, 2019, pp. 2555–2565.

[31] F. Torabi, G. Warnell, and P. Stone, "Behavioral cloning from observation," *arXiv preprint arXiv:1805.01954*, 2018.

[32] W. Sun, A. Vemula, B. Boots, and D. Bagnell, "Provably efficient imitation learning from observation alone," in *International conference on machine learning*. PMLR, 2019, pp. 6036–6045.

[33] A. Edwards, H. Sahni, Y. Schroecker, and C. Isbell, "Imitating latent policies from observation," in *International conference on machine learning*. PMLR, 2019, pp. 1755–1763.

[34] F. Ebert, C. Finn, S. Dasari, A. Xie, A. Lee, and S. Levine, "Visual foresight: Model-based deep reinforcement learning for vision-based robotic control," *arXiv preprint arXiv:1812.00568*, 2018.

[35] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain, "Time-contrastive networks: Self-supervised learning from video," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 1134–1141.