International Symposium on Medical Robotics, Atlanta, USA, April 2023.

Towards Safe and Efficient Reinforcement Learning for Surgical Robots Using Real-Time Human Supervision and Demonstration

Yafei Ou and Mahdi Tavakoli, *Senior Member, IEEE*

***Abstract*— Recent research in surgical robotics has focused on increasing the level of autonomy in order to reduce the workload of surgeons. While deep reinforcement learning (DRL) has shown promising results in automating some surgical subtasks, due to its demand for a large number of random explorations, safety and learning efficiency remain the primary challenges when applying it to surgical robot learning. In this work, we present a DRL framework with real-time human supervision during the training process for surgical robot learning to avoid significant failures and speed up training. A novel training methodology based on the combination of DRL and generative adversarial imitation learning (GAIL) is proposed to further improve learning efficiency by imitating human behaviors. The proposed method is validated using two simulated environments, where human intervention is performed via teleoperation. Results show that our method outperforms baseline algorithms and can achieve safe and efficient learning.**

I. INTRODUCTION

Among the various approaches to the automation of surgical robotic systems, machine learning-based approaches have gained increasing attention due to their generalizability and adaptability to complex tasks. Compared with hand-crafting task-specific control policies, these methods require less human knowledge and understanding of the task thanks to their data-driven nature. Reinforcement learning (RL), or more specifically, deep reinforcement learning (DRL) that utilizes deep neural networks as function approximators for RL, is one of the most frequently investigated learning-based approaches to automating surgical tasks in recent research and has already shown promising results in some surgical subtasks such as needle regrasping and tissue retraction [1], [2], [3], [4], [5].

As the learning agent in RL explores an environment and improves its policy based on reward feedback, all that is usually required of humans is a well-designed reward function. Although it is obvious that this approach significantly reduces the need for the understanding of the task, it requires a large number of explorations before it can learn a good policy. This problem is even worse in the case of complex surgical scenarios such as tissue manipulation or needle passing, where millions of steps of exploration can be necessary. Depending on the complexity of the task and the algorithm used, learning a good policy for a given task can often take from several hours to a few days. As a result, sample efficiency is one of the major issues when using RL for surgical robot learning. Furthermore, as the agent has no prior knowledge about the environment, random explorations during training may result in catastrophic failures, such as damage to the soft tissue. While it is possible to train a policy in the simulation and transfer it to the real environment, additional explorations in the real world are usually needed for fine-tuning the policy if the dynamics of the environment is complex [6]. This raises another issue when using RL in surgical robots, which is safety.

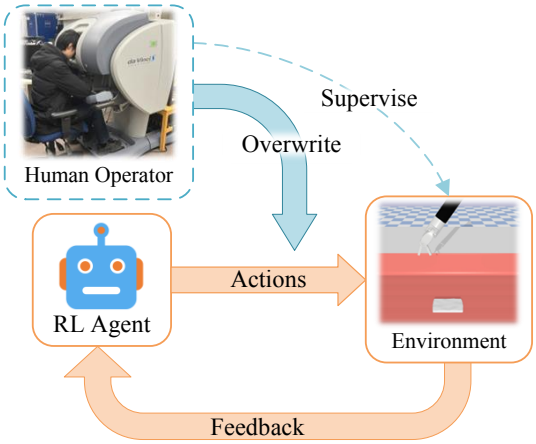


Fig. 1. Conceptual framework of DRL with real-time human guidance for surgical robot learning.

An intuitive approach to accelerating training and ensuring safe exploration is to incorporate more human knowledge. For example, a human expert can provide evaluative feedback by labeling how good an action taken by the agent is. This feedback can be directly applied to guide policy updates [7] or indirectly used as an additional reward signal [8] in order to speed up training. To avoid catastrophic failures, a safety critic [9] or an action blocker [10] can be trained based on human knowledge of dangerous situations, which prevents the agent from taking actions that can lead to catastrophes.

Leveraging real-time human intervention is a more straightforward approach that can both accelerate training and avoid catastrophes at the same time. During training, a human expert supervises the training process and occasionally takes over control by overwriting agent actions to avoid dangers. Additionally, the human can assist the agent in overcoming task performance bottlenecks by guiding it to an unseen state with better rewards, which is particularly important when the reward signal is so sparse that it is

This research was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada and the China Scholarship Council (CSC). *(Corresponding author: Yafei Ou.)*

Yafei Ou and Mahdi Tavakoli are with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, Canada (e-mail: yafei.ou@ualberta.ca; mahdi.tavakoli@ualberta.ca)

difficult for the agent to receive any positive rewards during random explorations. In fact, this is often the case in complex surgical tasks such as needle passing and knot tying, where designing a dense reward is challenging and only a sparse reward can be provided indicating whether the goal has been achieved or not.

Real-time human intervention or guidance in RL can be viewed as intermittent human demonstrations during training, as opposed to gathering them beforehand, as is the norm in the field of imitation learning or learning from demonstration (LfD). Different from LfD, RL with real-time human guidance allows the human to only provide important demonstrations when necessary without the need for demonstrating the whole task. Furthermore, this approach makes the training process much safer by including a human supervisor as the agent explores. Despite the fact that RL has been combined with LfD and applied to surgical robots to accelerate training [1], [11], RL with real-time human guidance has not been investigated in the context of surgical robot automation to the best of the authors' knowledge.

In this work, instead of focusing on automating a specific surgical task, we present a general DRL framework that leverages human guidance for teleoperated surgical robotic systems. Inspired by recent advances in generative adversarial imitation learning (GAIL) [12], we propose a novel training methodology based on the combination of DRL and LfD by introducing a discriminator and an imitation loss to further improve sample efficiency and accelerate learning. The main contributions of this work are: (1) we build a human-guided DRL setup for teleoperated surgical robots in a simulated environment, where a joystick controller and a real master tool manipulator (MTM) from the da Vinci Research Kit (dVRK) [13] provided by Intuitive Surgical, Inc. are used for human guidance through teleoperation; (2) we propose a novel training methodology for DRL with real-time human interventions by combining DRL with LfD; (3) we validate the performance of the proposed methodology using two experimental tasks in the simulated environment. A conceptual illustration of the framework is shown in Fig 1.

This paper is organized as follows. Section II provides a brief review of related research. Section III summarizes the mathematical preliminaries of our proposed method. In Section IV, we introduce the proposed training methodology. In Section V, we describe the experimental setup for validating the proposed method. The results are presented and discussed in Section VI. Lastly, concluding remarks and potential future work are provided in Section VII.

II. RELATED WORK

A. DRL with Real-time Human Intervention

Human intervention is an effective approach for increasing sample efficiency and preventing catastrophes during DRL training. Saunders et al. [9] proposed a straightforward training mechanism in which the human monitors the training process and overwrites agent actions when in dangerous situations, and a penalty is assigned when human intervention occurs. In addition, an action blocker is trained based on human interventions to automatically block dangerous actions, which eventually replaces the human. Wang et al. [14] developed an algorithm for RL with human intervention by modifying the loss of proximal policy optimization (PPO), which accelerates the training process. The method was extended to off-policy methods such as deep deterministic policy gradient (DDPG) [15] and twin-delayed DDPG (TD3) [16] with improvements. Although these methods have shown success in video games, unmanned aerial vehicles, and autonomous vehicles, DRL with real-time human interventions has not been exploited for the purpose of automating teleoperated surgical robots.

B. DRL and LfD for Surgical Robots

The automation of surgical robots using DRL and LfD has gained increasing attention in recent years, and various simulated environments have been developed for this purpose [2], [17], [18], [19]. A number of recent works have focused on automating surgical subtasks that commonly exist during surgeries in order to relieve surgeons of tedious and repetitive work. For instance, Tagliabue et al. [2] trained a policy using PPO for the robot to grasp and lift the tissue to reveal a region of interest underneath it. Li et al. [20] applied GAIL, an LfD approach, to the automation of laparoscope motion during surgery. Our group has previously applied LfD approaches to the automation of rehabilitation robots [21], [22], [23], [24], [25], [26].

Since LfD takes advantage of demonstrations from human experts, it is often incorporated into DRL to achieve better performance. In a follow-up study of [2], Pore et al. [11] combined GAIL with PPO to achieve a faster learning speed. Chiu et al. [1] used DDPG to learn autonomous bimanual needle regrasping, where behavior cloning (BC), a simple LfD approach, was utilized to help exploration. These methods incorporate LfD into DRL by using human demonstrations collected prior to training, which is different from this work where the human can start or stop intervention at any time during the training process.

III. BACKGROUND

A. Soft Actor-Critic (SAC)

In this subsection, we recall soft actor-critic (SAC), an off-policy DRL algorithm introduced in [27], which will act as a backbone of our proposed method. Compared with on-policy algorithms, off-policy algorithms are known to be more sample efficient and require less exploration, thanks to the usage of an experience replay buffer that stores all the experienced transitions, thus more suitable for our application.

An RL problem can be formulated as a Markov decision process (MDP) described by a five-tuple $(\mathcal{S},\mathcal{A},P,R,\gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, P: $\mathcal{S}\times\mathcal{A}\times\mathcal{S}\to[0,1]$ is the state transition function, R:$\mathcal{S}\times\mathcal{A}\to\mathbb{R}$ is the reward function, and $\gamma\in[0,1]$ is the discount factor. SAC algorithm considers the maximum entropy reinforcement learning problem whose learning objective is to find

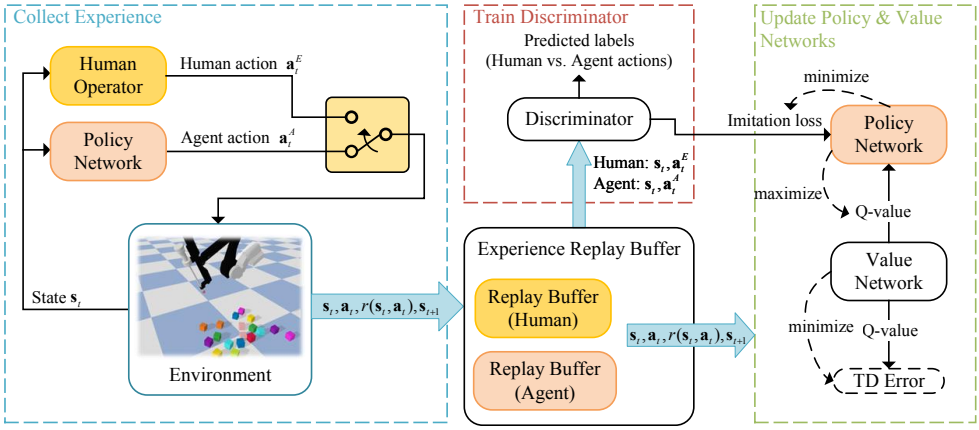


Fig. 2. Framework of the proposed human-guided RL scheme.

an optimal policy that maximizes the expectation of the cumulative reward and the policy entropy at the same time:

$$\pi^*=\underset{\pi}{\operatorname{argmax}}\sum_{t=0}^{T}\mathbb{E}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\rho_\pi}\left[\gamma^t r(\mathbf{s}_t,\mathbf{a}_t)+\alpha\mathcal{H}(\pi(\cdot|\mathbf{s}_t))\right] \quad (1)$$

where π is the policy to be optimized, ρ_π represents the trajectory distribution produced by the policy π, T is the horizon, $r(\mathbf{s}_t,\mathbf{a}_t)$ is the reward for the state-action pair $(\mathbf{s}_t,\mathbf{a}_t)$ at time step t, $\mathcal{H}(\pi(\cdot|\mathbf{a}_t))$ is the entropy of the action distribution under the state $\mathbf{s}_t$, and α is a weighting factor. Considering maximum entropy encourages exploration and enables learning a more robust policy [27].

SAC algorithm exploits an actor-critic structure. The critic is a function approximator $Q_\theta(\mathbf{s}_t,\mathbf{a}_t)$ parameterized by θ for estimating the soft Q-value (action-value), indicating how good an action $\mathbf{a}_t$ taken at $\mathbf{s}_t$ is, and the actor is the policy π_ϕ parameterized by ϕ that generates actions from given states. $Q_\theta(\mathbf{s}_t,\mathbf{a}_t)$ is trained using the temporal difference (TD) target

$$\hat{y}_t=r(\mathbf{s}_t,\mathbf{a}_t)+\gamma\mathbb{E}_{\mathbf{s}_{t+1}\sim p}[V_\theta(\mathbf{s}_{t+1})] \quad (2)$$

where $V_\theta(\mathbf{s}_t)$ is the soft state-value function implicitly parameterized by θ [27]. Therefore, θ can be updated by minimizing the Bellman residual

$$J_Q(\theta)=\mathbb{E}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\mathcal{R}}\left[\frac{1}{2}\left(Q_\theta(\mathbf{s}_t,\mathbf{a}_t)-\hat{y}_t\right)^2\right] \quad (3)$$

where $\mathcal{R}$ is the trajectories stored in the experience replay buffer. The policy π_ϕ is encouraged to generate actions that maximize the sum of the soft Q-value predicted by the critic and the α-weighted policy entropy. Therefore, the parameters ϕ can be updated by minimizing the loss

$$J_\pi(\phi)=\mathbb{E}_{\mathbf{s}_t\sim\mathcal{R}}\left[\mathbb{E}_{\mathbf{a}_t\sim\pi_\phi}[-Q_\theta(\mathbf{s}_t,\mathbf{a}_t)+\alpha\log(\pi(\mathbf{a}_t|\mathbf{s}_t))]\right] \quad (4)$$

In practice, both the actor and the critic are implemented using neural networks. To mitigate the overestimation problem and stabilize training, two Q networks (Q_{θ_1} and Q_{θ_2}) and two target networks ($Q_{\bar{\theta}_1}$ and $Q_{\bar{\theta}_2}$) are used.

B. Behavior Cloning (BC)

Learning from demonstration (LfD), or imitation learning, aims to learn a policy directly from human demonstrations, without knowing the reward function $r(\mathbf{s}_t,\mathbf{a}_t)$. Behavior cloning (BC) is a simple supervised learning approach for LfD. In behavior cloning, human demonstrations are collected and the state-action tuples are stored in a dataset $\mathcal{R}_E$. A policy π_ϕ parameterized by ϕ, usually a neural network, is trained by minimizing the BC loss:

$$J_\pi^{BC}(\phi)=\mathbb{E}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\mathcal{R}_E}\left[\|\pi(\mathbf{s}_t)-\mathbf{a}_t\|^2\right] \quad (5)$$

C. Generative Adversarial Imitation Learning (GAIL)

Generative adversarial imitation learning (GAIL) is an LfD algorithm that has recently shown great promise. Inspired by the idea of generative adversarial networks (GAN), GAIL utilizes a discriminator D_φ parameterized by φ to discriminate between expert human actions and the actions taken by the learning agent. The human demonstrations are stored in the dataset $\mathcal{R}_E$, and the trajectories generated by the agent during training are stored in another dataset $\mathcal{R}_A$. The discriminator and the agent are trained in an adversarial manner. The discriminator takes in the state-action pair as input and is trained to predict whether the action is taken by the human or by the agent, while the agent is trained towards cheating the discriminator by taking actions that are close to the human expert. The loss for the discriminator D_φ is

$$\begin{aligned}J_D^{GAIL}(\varphi)=&\mathbb{E}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\mathcal{R}_E}[\log D_\varphi(\mathbf{s}_t,\mathbf{a}_t)]+\\&\mathbb{E}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\mathcal{R}_A}[\log(1-D_\varphi(\mathbf{s}_t,\mathbf{a}_t))]\end{aligned} \quad (6)$$

After updating the discriminator, we can re-sample state-action pairs $(\mathbf{s}_t,\mathbf{a}_t)$ from $\mathcal{R}_A$ and use the output of the discriminator $D_\varphi(\mathbf{s}_t,\mathbf{a}_t)$ as the predicted rewards. Thereby, standard RL algorithms can now be used to learn a policy π_ϕ. Alternating between updating the discriminator and applying standard RL updates will eventually result in a policy that is close to the human.

Although on-policy RL algorithms such as PPO are more often used in GAIL, recent work has shown that GAIL can also be adapted for off-policy algorithms such as DDPG.

When using off-policy algorithms, the value network is updated directly using the predicted reward $D_\varphi(\mathbf{s}_t,\mathbf{a}_t)$ without changing the loss function, while the policy network can be updated by adding an imitation loss term to the original loss

$$J_\pi^{GAIL}(\phi)=J_\pi(\phi)+\mathbb{E}_{\mathbf{s}_t\sim\mathcal{R}_A}\left[\mathbb{E}_{\mathbf{a}_t\sim\pi_\phi}[-\omega\log D_\varphi(\mathbf{s}_t,\mathbf{a}_t)]\right] \tag{7}$$

where ω is a weighting factor. Updating the policy using $J_\pi^{GAIL}(\phi)$ will encourage the policy to generate actions close to the human demonstrations and accelerate the speed of convergence [28].

IV. PROPOSED METHOD

A. Leveraging Human Guidance

To incorporate human guidance in reinforcement learning, a human expert monitors the training process and provides guidance when necessary by directly overwriting the actions taken by the agent. Therefore, the actual action taken during training can be expressed by

$$\mathbf{a}_t=\mathcal{I}(\mathbf{s}_t)\mathbf{a}_t^E+(1-\mathcal{I}(\mathbf{s}_t))\mathbf{a}_t^A \tag{8}$$

where $\mathcal{I}(\mathbf{s}_t)\in\{0,1\}$ is a function representing whether the human intervenes or not, $\mathbf{a}_t^E$ is the action taken by the human, and $\mathbf{a}_t^A$ is the action taken by the RL agent.

With the real actions stored in the replay buffer $\mathcal{R}$, we can directly apply SAC algorithm using (2)-(4). However, simply replacing agent actions with human actions without further modification of the learning structure results in poor performance. This is due to the fact that the critic is always updated according to the trajectories extracted from the replay buffer $\mathcal{R}$, which includes both agent and human trajectories, while the actor loss is computed using the on-policy actions predicted by the current policy. Prior work proposes adding a BC loss term to the policy loss to encourage the policy to imitate human actions when human intervention occurs [14], [15]. Specifically for SAC, (4) can be modified by adding a BC loss,

$$\begin{aligned}J_\pi(\phi)=&\mathbb{E}_{\mathbf{s}_t\sim\mathcal{R}}\left[\mathbb{E}_{\mathbf{a}_t\sim\pi_\phi}[-Q_\theta(\mathbf{s}_t,\mathbf{a}_t)+\alpha\log(\pi(\mathbf{a}_t|\mathbf{s}_t))]\right]\\&+\mathbb{E}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\mathcal{R}_E}\left[\omega\cdot\|\pi(\mathbf{s}_t)-\mathbf{a}_t\|^2\right]\end{aligned} \tag{9}$$

where ω is a weighting factor and $\mathcal{R}_E$ is the buffer that stores the trajectories with human intervention.

Instead of adding a BC loss, in this work we incorporate the idea from GAIL by training a discriminator $D_\varphi(\mathbf{s}_t,\mathbf{a}_t)$ to discriminate between human and agent actions and use the predicted value as the imitation loss added to the policy loss. During training, the trajectories produced by the RL agent and the human are stored in two separate replay buffers $\mathcal{R}_A$ and $\mathcal{R}_E$ respectively, and the discriminator is trained by minimizing the classification loss, as in (6):

$$\begin{aligned}J_D(\varphi)=&\mathbb{E}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\mathcal{R}_E}[\log D_\varphi(\mathbf{s}_t,\mathbf{a}_t)]+\\&\mathbb{E}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\mathcal{R}_A}[\log(1-D_\varphi(\mathbf{s}_t,\mathbf{a}_t))]\end{aligned} \tag{10}$$

The critic is updated directly using (3) without modification:

$$J_Q(\theta)=\mathbb{E}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\mathcal{R}}\left[\frac{1}{2}(Q_\theta(\mathbf{s}_t,\mathbf{a}_t)-\hat{y}_t)^2\right] \tag{11}$$

Here, the replay buffer $\mathcal{R}$ now contains all the trajectories produced by both the agent and the human. Similar to (7), an imitation loss term is added to the policy loss:

$$\begin{aligned}J_\pi(\phi)=\mathbb{E}_{\mathbf{s}_t\sim\mathcal{R}}\big[&\mathbb{E}_{\mathbf{a}_t\sim\pi_\phi}[-Q_\theta(\mathbf{s}_t,\mathbf{a}_t)+\alpha\log(\pi(\mathbf{a}_t|\mathbf{s}_t))\\&-\omega\log D_\varphi(\mathbf{s}_t,\mathbf{a}_t)]\end{aligned} \tag{12}$$

Based on prior work in the context of LfD [11], it is intuitive that this modification will encourage the agent to imitate human behaviors and can achieve faster convergence, as will be shown through experiments in Section V.

B. Implementation Details

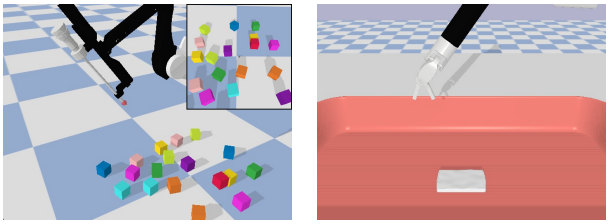
The proposed human-guided RL framework is shown in Fig. 2. The detailed procedure is summarized in Algorithm 1.

Algorithm 1: Human guided reinforcement learning

```
Initialize actor network π_φ, critic networks Q_θ1, Q_θ2,
  discriminator network D_φ;
Initialize target networks Q_θ̄1=Q_θ1, Q_θ̄2=Q_θ2;
Initialize empty human replay buffer R_E and empty
  agent replay buffer R_A, R≡R_E∪R_A;
for each iteration do
    for each environment step do
        a_t^A ~ π_φ(s_t)                    ▷ Sample agent action
        if human intervenes then
            a_t ← a_t^E;
            R_store ← R_E                   ▷ Set R_E as the replay
              buffer to store the transition
        else
            a_t ← a_t^A;
            R_store ← R_A                   ▷ Set R_A as the replay
              buffer to store the transition
        end
        s_{t+1} ~ p(s_{t+1}|s_t,a_t)        ▷ Sample transition from
          the environment
        R_store ← R_store ∪ {s_t,a_t,r(s_t,a_t),s_{t+1}}  ▷ Store
          transition
    end
    if train discriminator now then
        for each discriminator gradient step do
            Update D_φ using Equation (10)
        end
    end
    for each policy gradient step do
        Update Q_θ1, Q_θ2 using Equation (11);
        Update π_φ using Equation (12) ;
        θ̄_i ← τθ_i+(1−τ)θ̄_i for i∈{1,2}     ▷ Update the
          target networks using Polyak averaging
    end
end
```

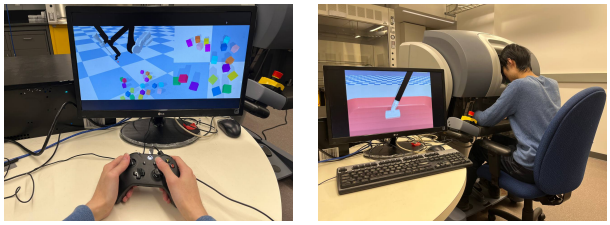
V. EXPERIMENTAL SETUP

To validate the proposed human-guided reinforcement learning scheme for automating surgical robots, we design two different tasks in a simulated environment based on SurRol [19], which simulates the dVRK medical robotic system.



(a) (b)

Fig. 3. Task environments: (a) ActiveTrack; (b) GauzeRetrieve (Modified).



(a) (b)

Fig. 4. Experimental setup using teleoperation: (a) ActiveTrack using a joystick controller; (b) GauzeRetrieve using MTM.

A. Tasks

1) ActiveTrack: The ActiveTrack environment proposed in [19] is used without modification in this work. The goal of the task is for the endoscopic camera manipulator (ECM) to keep tracking a red cube moving in a 2D plane, as shown in Fig 3a. The action is the camera velocity in its own frame coordinate ${}^{c}V_{c}$. The observation includes the robot pose and the object pose in the Cartesian space. The reward function is

$$r(\mathbf{s}_t,\mathbf{a}_t)=C-(\|p_t^{ij}-p_c\|_2+\lambda\cdot|\theta^*|) \tag{13}$$

where $C=1$ and $\lambda=0.1$. While the maximum number of steps in each episode during training is 500, we reduce the number to 200 during the evaluation to eliminate the repeated motion of the object. It is also worth noting that early termination will be triggered when the camera totally loses track of the object, although there will be no penalty for this situation. The purpose of choosing this task is to test the performance of the proposed method when using environments with dense rewards and to examine the capability of the proposed method in learning to avoid significant failures, i.e. losing track of the object in camera images.

2) GauzeRetrieve (Modified): We build a modified version of the original GauzeRetrieve task, where the patient side manipulator (PSM) needs to grasp a piece of gauze and lift it above a certain height. The movement of the PSM is restricted in a 2D plane, and the orientation of the end-effector (EE) is locked. The initial position of the PSM is randomized between each episode. The action contains 3 elements in continuous space, including the movement of the EE in the 2D space, plus the closing or opening of the gripper. The observation space is the same as the original task environment, which includes the robot pose and the object pose in the Cartesian space, and the position of the object relative to the EE. The environment will return a sparse reward of 100 when the gauze is lifted above a certain height; otherwise, the reward is zero. A screenshot of the environment is shown in Fig 3b. The purpose of choosing this task is to verify the proposed method when using environments with sparse rewards and requiring human guidance to help overcome a bottleneck in the task performance.

B. Baseline Algorithms

For a comparison, we also implement several related baseline algorithms.

1) IA-SAC: Intervention-aided reinforcement learning (IARL) [14] and human-guidance-based deep reinforcement learning (Hug-DRL) [15] add a behavior cloning loss (BC loss) to the policy loss for the human-intervened state-action pairs. The original method was implemented based on PPO and was re-implemented for DDPG in [15]. In this work, we re-implement this method based on SAC and name it IA-SAC.

2) HI-SAC: Human intervention reinforcement learning (HIRL) is derived from [9]. In this method, the human directly overwrites agent actions while no modification is made to the learning algorithm. We re-implement this method based on SAC and name it HI-SAC.

3) Standard SAC: This method is the standard SAC algorithm without human intervention.

The hyperparameters for each method are set to be the same and the imitation weight $\omega=4$. It is worth noting that although in some of the related methods (and their improvements), training techniques such as penalizing human interventions and auto-tuning weighting parameters are exploited, we do not include these implementations for a fair comparison, because these approaches are also applicable to our method and can be implemented in future work.

C. Human Guidance using Teleoperation

Human interventions are achieved through teleoperation using a joystick controller and an MTM of the dVRK, as shown in Fig 4. To ensure a fair comparison, the interventions of the human supervisor follow a similar pattern regarding their quality and timing during each training instance, where the frequency of human intervention is reduced throughout training, and all the interventions are done in the first half of the training process. In addition, the human provides approximately 200 steps of initial full demonstrations in the GauzeRetrieve task in order to help the agent receive positive reward feedback.

VI. RESULTS AND DISCUSSION

For the ActiveTrack task, we train using each of the methods for 10,000 steps and repeat for 3 instances (trials). The number of total human interventions is fixed at 500 steps for all methods and training instances for a fair comparison. The learning curves of the DRL algorithms are shown in 5. As shown, since the reward signal is dense, a standard DRL agent can reach a relatively high return after being trained

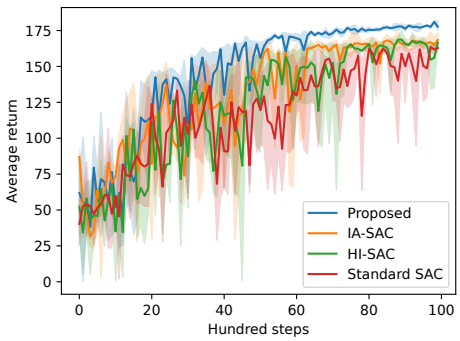


Fig. 5. Learning curves for the ActiveTrack task. We train 3 instances for each method and evaluate for 5 episodes every 100 steps. The solid lines are the mean values and the shaded areas are the standard deviations.

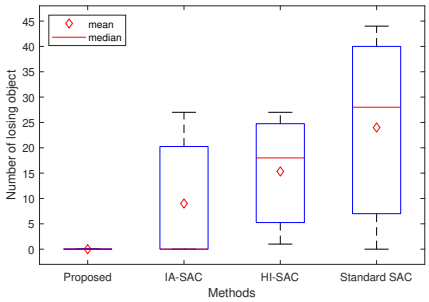


Fig. 6. Total number of losing the view of the object during evaluation starting from 5,000 steps. Box plot using data from 3 different instances of training for each method.

for 10,000 steps without human intervention. However, the methods with human interventions outperform the standard DRL method and achieve much faster learning. Moreover, the proposed method outperforms the two other baseline methods that utilize human guidance (IA-SAC and HI-SAC) from the aspect of the convergence speed and final return achieved after training for 10,000 steps. This shows that the proposed method is more efficient compared with the baseline algorithms and aligns with our expectation that using a discriminator and imitation loss will help drive the agent toward learning human behaviors. In addition, fewer fluctuations, which are caused by the early termination scheme when the camera loses the object completely, exist in the learning curve of our method after training for 50,000 steps. As the human operator intervenes when the camera is about to lose the object, by imitating human behaviors, our method learns to avoid significant failures more efficiently. To further investigate the matter, we count the total number of times of completely losing the view of the object during the evaluation phase starting from 5,000 steps, as shown in Fig 6. We saw no failures in any of the training instances after 5,000 training steps when using the proposed method, while IA-SAC and HI-SAC saw a number of failures ranging from 0 to 30 among the training instances (IA-SAC has a slightly better performance than HI-SAC). The standard SAC algorithm yields the highest number of failures.

For the GauzeRetrieve task, we train using each of the methods for 3 different instances with 40,000 steps. We evaluate the policy every 1,000 steps for 100 episodes and use the policy that yields the highest success rate for comparison. In addition, to examine the effect of using different numbers of human interventions during training, we limit the total number of human interventions to 1,000, 1,500, and 2,000 and compare the resulting best policy. The results are shown in Fig 7. As expected, standard SAC without human intervention cannot learn the task, since it is almost impossible for the agent to receive any positive rewards with random explorations. For all three methods with human interventions, there is a trend where more human interventions result in better performance. While the proposed method has only a small advantage over the other two baselines when the total number of interventions is 1,000, it yields a much higher success rate when the number of interventions is 1,500 and 2,000. Although variations of other factors between each training instance such as the quality and the timing of human interventions are inevitable and may affect the results, it is nonetheless reasonable to state that the proposed method is generally more efficient in learning when a sufficient number of human interventions is allowed.

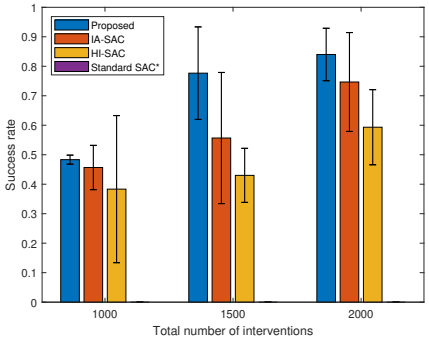


Fig. 7. Success rate of GauzeRetrieve with different number of human interventions. Error bars represent the standard deviations. (*: Standard SAC always yields zero success rate and the bars are not visible in the graph.)

VII. CONCLUSIONS

In this work, we presented a DRL framework for surgical robot learning that leverages real-time human supervision during training to speed up the training process and avoid significant failures. A novel training methodology that combines the DRL algorithm with LfD was proposed to further accelerate learning by encouraging the agent to imitate human behaviors. Experimental results in simulated environments show that the proposed method achieves safe and efficient learning for surgical robots, outperforming the compared baseline algorithms. Since human supervision is utilized during training to speed up learning and avoid danger, the proposed method has a strong potential for application to surgical robot learning in the real world. While this work features simple learning tasks in the simulated environment, future work will include extending the method to real surgical scenarios and more specific surgical tasks, such as soft tissue manipulation and needle regrasping. Additional research is also needed to investigate the feasibility of employing experienced physicians as supervisors.

REFERENCES

[1] Z.-Y. Chiu, F. Richter, E. K. Funk, R. K. Orosco, and M. C. Yip, "Bimanual regrasping for suture needles using reinforcement learning for rapid motion planning," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 7737–7743.

[2] E. Tagliabue, A. Pore, D. Dall'Alba, E. Magnabosco, M. Piccinelli, and P. Fiorini, "Soft tissue simulation environment to learn manipulation tasks in autonomous robotic surgery," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 3261–3266.

[3] S. C. Bacha, W. Bai, Z. Wang, B. Xiao, and E. M. Yeatman, "Deep reinforcement learning-based control framework for multilateral telesurgery," *IEEE Transactions on Medical Robotics and Bionics*, vol. 4, no. 2, pp. 352–355, 2022.

[4] G. Ji, J. Yan, J. Du, W. Yan, J. Chen, Y. Lu, J. Rojas, and S. S. Cheng, "Towards safe control of continuum manipulator using shielded multiagent reinforcement learning," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7461–7468, 2021.

[5] R. Zhu, D. Zhang, and B. Lo, "Deep reinforcement learning based semi-autonomous control for robotic surgery," *arXiv preprint arXiv:2204.05433*, 2022.

[6] J. Van Baar, A. Sullivan, R. Cordorel, D. Jha, D. Romeres, and D. Nikovski, "Sim-to-real transfer learning using robustified controllers in robotic tasks involving complex dynamics," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6001–6007.

[7] S. Griffith, K. Subramanian, J. Scholz, C. L. Isbell, and A. L. Thomaz, "Policy shaping: Integrating human feedback with reinforcement learning," *Advances in neural information processing systems*, vol. 26, 2013.

[8] G. Warnell, N. Waytowich, V. Lawhern, and P. Stone, "Deep tamer: Interactive agent shaping in high-dimensional state spaces," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[9] W. Saunders, G. Sastry, A. Stuhlmueller, and O. Evans, "Trial without error: Towards safe reinforcement learning via human intervention," *arXiv preprint arXiv:1707.05173*, 2017.

[10] B. Thananjeyan, A. Balakrishna, S. Nair, M. Luo, K. Srinivasan, M. Hwang, J. E. Gonzalez, J. Ibarz, C. Finn, and K. Goldberg, "Recovery rl: Safe reinforcement learning with learned recovery zones," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4915–4922, 2021.

[11] A. Pore, E. Tagliabue, M. Piccinelli, D. Dall'Alba, A. Casals, and P. Fiorini, "Learning from demonstrations for autonomous soft-tissue retraction," in *2021 International Symposium on Medical Robotics (ISMR)*. IEEE, 2021, pp. 1–7.

[12] J. Ho and S. Ermon, "Generative adversarial imitation learning," *Advances in neural information processing systems*, vol. 29, 2016.

[13] P. Kazanzides, Z. Chen, A. Deguet, G. S. Fischer, R. H. Taylor, and S. P. DiMaio, "An open-source research kit for the da vinci® surgical system," in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 6434–6439.

[14] F. Wang, B. Zhou, K. Chen, T. Fan, X. Zhang, J. Li, H. Tian, and J. Pan, "Intervention aided reinforcement learning for safe and practical policy optimization in navigation," in *Conference on Robot Learning*. PMLR, 2018, pp. 410–421.

[15] J. Wu, Z. Huang, Z. Hu, and C. Lv, "Toward human-in-the-loop ai: Enhancing deep reinforcement learning via real-time human guidance for autonomous driving," *Engineering*, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2095809922004878

[16] J. Wu, Z. Huang, W. Huang, and C. Lv, "Prioritized experience-based reinforcement learning with human guidance for autonomous driving," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[17] F. Richter, R. K. Orosco, and M. C. Yip, "Open-sourced reinforcement learning environments for surgical robotics," *arXiv preprint arXiv:1903.02090*, 2019.

[18] V. M. Varier, D. K. Rajamani, F. Tavakkolmoghaddam, A. Munawar, and G. S. Fischer, "Ambf-rl: A real-time simulation based reinforcement learning toolkit for medical robotics," in *2022 International Symposium on Medical Robotics (ISMR)*. IEEE, 2022, pp. 1–8.

[19] J. Xu, B. Li, B. Lu, Y.-H. Liu, Q. Dou, and P.-A. Heng, "Surrol: An open-source reinforcement learning centered and dvrk compatible platform for surgical robot learning," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021.

[20] B. Li, R. Wei, J. Xu, B. Lu, C. H. Yee, C. F. Ng, P.-A. Heng, Q. Dou, and Y.-H. Liu, "3d perception based imitation learning under limited demonstration for laparoscope control in robotic surgery," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 7664–7670.

[21] M. Maaref, A. Rezazadeh, K. Shamaei, R. Ocampo, and T. Mahdi, "A bicycle cranking model for assist-as-needed robotic rehabilitation therapy using learning from demonstration," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 653–660, 2016.

[22] M. Najafi, M. Sharifi, K. Adams, and M. Tavakoli, "Robotic assistance for children with cerebral palsy based on learning from tele-cooperative demonstration," *International Journal of Intelligent Robotics and Applications*, vol. 1, no. 1, pp. 43–54, 2017.

[23] C. Martínez and M. Tavakoli, "Learning and robotic imitation of therapist's motion and force for post-disability rehabilitation," in *2017 IEEE international conference on systems, man, and cybernetics (SMC)*. IEEE, 2017, pp. 2225–2230.

[24] J. Fong, R. Ocampo, D. P. Gross, and M. Tavakoli, "A robot with an augmented-reality display for functional capacity evaluation and rehabilitation of injured workers," in *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 2019, pp. 181–186.

[25] J. Fong, H. Rouhani, and M. Tavakoli, "A therapist-taught robotic system for assistance during gait therapy targeting foot drop," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 407–413, 2019.

[26] C. Martinez and M. Tavakoli, "Learning and reproduction of therapist's semi-periodic motions during robotic rehabilitation," *Robotica*, vol. 38, no. 2, pp. 337–349, 2020.

[27] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel *et al.*, "Soft actor-critic algorithms and applications," *arXiv preprint arXiv:1812.05905*, 2018.

[28] G. Zuo, K. Chen, J. Lu, and X. Huang, "Deterministic generative adversarial imitation learning," *Neurocomputing*, vol. 388, pp. 60–69, 2020.