

# Data-Efficient Learning from Human Interventions for Mobile Robots

Zhenghao Peng<sup>1</sup>, Zhizheng Liu<sup>1</sup>, Bolei Zhou<sup>1</sup>

**Abstract**—Mobile robots are essential in applications such as autonomous delivery and hospitality services. Traditional rule-based and planning-based methods for navigation face challenges in dynamic environments due to the needs of calibration, environment modeling and a lack of robustness against unpredictable changes. Learning-based methods such as Imitation Learning (IL) and Reinforcement Learning (RL) offer adaptability but require large datasets, carefully crafted reward functions, and face sim-to-real gaps, making them challenging for efficient and safe real-world deployment. We propose an online human-in-the-loop learning method that combines IL and RL to address these issues. Our approach enables efficient real-time policy learning from online human intervention, significantly improving data efficiency and training safety. We validate our method by training two different robots — a legged quadruped, and a wheeled delivery robot — in two mobile robot tasks, one of which even uses raw RGBD image as observation. The training finishes . Our experiments demonstrate that online human-in-the-loop learning can achieve high efficiency and effectiveness in real-world robotic training.

## I. INTRODUCTION

Mobile robots have become increasingly important in a variety of applications, ranging from autonomous delivery [1] in urban environments to service robots in healthcare [2] and hospitality [3]. Previously, these robots have relied on rule-based [4] and planning-based [5] methods to navigate and perform tasks. However, while these methods have been effective for structured environments, they face significant challenges in dynamic or unstructured settings due to the cost of sensor calibration, environmental modeling, and complex engineering. Moreover, they are not as robust when dealing with unpredictable human behaviors or environmental changes, often leading to failures.

In contrast, learning-based methods offer more robustness and adaptability by allowing robots to learn from experience rather than relying on pre-defined models of the environment. Two common learning-based approaches are Imitation Learning (IL) and Reinforcement Learning (RL). IL trains robots by mimicking human demonstrations [6], [7], [8], [9], [10], allowing them to learn complex behaviors directly from experts. However, large amounts of demonstration data are required, which is expensive and time-consuming to collect. If learning from insufficient data, the learned agents can easily overfit the data. IL methods also suffer from compounding errors due to the distribution shifts between the training data and real-world environments, which is more prominent when the data is insufficient. On the other hand, RL involves training robots through trial and error by interacting with the environment and optimizing a reward

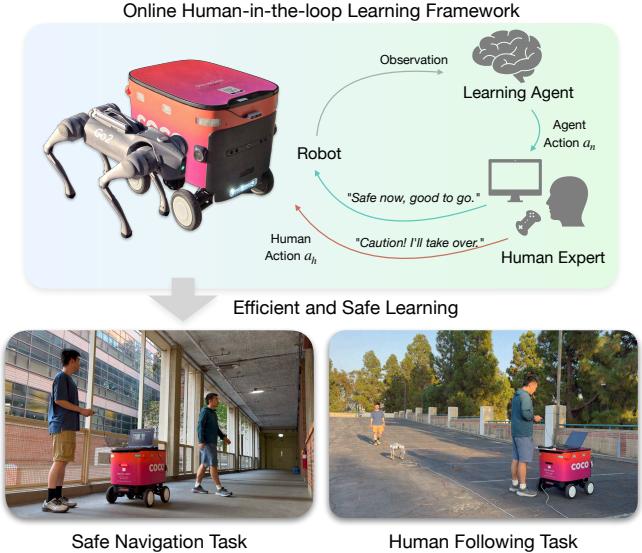


Fig. 1: We train mobile robots in dynamic environments with a human-in-the-loop learning method. We can solve challenging tasks with raw camera input in . Our method can work with different robots and observation types, showing the promising future of human-in-the-loop learning to address the data efficiency issue in real-world robotic tasks.

function. RL methods often depend on carefully engineered reward functions, which can be difficult to design and tune for complex, real-world tasks. Also, the learned agent may obtain biased behaviors or figure out the loophole to finish the task [11], [12], [13]. Additionally, there is the sim-to-real gap problem, where policies trained in simulation do not transfer well to the real world due to the noisy sensory data and differences in the environments. Furthermore, training RL models directly in the real world is risky because it requires extensive exploration that could lead to unsafe actions, such as collisions with surroundings. Given these challenges, there is a strong need for more data-efficient and safer learning methods for robotic training in the real world.

Human-in-the-loop approaches, which involve active human intervention and demonstration during training, provide a promising alternative. Instead of generating an offline dataset and training novice policy against it [14], [15], we can incorporate a human subject into the loop of the training to provide online data. This can mitigate the distributional shift since the data generated with human-in-the-loop has closer state distribution to that of the novice policy, compared to the data collected completely from human demonstration [16]. These methods can significantly reduce the required amount of data and ensure safety during training by allowing humans

<sup>1</sup> Department of Computer Science, University of California, Los Angeles, {pzhang, zhizheng, bolei}@cs.ucla.edu

to provide corrective demonstrations in real time, guiding the robot toward safe and preferable behaviors. For safety-critical scenarios which the mobile robots are commonly in, safety is undoubtedly the first priority. Therefore, instead of collecting human feedback after robot’s interactions with the environment and then learning from these preference feedback [17], [18], [19], [20], [21], [22], [23], we opt for **online human-in-the-loop learning**, where human subjects actively intervene and provide demonstrations during training [24], [25], [26], [27], [28]. With online correction and demonstration from human subjects, AI alignment and training-time safety of the system can be substantially enhanced. However, most of the previous works focus on the simulation environment [27], [28], [29]. Whether human-in-the-loop method is applicable to real world and can overcome the challenges such as noisy sensory reading, teleoperation delay, and the complexity of visual appearance and robot kinematics is not yet examined.

In this work, we develop a data-efficient policy learning method to harness online human interventions and demonstrations. It can be generalized across different robots, tasks, and input representations in the real world. As shown in Fig. 1, we showcase that we are able to train two different mobile robots in two challenging tasks with high data efficiency. In the safe navigation task, the robot need to move forward while avoiding collision. The human subject might jump out suddenly so the robot needs to learn emergency stop. In the human following task, the robot receives the bounding box input of a tracked person and must follow the human while keeping away in a certain distance and stop if the human stops. The robot need to be responsive and agile, otherwise the tracked person might move out of the view. Those two tasks are well fitted to the online human-in-the-loop learning, as both tasks are human conductible, meaning humans can give control signals with the user interface. Meanwhile both tasks suffer from the huge sim-to-real gap as it is hard to simulate noisy camera images as well as complex human behaviors. **Our method is able to train robots solving these tasks** Behind the successful experiments is an online human-in-the-loop learning algorithm, which combines the RL and IL and learns from human interventional data and online demonstration. Even though we adopt reward-free setting, the TD learning of RL helps the model generalizes learned human preference across different state space [30]. The IL component constrains the policy to be closed to the data distribution seen before and regularizes the policy [31]. We summarize our main contributions as follows: 1) We propose a data-efficient human-in-the-loop learning method. Our method is *reward-free* and can be generalized across various task settings, sensory data and robotic embodiments. 2) We build real-world experiment systems with two robots and deploy the proposed method directly on the systems. The experiments show that the proposed method enables superior performance and high learning efficiency in two important mobile robot tasks: safe navigation and human following .

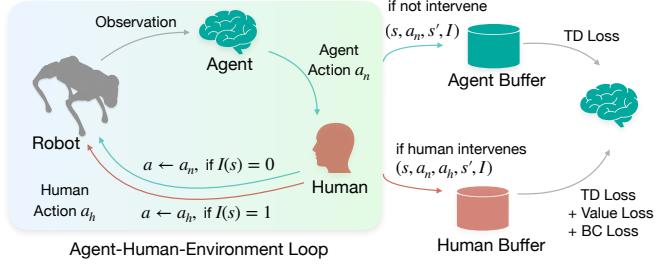
## II. RELATED WORK

### A. Human-in-the-loop policy learning

Online human-in-the-loop learning allows human subjects to proactively involve the agent-environment interactions based on their own judgment to ensure safety. Human subjects can terminate the episode if a near-accidental situation happens and such intervention policy can be learned [32], [33], [34], [35], [36], [37]. Recent studies explore learning from both intervention and demonstration in the human-agent shared autonomy [38], [39], [24], [25], [27], [40], [36]. However, previous methods do not fully utilize the power of human involvement. Interactive imitation learning method (HG-Dagger) [24] does not leverage data collected by agents, while Intervention Weighted Regression (IWR) [26] does not suppress undesired actions likely intervened by humans. Meanwhile, Expert Intervention Learning (EIL) [25] and IWR [26] focus on imitating actions step-wise without considering the temporal correlation between steps. Proxy Value Propagation [28] learns a proxy value function from the human intervention and demonstration without explicitly regularizing the policy to be closed to human behavior. These drawbacks harm learning efficiency and thus incur more human involvement. Most of the previous works focus on the simulation environment [26], [28], [29], and only a few works have directly applied human-in-the-loop online learning to the real-world robots due to practical challenges such as noisy observation, teleoperation delay, complex visual appearance and robot kinematics. In EIL [25], the learning method has been applied to a simple task: drive a racing robot on a rectangular track without collisions. In HG-Dagger [24], the task is to drive a vehicle on a straight road. In these works, the observation is a low-dimensional state vector of the location and velocity of the robot. In contrast, in the safe navigation task we experiment, a CNN network is trained from scratch with raw RGBD images. Also, our setting is more complex than the previous works, which includes different tasks, embodiments, and input formats.

### B. Learning for Mobile Robot

Mobile robots are designed to navigate and interact autonomously within their environments, necessitating continuous perception, localization, and navigation capabilities. Traditional methods rely on rule-based localization techniques, such as SIFT feature descriptors for feature matching [41] and Kalman filters for state estimation [42], coupled with classical navigation algorithms like A\* [43] and RRT\* [44]. While effective in structured settings, these approaches often struggle in dynamic or unstructured environments. To overcome these limitations, learning-based methods have been introduced in mobile robotics. One line of work adopts a modular design, training separate models for each sub-task — for instance, using object detectors for perception [45], learning-based SLAM for localization [46], and reinforcement learning (RL) controllers for navigation [47]. This approach allows for specialized optimization of each component but may suffer from integration challenges and cumula-



**Fig. 2: Method Overview.** In human-in-the-loop learning, the human subject supervises the action of the learning agent  $a_h$  and decides whether they should intervene and use the human action  $a_h$  instead. Depending on the intervention results, we store the transitions in two separate replay buffers. Data from both buffers are sampled to update the policy and value networks in real time, alongside the ongoing interactions between the agent, human, and environment.

tive errors across modules. Another line of work embraces an end-to-end paradigm that directly maps sensor observations to control actions [48], [49]. End-to-end learning removes the need for handcrafted intermediate representations and potentially captures complex correlations in the data. However, it often requires large amounts of training data and may lack interpretability for error diagnosis. Our experiments cover both approaches. In the human following task, we employ a modular setup where an object detector and tracker provide bounding box inputs of the target person, similar to methods used in [50], [51]. Conversely, in the safe navigation task, we adopt an end-to-end learning strategy by training a policy that takes raw RGBD images as input, inspired by works like [52], [47]. This dual approach demonstrates the versatility of our human-in-the-loop learning method across different robotic embodiments and sensory inputs.

### III. METHOD

In this section, we introduce an efficient online human-in-the-loop learning algorithm that trains agents from human interventions and demonstrations.

#### A. Problem Formulation

Our goal is to train a policy to solve the sequential decision-making problem. This is usually modeled by a Markov decision process (MDP). MDP is defined by the tuple  $M = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma, d_0 \rangle$  consisting of a state space  $\mathcal{S}$ , an action space  $\mathcal{A}$ , a state transition function  $\mathcal{P} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ , a reward function  $r : \mathcal{S} \times \mathcal{A} \rightarrow [R_{\min}, R_{\max}]$ , a discount factor  $\gamma \in (0, 1)$ , and an initial state distribution  $d_0 : \mathcal{S} \rightarrow [0, 1]$ . In traditional reinforcement learning, the goal is to learn a *novice policy*  $\pi_n(a|s) : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$  that can maximize the expected cumulative return:  $\pi_n = \arg \max_{\pi_n} \mathbb{E}_{\tau \sim P_{\pi_n}} [\sum_{t=0}^T \gamma^t r(s_t, a_t)]$ , wherein  $\tau = (s_0, a_0, \dots, s_T, a_T)$  is the trajectory sampled from trajectory distribution  $P_{\pi_n}$  induced by  $\pi_n$ ,  $d_0$  and  $\mathcal{P}$ . We denote the state-action value and state value of a policy  $\pi$  as  $Q(s, a) = \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)]$  and  $V(s) = \mathbb{E}_{a \sim \pi(\cdot|s)} Q(s, a)$ , respectively. Note that we will reuse the concept of values in the following, but we are in the reward-free setting, and the

environmental reward is not accessible. The goal of this paper is to learn human-compatible policy that minimize human intervention and can best capture human preferences.

Our method approaches policy learning with human-in-the-loop online intervention. Assuming the human expert has a *human policy*  $\pi_h(a_h|s) : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$ , which outputs human action  $a_h \in \mathcal{A}$ , sharing the same action space as novice action. We can model human intervention by an *intervention policy*  $I(s)$  to describe human subjects' interventional behaviors. The intervention is boolean  $I \sim I(\cdot|s), I \in \{0, 1\}$ . As shown in Fig. 2, during training, a human subject accompanies the novice policy and can intervene the agent by taking over the control to demonstrate desired behaviors. The behavior policy and the behavior action  $a$  can be written as:

$$\pi_b(\cdot|s) = (1 - I(s))\pi_n(\cdot|s) + I(s)\pi_h(\cdot|s), a \sim \pi_b(\cdot|s). \quad (1)$$

As we directly learn decision-making policy with real-world robots, there is no reward function defined. The only data we can collect from agent-human-environment interaction is the transitions  $(s, a_n, a_h, I, s')$  where  $s'$  is the next state and  $a_h$  might be null if human does not intervene. Thus, the key challenge to be addressed by this work is how can we effectively utilize these transitions to recover human preferences and learn high-performing policy.

#### B. Human-in-the-loop Training

As suggested by [12], the ultimate source of information about human preferences are human behaviors. Our first observation is that we can assume human actions to be optimal and the agent should reproduce those actions when in the same states. A straightforward idea is to apply Behavior Cloning (BC) loss on human data by minimizing the negative log-likelihood loss.

$$J^{BC}(\phi) = - \mathbb{E}_{(s, a_n, a_h)} I(s) \log \pi_n^\phi(a_h|s), \quad (2)$$

wherein the novice policy  $\pi_n$  is parameterized by  $\phi$ . This resembles the objective of HG-Dagger [24] and the intervention loss in EIL [25].

The second observation is that human intervention implies that the agent's actions being interrupted is undesirable and needs to be avoided. Previous works like IWR [26] and HG-Dagger do not suppress undesired actions intervened by human. EIL [25] directly modifies the Q values of those actions so that the policy (an argmax policy on Q values) will avoid those actions. As we want to work with mobile robots whose action spaces typically have velocity commands and are continuous, we can incorporate the value iteration in RL literature and first update the Q value, then update the policy to maximize the Q value. Here, we introduce a value network  $Q^\theta(s, a) \rightarrow \mathbb{R}$  parameterized by  $\theta$ . We can let the Q network predict a negative value for those novice actions that are intervened by human subject:

$$J^Q(\theta) = \mathbb{E}_{(s, a_n, a_h)} |Q^\theta(s, a_n) + B|^2 I(s), \quad (3)$$

wherein  $B > 0$  is a constant.  $J^Q(\theta)$  will be minimized.

**Algorithm 1:** Online Human-in-the-loop Learning

---

**Input:** Environment; Minibatch size  $N$ ; Human subject modeled by  $\pi_h$  and intervention policy  $I$ .  
**Result:** Learned novice policy  $\pi_n^\phi$ .

---

```

1 Initialize replay buffer  $\mathcal{D}_h, \mathcal{D}_n$ .
2 for every 0.2s do
3   Receive observation  $s$  from the environment.
4   Query novice action  $a_n \leftarrow \pi_n^\phi(\cdot|s)$ .
5   Query human intervention  $I(s)$ .
6   if  $I(s) = 0$  then
7      $a_h \leftarrow \emptyset, a \leftarrow a_n$ .
8   else
9      $a_h \leftarrow \pi_h(\cdot|s), a \leftarrow a_h$ .
10    Send  $a$  to the robot and get the next  $s'$ .
11    Store  $(s, a_n, a_h, I, s')$  to  $\mathcal{D}_h$  (if intervenes) or  $\mathcal{D}_n$ .
12    Sample  $N$  transitions from  $\mathcal{D}_n$  and  $\mathcal{D}_h$  separately and
        update  $\theta, \phi$  by minimizing Eq. (5) and Eq. (6).

```

---

The third observation is that those agent actions in which humans do not intervene can be considered human-compatible actions. EIL proposes to conduct IL on those actions. However, a carefully tuned hyper-parameter is needed to determine whether the agent's actions will cause later human intervention. Meanwhile, the step-wise IL on those unreliable agent actions might cause compounding errors. Our insight is that we can utilize this part of the data from the RL perspective. RL policies can establish causal relationships between observations, actions, and outcomes as they are trained in closed-loop. This makes them less vulnerable to covariate shifts and spurious correlations commonly seen in open loop IL [30]. Also, those transitions contain information on the forward dynamics of the environment [53], [54]. We propose to conduct TD learning instead of step-wise imitation learning. We can apply TD learning on all data (reward is removed as our method is in reward-free setting):

$$J^{\text{TD}}(\theta) = \mathbb{E}_{(s, a, s')} |Q^\theta(s, a) - \gamma \max_{a'} Q^{\hat{\theta}}(s', a')|^2, \quad (4)$$

where  $Q^{\hat{\theta}}$  is the delay-updated target network. By incorporating the RL framework, the objective to update the value network becomes:

$$J^{\text{Ours}}(\theta) = J^Q(\theta) + J^{\text{TD}}(\theta). \quad (5)$$

The objective of updating the policy network becomes:

$$J^{\text{Ours}}(\phi) = -\mathbb{E}_{s, a \sim \pi_n^\phi(\cdot|s)} Q^\theta(s, a) + J^{\text{BC}}(\phi). \quad (6)$$

Note that as suggested by [31], [30], adding behavior cloning loss into the RL objective can balance between exploiting the data in the dataset and regularize the policy towards actions seen in the dataset. This benefits human-in-the-loop learning as it makes policy closer to the human policy.

**Implementation Details.** With human in the loop, the transitions can be categorized into two sets: human-involved transitions and not-involved transitions. We create two replay buffers to store those transitions separately. Denote  $\mathcal{D}_h$  as the human buffer, which stores transitions  $(s, a_n, a_h, I, s')$  when human intervenes  $I(s) = 1$ . Another buffer  $\mathcal{D}_n$ , named

agent buffer, stores the transitions  $(s, a_n, I, s'), I(s) = 0$ . We will sample same amount of data from these two buffer to avoid data imbalance in training. Algorithm 1 describing the procedure of our method: we conduct online human-in-the-loop learning and update the policy and value network at each environmental interaction to minimize Eq. (5) and Eq. (6). We implement our method in Stable-Baselines3 [55] and set the minibatch size to  $N=1024$ , learning rate  $1e-4$ . The value network and policy network do not share parameters. There is a task-specific feature extractor that outputs a 256-dim feature vector, followed by a two-layer MLP with 256 hidden size. We set  $B = 1$ . The target Q network is delayed updated by coefficient  $\tau = 0.05$ . See Sec. V for details on real-world robot experiments.

#### IV. SIMULATION EXPERIMENTS

We conduct simulation experiments in the MetaDrive Safety Environment [56] following previous works [27], [28], [29]. The task for the agent is to steer the target vehicle with acceleration, brake, and steering, to reach the predefined destination. The ratio of episodes where the agent successfully reaches the destination is called the *success rate*. To increase the task's difficulty, we scatter obstacles randomly in each driving scene such as movable traffic vehicles, fixed traffic cones, and warning triangles. Crash to the obstacles will terminate the episode and the ratio of these episodes is called *crash rate*. Similarly, *out rate* is defined by the ratio of episodes where the agent drives out of the road. We evaluate the generalization performance of the learned policy by testing the trained agents in separate test environments. We split the driving scenes into the training set and test set with 50 different scenes in each set. The observation is a vector containing current states, the navigation information as well as the surrounding information. We use two-layer MLPs with 256 hidden nodes as the policy and the critic networks. A PPO [57] expert is trained with 20 million environmental interactions. The expert achieves  $\sim 0.80$  success rate and  $\sim 0.10$  crash rate. We follow previous works and define a threshold function to automatically issue intervention and provide expert demonstration [58], [59]. We will let the PPO expert take over if the probability of the agent's action under PPO's action distribution is lower than a threshold  $I(s) = \mathbf{1}_{\pi_{\text{PPO}}(a|s) < \epsilon}, a \sim \pi_n(\cdot|s)$ . We set  $\epsilon = 0.05$ . Per each 200 environment interactions, we roll out the learning agent without the PPO expert in the test environments for 100 episodes to get the evaluation results. All experiments are repeated 8 times with different random seeds. We compare with human-in-the-loop baselines Human-AI Copilot Optimization (HACO) [27] and Expert Intervention Learning (EIL) [25]. For RL baselines, we compared to PPO [57] and TD3 [60]. We also compare Behavior Cloning. For BC, instead of precollecting a dataset, we let the expert run in the environment and keep expanding the dataset. That is, we only update the policy with  $J^{\text{BC}}(\phi)$  where demonstrations are all from the expert. As shown in Fig. 3, by training on such insufficient 40k environment interactions, RL methods TD3 and PPO hardly learn any useful behavior. The evaluation

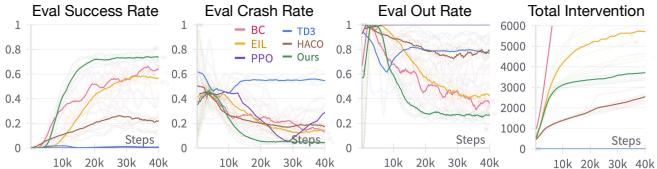


Fig. 3: **Experiment on Simulation Environment.** Human-in-the-loop methods achieves much better sample efficiency compared to the RL baselines.

results of the BC-trained agents is inferior as the behavior policy has gap to the learning policy [61]. On the contrary, human-in-the-loop methods all progress in solving the task. Our method outperforms EIL and HACO in term of task completion while we use less expert intervention.

## V. REAL-WORLD EXPERIMENTS

### A. Setup

We conduct experiments on two practical tasks for mobile robots: Safe Navigation and Human Following. An overview of the tasks is shown in Fig. 4. These tasks feature different settings and input representations to show the generalization ability of our method on real-world applications.

**Safe Navigation.** In this task, the robot navigates in a corridor environment by moving forward along the corridor. The size of the corridor is about 20m in length and 2.5m in width. The environment also has static obstacles and moving humans, and the robot needs to avoid collisions while reaching the goal. The robot is not given any high-level observation besides the RGB and depth images as the input. This is a highly challenging setting, as the robot needs to learn an *end-to-end* visual-motor policy that can safely traverse the scene with only a few human demonstrations without any external reward and prior knowledge. During training, the environment is randomly initialized with the obstacle positions and moving human trajectories. Some trajectories would directly intersect with the robot’s moving path, so the robot also needs to perform an emergency stop if necessary. We evaluate the success rate of four different sub-tasks the robot should learn in order to successfully traverse the corridor: 1) *Moving Straight*: the robot starts from one side of the corridor, and the goal is to reach the other side, where the surrounding is obstacle-free. 2) *Static Obstacle Avoidance*: We place static obstacles at an interval of 5m along the path, and the robot succeeds each time it passes an obstacle without collision. 3) *Dynamic Obstacle Avoidance*: A person walks around the robot at an interval of 5m along the path, and the robot succeeds each time it moves forward without colliding with the person. 4) *Emergency Stop*: A person who was not in the robot’s view suddenly appears in front of the robot and then leaves after 2s. The robot succeeds if it stops immediately when it sees the person and resumes moving after the person leaves. For all sub-tasks, the robot fails if it collides with anything or gets stuck in a position where it should keep moving. We perform 20 trials for each type of sub-task and evaluate the average success rate.

**Human Following.** In this task, the robot aims to follow a person within a certain distance ( $\sim 4\text{m}$ ) in an empty parking

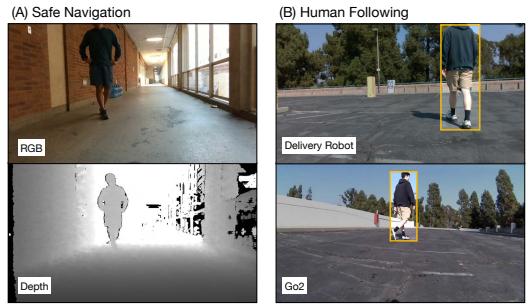


Fig. 4: **Overview of the tasks.** (A) In Safe Navigation, the robot is given the raw RGB and depth images as the input. The goal is to traverse the corridor environment while avoiding dynamic and moving obstacles, like a walking person, and perform emergency stops if necessary. (B) In Human Following, the robot is given the 2D bounding box of the tracked human. It needs to follow the human within a certain distance under different human actions and keep the target within the view. We experiment with two embodiments, which have different robot dynamics and camera parameters.

lot. It is given the 2D bounding box of the tracked person, provided by an off-the-shelf detector [62] and tracker [63], which run on the RGB images captured by the onboard camera. During training, the target person demonstrates random trajectories including sudden stops, and we test the trained policy on four specific types of human actions during evaluation: 1) *Going Straight*: the person walks in a straight line in the same direction as the robot for 10 meters, then stops. 2) *Following a Curve*: the person moves forward while turning for 10 meters, ending up facing a direction that is perpendicular to where they started. We separately evaluate when the curve is turning left or right. 3) *Sharp Turn*: the person starts walking in a direction that is perpendicular to the robot for 10 meters. We note that this action is never demonstrated during training. 4) *Sudden Stop*: the person suddenly stops while walking. The robot fails the task if it loses track of the person or is too close ( $< 1\text{m}$ ) or too far ( $> 5\text{m}$ ) from the person. We perform 20 trials for each type of human action and evaluate the average success rate.

**Robots.** We deploy our method in two different embodiments: the Unitree Go2 robot and the four-wheel delivery robot from Coco Robotics, as shown in Fig. 1. We use the native RGB camera from the Unitree Go2 to perform human tracking. For the delivery robot, we mount a Realsense D435i depth camera to the front of its chassis for both the task of human following and safe navigation. All cameras are streamed with a resolution of  $1280 \times 720$  in 30 fps.

**Baseline.** Our primary baseline is behavioral cloning (BC). For each task, we train the policy network with the same amount of human demonstration data as the human-in-the-loop method for 5,000 epochs using the MSE loss.

**Implementation Details.** For both tasks and robots, the action space is 2-dimensional with linear and angular velocities, and we use the robots’ low-level controllers to convert the velocity commands to motor torques. As for the observation space in the safe navigation task, the images

TABLE I: Success rate of the Safe Navigation task.

	Moving Straight	Static Obstacle Avoidance	Dynamic Obstacle Avoidance	Emergency Stop
BC	0.25	0.50	0.45	0.60
Ours	1.00	0.85	0.90	1.00

TABLE II: Success rate of the Human Following task.

(a) Evaluation on the CoCo delivery Robot.

	Straight	Left Curve	Right Curve	Sudden Stop
BC	1.00	0.60	0.60	0.45
Ours	1.00	0.80	1.00	0.85

(b) Evaluation on the Unitree Go2 robot.

	Straight	L. Curve	R. Curve	Sharp Turn	Sudden Stop
BC	0.80	0.60	1.00	0.00	0.80
Ours	1.00	1.00	1.00	1.00	1.00

are first resized to  $84 \times 84$  resolution. We stack the normalized RGB and depth images along the channel dimension and concatenate the images of the past five frames. The feature extractor is a 5-layer convolutional neural network with channels [16, 32, 64, 128]. The observation space of the human following task is 5-dimensional, including the position and dimension of the normalized 2D bounding box of the tracked person and the validity. We then concatenate the observations of the past five frames into a single vector. We use two-layer MLPs with 256 hidden nodes as the policy and value networks. By stacking past 5 steps' observations, the agent is aware of the history information in past 1s, as the whole system operates at 5 Hz. We implement a human intervention interface on a laptop with an Nvidia RTX 4080 mobile GPU to support online training and human control with the joystick. In both tasks, we perform 2,000 steps of environment interactions, which is less than 10 minutes in wall time. All experiments with human participants have obtained approval through our institute's IRB.

## B. Results

**Safe Navigation.** We provide quantitative results of the safe navigation task in Table I, in which the model needs to learn an end-to-end policy with the raw RGBD images as the input. It is clear that BC struggles with most sub-tasks and only achieves a 25% success rate on moving straight. Our method outperforms BC by a clear margin and performs well even on the most difficult dynamic obstacle avoidance, even the agent is trained in less than 10 minutes. We observe in the experiments that the BC policy is prone to being stuck in unfamiliar situations, such as facing toward the wall. Furthermore, the BC policy would also perform emergency stops at locations where it shouldn't stop. We speculate that with an end-to-end policy, BC is more likely to overfit the raw pixel values that are irrelevant to the task and can easily be changed by the lightning. Our method suffers less from overfitting and achieves better safety and efficiency as the human operator can intervene and correct wrong actions during training if any out-of-distribution state is encountered.

**Human Following.** Table II shows results for the human following task on different embodiments. Our method performs



Fig. 5: Qualitative results for the Safe Navigation and Human Following tasks. **(A)** The stacked trajectories for static obstacle avoidance. **(B)** The stacked trajectories for human following during sharp turns.

better than BC under all human actions with two different robots. Notably, our method achieves 100% success rate on the Unitree Go2 robot, where BC fails completely when the human is doing a sharp turn, an action never demonstrated during training. As the human bounding box is not likely to be lost in the demonstration data, the BC agent is incapable to recover from the out-of-distribution states of lost tracks during the sharp turn. Our method can still infer human direction from the history information by learning from the intervened human demonstrations of these challenging states. **Qualitative Results** As shown in Fig. 5. In the Safe Navigation task, our method successfully avoids a static human on its path while BC collides with the human without alternating its path in the Safe Navigation task. In the Human Following task, we can notice that BC loses its track and turns in the opposite direction while our method keeps following the person. Please see the supplementary video for more results.

## VI. CONCLUSION

We present a data-efficient human-in-the-loop learning method that combines imitation learning and reinforcement learning to address challenging tasks on mobile robots. This work showcases the successful application of online human-in-the-loop learning methods on real-world robotic tasks. We can train mobile robots to conduct safe navigation and human following tasks in no more than 10 minutes. Compared to the IL baselines, our method significantly improves performance and safety in challenging scenarios, including sharp turn and sudden stop. In the simulation experiment, we show that our method can achieve near-optimal performance with fewer human interventions. While this work mainly focuses on proving the applicability of the online human-in-the-loop method on real-world robots, a natural extension is to incorporate even more challenging tasks such as long-horizon goal-conditioned navigation.

## REFERENCES

- [1] V. Engesser, E. Rombaut, L. Vanhaverbeke, and P. Lebeau, “Autonomous delivery solutions for last-mile logistics operations: A literature review and research agenda,” *Sustainability*, vol. 15, no. 3, p. 2774, 2023.
- [2] A. A. Morgan, J. Abdi, M. A. Syed, G. E. Kohen, P. Barlow, and M. P. Vizcaychipi, “Robots in healthcare: a scoping review,” *Current robotics reports*, vol. 3, no. 4, pp. 271–280, 2022.
- [3] A. Tuomi, I. P. Tussyadiah, and J. Stienmetz, “Applications and implications of service robots in hospitality,” *Cornell Hospitality Quarterly*, vol. 62, no. 2, pp. 232–247, 2021.
- [4] D. Fox, W. Burgard, and S. Thrun, “The dynamic window approach to collision avoidance,” *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.
- [5] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim, “Dynamic locomotion in the mit cheetah 3 through convex model-predictive control,” in *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2018, pp. 1–9.
- [6] X. Pan, T. Zhang, B. Ichter, A. Faust, J. Tan, and S. Ha, “Zero-shot imitation learning from demonstrations for legged robot visual navigation,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 679–685.
- [7] D. Hoeller, L. Wellhausen, F. Farshidian, and M. Hutter, “Learning a state representation and navigation in cluttered and dynamic environments,” *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5081–5088, 2021.
- [8] M. Seo, R. Gupta, Y. Zhu, A. Skoutnev, L. Sentis, and Y. Zhu, “Learning to walk by steering: Perceptive quadrupedal locomotion in dynamic environments,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5099–5105.
- [9] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, et al., “Rt-2: Vision-language-action models transfer web knowledge to robotic control,” *arXiv preprint arXiv:2307.15818*, 2023.
- [10] A. Sridhar, D. Shah, C. Glossop, and S. Levine, “Nomad: Goal masked diffusion policies for navigation and exploration,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 63–70.
- [11] J. Leike, D. Krueger, T. Everitt, M. Martic, V. Maini, and S. Legg, “Scalable agent alignment via reward modeling: a research direction,” *arXiv preprint arXiv:1811.07871*, 2018.
- [12] S. Russell, *Human compatible: Artificial intelligence and the problem of control*. Penguin, 2019.
- [13] A. Pan, K. Bhatia, and J. Steinhardt, “The effects of reward misspecification: Mapping and mitigating misaligned models,” *International Conference on Learning Representations*, 2022.
- [14] J. Ho and S. Ermon, “Generative adversarial imitation learning,” in *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, Eds., 2016, pp. 4565–4573. [Online]. Available: <https://proceedings.neurips.cc/paper/2016/hash/cc7e2b878868cbae992d1fb743995d8f-Abstract.html>
- [15] J. Fu, K. Luo, and S. Levine, “Learning robust rewards with adversarial inverse reinforcement learning,” in *International Conference on Learning Representations*, 2018.
- [16] S. Ross and D. Bagnell, “Efficient reductions for imitation learning,” in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 661–668.
- [17] C. Wirth, R. Akrour, G. Neumann, J. Fürnkranz, et al., “A survey of preference-based reinforcement learning methods,” *Journal of Machine Learning Research*, vol. 18, no. 136, pp. 1–46, 2017.
- [18] P. F. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei, “Deep reinforcement learning from human preferences,” in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 4299–4307. [Online]. Available: <https://proceedings.neurips.cc/paper/2017/hash/d5e2c0adad503c91f91df240d0cd4e49-Abstract.html>
- [19] S. Reddy, A. D. Dragan, and S. Levine, “Shared autonomy via deep reinforcement learning,” *Robotics: Science and Systems*, 2018.
- [20] G. Warnell, N. R. Waytowich, V. Lawhern, and P. Stone, “Deep TAMER: interactive agent shaping in high-dimensional state spaces,” in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, S. A. McIlraith and K. Q. Weinberger, Eds. AAAI Press, 2018, pp. 1545–1554. [Online]. Available: <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16200>
- [21] M. Palan, G. Shevchuk, N. Charles Landolfi, and D. Sadigh, “Learning reward functions by integrating human demonstrations and preferences,” in *Robotics: Science and Systems*, 2019.
- [22] L. Guan, M. Verma, S. Guo, R. Zhang, and S. Kambhampati, “Widening the pipeline in human-guided reinforcement learning with explanation and context-aware data augmentation,” *Advances in Neural Information Processing Systems*, vol. 34, 2021.
- [23] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al., “Training language models to follow instructions with human feedback,” *arXiv preprint arXiv:2203.02155*, 2022.
- [24] M. Kelly, C. Sidrane, K. Driggs-Campbell, and M. J. Kochenderfer, “Hg-dagger: Interactive imitation learning with human experts,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8077–8083.
- [25] J. Spencer, S. Choudhury, M. Barnes, M. Schmittle, M. Chiang, P. Ramadge, and S. Srinivasa, “Learning from interventions,” in *Robotics: Science and Systems (RSS)*, 2020.
- [26] A. Mandlekar, D. Xu, R. Martín-Martín, Y. Zhu, L. Fei-Fei, and S. Savarese, “Human-in-the-loop imitation learning using remote teleoperation,” *ArXiv preprint*, vol. abs/2012.06733, 2020. [Online]. Available: <https://arxiv.org/abs/2012.06733>
- [27] Q. Li, Z. Peng, and B. Zhou, “Efficient learning of safe driving policy via human-ai copilot optimization,” in *International Conference on Learning Representations*, 2022.
- [28] Z. Peng, W. Mo, C. Duan, Q. Li, and B. Zhou, “Learning from active human involvement through proxy value propagation,” in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.
- [29] Z. Huang, Z. Sheng, C. Ma, and S. Chen, “Human as ai mentor: Enhanced human-in-the-loop reinforcement learning for safe and efficient autonomous driving,” *Communications in Transportation Research*, vol. 4, p. 100127, 2024.
- [30] Y. Lu, J. Fu, G. Tucker, X. Pan, E. Bronstein, R. Roelofs, B. Sapp, B. White, A. Faust, S. Whiteson, et al., “Imitation is not enough: Robustifying imitation with reinforcement learning for challenging driving scenarios,” in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 7553–7560.
- [31] S. Fujimoto and S. S. Gu, “A minimalist approach to offline reinforcement learning,” *Advances in neural information processing systems*, vol. 34, pp. 20 132–20 145, 2021.
- [32] J. Zhang and K. Cho, “Query-efficient imitation learning for end-to-end simulated driving,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, 2017.
- [33] D. Abel, J. Salvatier, A. Stuhlmüller, and O. Evans, “Agent-agnostic human-in-the-loop reinforcement learning,” *ArXiv preprint*, vol. abs/1701.04079, 2017. [Online]. Available: <https://arxiv.org/abs/1701.04079>
- [34] W. Saunders, G. Sastry, A. Stuhlmüller, and O. Evans, “Trial without error: Towards safe reinforcement learning via human intervention,” in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2018, pp. 2067–2069.
- [35] E. Pakdamanian, S. Sheng, S. Baee, S. Heo, S. Kraus, and L. Feng, “Deeptake: Prediction of driver takeover behavior using multimodal data,” in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–14.
- [36] Y. Xu, Z. Liu, G. Duan, J. Zhu, X. Bai, and J. Tan, “Look before you leap: Safe model-based reinforcement learning with human intervention,” in *Conference on Robot Learning*. PMLR, 2022, pp. 332–341.
- [37] Z. Wang, X. Xiao, B. Liu, G. Warnell, and P. Stone, “Appli: Adaptive planner parameter learning from interventions,” in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 6079–6085.
- [38] J. MacGlashan, M. K. Ho, R. Loftin, B. Peng, G. Wang, D. L. Roberts, M. E. Taylor, and M. L. Littman, “Interactive learning from policy-

- dependent human feedback,” in *International conference on machine learning*. PMLR, 2017, pp. 2285–2294.
- [39] K. Menda, K. Driggs-Campbell, and M. J. Kochenderfer, “Ensembleddagger: A bayesian approach to safe imitation learning,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 5041–5048.
- [40] A. Jonnavittula and D. P. Losey, “Learning to share autonomy across repeated interaction,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 1851–1858.
- [41] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International journal of computer vision*, vol. 60, pp. 91–110, 2004.
- [42] R. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [43] P. E. Hart, N. J. Nilsson, and B. Raphael, “A formal basis for the heuristic determination of minimum cost paths,” *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.
- [44] S. Karaman and E. Frazzoli, “Sampling-based algorithms for optimal motion planning,” *The international journal of robotics research*, vol. 30, no. 7, pp. 846–894, 2011.
- [45] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.
- [46] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, “Orb-slam: a versatile and accurate monocular slam system,” *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.
- [47] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, “Target-driven visual navigation in indoor scenes using deep reinforcement learning,” in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 3357–3364.
- [48] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al., “End to end learning for self-driving cars,” *ArXiv preprint*, vol. abs/1604.07316, 2016. [Online]. Available: <https://arxiv.org/abs/1604.07316>
- [49] M. Pfeiffer, M. Schaeuble, J. Nieto, R. Siegwart, and C. Cadena, “From perception to decision: A data-driven approach to end-to-end motion planning for autonomous ground robots,” in *2017 ieee international conference on robotics and automation (icra)*. IEEE, 2017, pp. 1527–1533.
- [50] M. Müller, A. Dosovitskiy, B. Ghanem, and V. Koltun, “Driving policy transfer via modularity and abstraction,” *arXiv preprint arXiv:1804.09364*, 2018.
- [51] T. Gervet, S. Chintala, D. Batra, J. Malik, and D. S. Chaplot, “Navigating to objects in the real world,” *Science Robotics*, vol. 8, no. 79, p. eadfc6991, 2023.
- [52] M. Pfeiffer, S. Shukla, M. Turchetta, C. Cadena, A. Krause, R. Siegwart, and J. Nieto, “Reinforced imitation: Sample efficient deep reinforcement learning for mapless navigation by leveraging prior demonstrations,” *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4423–4430, 2018.
- [53] S. Levine, A. Kumar, G. Tucker, and J. Fu, “Offline reinforcement learning: Tutorial, review, and perspectives on open problems,” *ArXiv preprint*, vol. abs/2005.01643, 2020. [Online]. Available: <https://arxiv.org/abs/2005.01643>
- [54] T. Yu, A. Kumar, Y. Chebotar, K. Hausman, C. Finn, and S. Levine, “How to leverage unlabeled data in offline reinforcement learning,” *arXiv preprint arXiv:2202.01741*, 2022.
- [55] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, “Stable-baselines3: Reliable reinforcement learning implementations,” *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021. [Online]. Available: <http://jmlr.org/papers/v22/20-1364.html>
- [56] Q. Li, Z. Peng, L. Feng, Q. Zhang, Z. Xue, and B. Zhou, “Metadrive: Composing diverse driving scenarios for generalizable reinforcement learning,” *IEEE transactions on pattern analysis and machine intelligence*, 2022.
- [57] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” *arXiv preprint arXiv:1707.06347*, 2017.
- [58] Z. Peng, Q. Li, C. Liu, and B. Zhou, “Safe driving via expert guided policy optimization,” in *5th Annual Conference on Robot Learning*, 2021.
- [59] Z. Xue, Z. Peng, Q. Li, Z. Liu, and B. Zhou, “Guarded policy optimization with imperfect online demonstrations,” in *The Eleventh International Conference on Learning Representations*.
- [60] S. Fujimoto, H. van Hoof, and D. Meger, “Addressing function approximation error in actor-critic methods,” in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholm, Sweden, July 10–15, 2018*, ser. Proceedings of Machine Learning Research, J. G. Dy and A. Krause, Eds., vol. 80. PMLR, 2018, pp. 1582–1591. [Online]. Available: <http://proceedings.mlr.press/v80/fujimoto18a.html>
- [61] S. Ross, G. Gordon, and D. Bagnell, “A reduction of imitation learning and structured prediction to no-regret online learning,” in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 627–635.
- [62] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu, et al., “Grounding dino: Marrying dino with grounded pre-training for open-set object detection,” *arXiv preprint arXiv:2303.05499*, 2023.
- [63] G. Bhat, M. Danelljan, L. V. Gool, and R. Timofte, “Learning discriminative model prediction for tracking,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6182–6191.