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

Environmental and Behavioral Imitation for Autonomous Navigation

Junki Aoki[1,2], Fumihiro Sasaki[1], Kohei Matsumoto[3], Ryota Yamashina[1] and Ryo Kurazume[3]

***Abstract*— In this paper, we introduce a framework for imitation learning in navigation that enables policy learning from one-shot images without a physical robot and facilitates the transfer of this policy from simulation to reality. Utilizing Neural Radiance Fields (NeRF), our approach generates a simulated environment and simultaneously models expert behavior. This removes the necessity for a physical robot during both the expert teaching phase and the agent's learning process, allowing for the application of policies learned within the NeRF simulation to real-world robots. We validate our method by demonstrating the navigation with an actual robot using the policy learned by our approach. Moreover, we present a method for adapting to changes in the robot configuration, such as camera parameters and robot dimensions, by simulating adjustments in the robot configuration throughout the learning and assessing its generalizability.**

I. INTRODUCTION

Vision-based robot navigation presents a promising avenue, offering high-dimensional data, diversity in applicable areas, and the advantage of compact, cost-effective camera sensors over traditional LiDAR and GPS-based methods. Machine learning and reinforcement learning (RL)–based visual navigation garners interest for its capacity to enable navigation absent maps and explicit rules [1], [2]. Specifically, vision-based imitation learning (V-IL) has emerged as an efficient training strategy over RL-based approaches, enabling robots to learn policies through expert demonstrations of desired behavior effectively [3]–[5].

Nevertheless, the practical application of V-IL in robot navigation faces four challenges: **Problem (Prob.) 1**: Environmental damage may arise from collisions with obstacles or walls as robots gather training data in real settings. In the case of IL, human demonstrators are needed to operate a robot in a navigation environment, while learning with RL is even more so because the agent learns the policy by interacting with the environment. **Prob. 2**: Typical IL approaches require state and action pairs, necessitating human facilitation for bringing the robot to the environment and operation within the navigational environment for demonstration acquisition. **Prob. 3**: These methods impose significant human labor to both operate robots for data collection and physically bring robots to environments. **Prob. 4**: The effectiveness of a trained policy depends on the robot configuration, such as camera parameters and robot dimensions, with alterations in camera specifications or mounting positions necessitating the collection of new demonstrations.

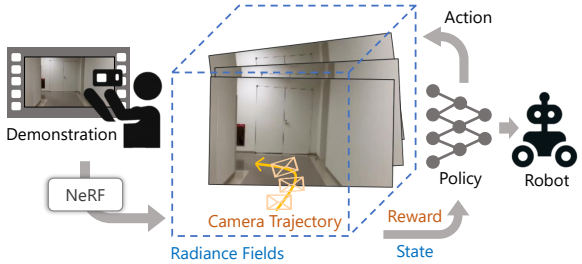

Fig. 1. The concept of the EBIAN. After humans capture images of the navigation course, radiance fields and camera trajectory are estimated by NeRF all at once. They are dealt with as an environment for observing a state and expected behavior for a reward and are leveraged by reinforcement learning agent.

This paper introduces an imitation learning framework named *Environmental and Behavioral Imitation for Autonomous Navigation (EBIAN)*, utilizing Neural Radiance Fields (NeRF). EBIAN features imitating an environment and expected behavior via NeRF, which optimize radiance fields for a photorealistic image rendering and an associated camera trajectory [6], [7], to address these issues by creating photorealistic simulation environments for V-IL training. The capability of NeRF to produce photorealistic images surpasses traditional 3D reconstruction techniques, such as structure from motion [8], potentially reducing the simulation-to-reality (sim2real) gap [9]. NeRF facilitates free-viewpoint rendering, allowing agents to simulate observations from any chosen pose, thus leveraging NeRF as the simulated environment could eliminate the need for real-world interaction (**Prob. 1**). Moreover, NeRF provides state-action pairs through rendered images and camera poses for navigation training (**Prob. 2**). This approach requires only a one-shot video sequence, captured using simple devices like smartphones, without direct robot operation, obviating the need to bring the actual robot to the environment, and facilitating navigation in environments the robot has not previously encountered (**Prob. 3**). Furthermore, the adaptability of NeRF to camera parameter variations ensures the flexibility of a policy for variation of robot configurations (**Prob. 4**). The concept of our method is shown in Fig. 1.

Our contributions are as follows:

- We propose a one-shot imitation learning framework for mobile robot navigation, by imitating an environment and expected behavior optimized by NeRF, is capable of sim2real transfer and accommodating diverse intrinsic

[1] Junki Aoki, Fumihiro Sasaki and Ryota Yamashina are with Ricoh Company, Ltd. {junki.aoki, fumihiro.fs.sasaki, ryohta.yamashina}@jp.ricoh.com
[2] Junki Aoki is with the Graduate School of Information Science and Electrical Engineering, Kyushu University.
[3] Ryo Kurazume and Kohei Matsumoto are with the Faculty of Information Science and Electrical Engineering, Kyushu University. kurazume@ait.kyushu-u.ac.jp, matsumoto@irvs.ait.kyushu-u.ac.jp

and extrinsic camera parameters.
- We assess the adaptability of our framework for changes in intrinsic and extrinsic camera parameters—critical aspects of robot configuration.
- We empirically demonstrate that the policy learned by our method successfully works for an actual mobile robot in the real environment.

The structure of this paper is outlined as follows: Section II reviews related work. Section III details the methodology behind our proposed framework. Section IV describes the experimental setup and findings. Section V discusses the limitations of our study. Section VI concludes this paper.

II. RELATED WORK

A. Imitation from Observation

Imitation from observation (IfO) is a methodology that bypasses the need to access an expert's action space, relying solely on optimizing action sequences from observed sequences [10]–[12]. IfO enables experts to teach behavior using a camera device not mounted on the robot, such as a smartphone, without operating the physical robot. However, these methods require the robot to interact with environments during the learning phase through RL, posing a risk of environmental damage when interactions occur in real settings. Additionally, utilizing simulation environments face the challenge of high preparation costs. Unlike these methods, our method eliminates the need for real-world interaction and allows humans to prepare simulation environments easily by capturing images with everyday devices.

B. Visual Teach and Repeat

Visual Teach and Repeat (VT&R) represents a conventional strategy for following expert trajectory in mobile robot navigation. One approach has been implemented using visual SLAM techniques like ORB-SLAM [13] and RTAB-MAP [14], offering cost-effective solutions [15]. However, visual SLAM-based approaches often fall short in feature-sparse environments, and their performance heavily relies on the robot configuration. Differences in camera specifications or mounting heights from those used during the SLAM mapping process can impede feature point correspondence extraction and navigation. One of the other approaches, a monocular image-based method without SLAM that incorporates misalignment correction [16], still needs accurate robot odometry and bringing the robot to the environment.

Efforts to facilitate VT&R using handheld devices for trajectory teaching have been researched [17], [18]. VOILA [17], for example, achieved autonomous navigation using different camera specifications and mounting heights by learning policies through the correspondence of feature points from image sequences captured by human demonstration. Despite these advancements, such methods still require the robot to interact with the real environment for policy learning, posing risks of environmental damage.

Our approach eliminates the need for real-world interaction, as the navigation policy is learned entirely within a NeRF environment. The capability of optimization of radiance fields and camera poses by NeRF simplifies demonstration acquisition and simulated environment creation only by recording a video along the intended route using everyday devices like smartphones. This circumvents the issues associated with environmental damage and the preparation costs to make simulation environments.

C. Sim2Real Transfer

Training within simulated environments offers a viable solution to mitigate environmental damage, obviate the need for physical robot transfer during the learning phase, and enhance adaptability to changes in robot configuration. The primary hurdle in leveraging simulations is bridging the domain gap between simulated and real-world environments, known as the sim2real transfer challenge. Traditional methods for this issue have involved domain transformation techniques [19]–[22], randomization [23], and adaptation methods [24]. These solutions typically presuppose the creation of 3D simulation environments through labor-intensive modeling processes, leading to significant preparation costs.

Recent advancements with tools like Luma AI [25] and nerfstudio [26] have simplified the generation of photorealistic simulation environments, substantially reducing setup costs. Our approach represents the first effort to take full advantage of the great ease of creating the NeRF environment and to imitate expert behavior by employing NeRF alongside an optimized camera trajectory. Unlike conventional sim2real methods that need image data from both simulated and real environments, our approach requires only a one-shot video sequence to be used as a demonstration.

D. Robot Navigation with Neural Radiance Fields

Recent initiatives have employed NeRF for vision-based robot navigation [9], [27], [28]. The work presented in [27] aligns with our aim of imitating expert behavior from a one-shot demonstration, applying NeRF to forecast movements and deformations of rigid objects in robot manipulation tasks. Our approach leverages NeRF to simulate both the environment and expert trajectories for visual navigation tasks. The study in [28] has similarities with our approach in the application of NeRF for visual navigation. However, it presupposes comprehensive environmental sensing via camera, differing from our framework, which integrates NeRF-generated environments and expert trajectories from video sequences for learning. Furthermore, it does not validate navigation performance on actual robots, a critical aspect of our research. Additionally, [9] explores the sim2real transfer potential for policies in bipedal robot tasks, focusing on the application of NeRF for behavior optimization. NeRF environment generation and the definition of expected tasks are separate in the [9], whereas our approach defines the environment generation and task definition in one step.

Our work features being the first to employ NeRF-generated radiance fields and an expert trajectory for navigation simulation, specifically evaluating the generalizability across variations in camera intrinsic and extrinsic parameters and demonstrating real-world navigation.

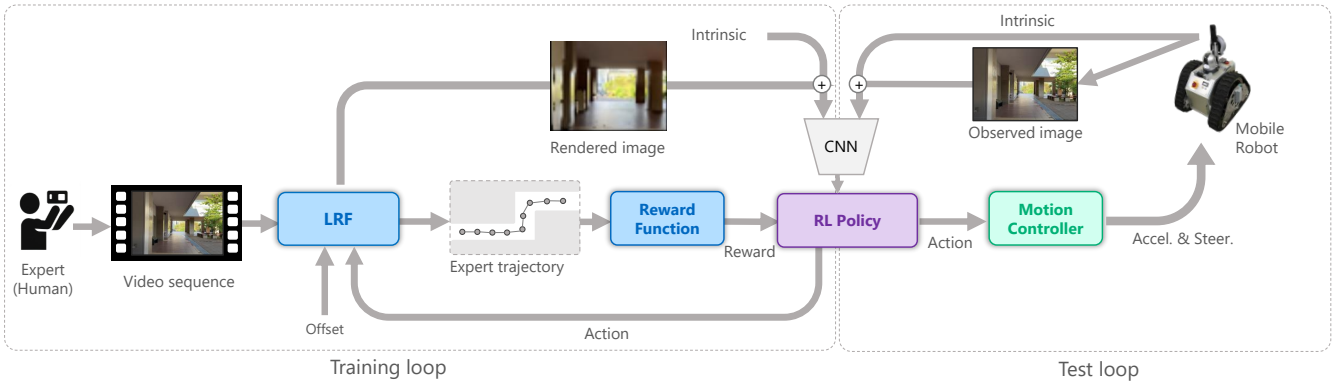

Fig. 2. Pipeline of the EBIAN. From a video captured by an expert, local radiance fields (LRF) generates radiance fields and camera trajectories. The RL agent optimizes its actions based on the images observed within the radiance fields. The RL agent can earn high rewards by generating actions that follow the camera trajectories estimated by LRF.

III. METHOD

A. Overview

This section outlines the pipeline of our proposed method, depicted in Fig. 2. As a 3D reconstruction and pose optimization method, we leverage local radiance fields (LRF) [7], which optimizes camera poses and radiance fields progressively and is robust for a front-facing video sequence. We propose the V-IL framework named *EBIAN*, using LRF.

In the training loop, humans record a video while navigating the intended path. This video is processed by LRF, which then commences the optimization of radiance fields and camera poses, the latter serving as the expert trajectory for navigation. An RL policy is trained using these radiance fields and the trajectory. To accommodate variations in camera mounting positions and specifications, we apply extrinsic and intrinsic offsets to state representations, using height and field of view (FoV) as examples, respectively.

During the testing loop, a mobile robot observes images through its camera. These images are fed into the RL policy as state inputs, based on which the policy generates an action, navigating the robot.

B. State and Action Representation

We define an observation $O_t \in \mathbb{R}^{H\times W\times 3}$ as an RGB image, formulated as follows:

$$O_t = \begin{cases} LRF(P_t, \omega_h), & \textit{if training} \\ I_t(\omega_h), & \textit{if test}, \end{cases} \tag{1}$$

where $LRF(P_t, \omega_h)$ denotes an image rendered by LRF, $P_t \in \mathbb{R}^6$ represents the camera pose within the radiance fields at a time step t, I_t is an image captured by a robot's camera, and ω_h is the horizontal FoV of the camera, as an intrinsic parameter.

State representations are derived through a convolutional neural network (CNN), expressed as:

$$S_t = CNN(O_t, C) \tag{2}$$

$C \in \mathbb{R}^{H\times W}$ is an intrinsic condition described in Section III-E. The action space is continuous, characterized by Euclidean Distance. An action $A_t \in \mathbb{R}^D$ is sampled from the policy $A_t \sim \pi(\cdot|S_t)$, with D dimensionally depends on $SE(2)$ or $SE(3)$ for 2D and 3D movements, respectively. Specifically, A_t is the Euclidean Distance between camera poses from timestep t to $t+1$. For ground vehicles, 2D movement (x, y) is applicable, whereas for quadcopters, 3D movement (x, y, z) is pertinent, with x indicating forward, y leftward, and z upward directions. Additionally, ϕ, θ and ψ denote pose in roll, pitch and yaw direction, respectively.

Given the scale discrepancy between real environments and radiance fields—attributable to monocular image-based pose estimation not aligning with real-world scales—we employ two manually selected keyframes (a, b), such as the endpoints of a corridor, to calculate the scale as follows:

$$sc = \frac{dist(P^{real}(I_a), P^{real}(I_b))}{dist(P_a^{LRF}, P_b^{LRF})}, \tag{3}$$

where $dist(a, b)$ denotes Euclidean Distance between a and b. During testing, actions are scaled accordingly:

$$A_t^{real} = sc \times A_t^{LRF}. \tag{4}$$

Here, I_a and I_b are visually chosen by humans, with $dist(P^{real}(I_a), P^{real}(I_b))$ measured by Euclidean Distance.

C. Expert Demonstration Extraction

Expert demonstrations $\mathcal{D} = \{(O_1^e, A_1^e), \ldots, (O_n^e, A_n^e)\}$ are derived from poses $P_k^e, k \in [1 \ldots N]$ optimized via LRF, where the superscript by e denotes data from the expert demonstration. For each pose P_t^e, LRF renders an image, which is captured as an observation $O_t^e = LRF(P_t^e, \omega_h)$. The expert action A_t^e is determined as $A_t^e = P_{t+1}^e - P_t^e$. The upper and lower bounds of the actions are determined by the upper and lower bounds of the expert actions.

Given that trajectories based on precise step-by-step poses may be overly sensitive to human gait-induced vibrations, resulting in an undesirably meandering path, we apply sampling to select the next expert pose after surpassing a set stride interval, aiming for a smoother trajectory conducive to autonomous navigation.

It is crucial to adapt the raw LRF-estimated poses to the type of mobile robot. For experiments with a non-holonomic ground vehicle, directly applying the yaw direction from

LRF poses may lead to suboptimal trajectories. For instance, using action dimensions (x, y, ψ) and training an RL policy on unmodified LRF trajectories could result in inefficient maneuvers, such as unnecessary turns. To address this, we adjust the yaw direction based on the (x, y) movement from the expert trajectory, calculating $\psi = atan(y, x)$. This correction is specific to non-holonomic robots and may not be necessary for holonomic ones.

During training, expert actions are applied to actions other than those to be optimized. For a non-holonomic robot at timestep t, the action $A_t = (a_t^x, a_t^y, a_t^{e,z}, a_t^{e,\phi}, a_t^{e,\theta}, atan(a_t^y, a_t^x))$ is added to the previous pose $P_{t-1} = (p_{t-1}^x, p_{t-1}^y, p_{t-1}^z, p_{t-1}^\phi, p_{t-1}^\theta, p_{t-1}^\psi)$ to obtain $P_t = P_{t-1} + A_t$, with $a_t^{e,z}, a_t^{e,\phi}, a_t^{e,\theta}$ derived from the expert trajectory. This also should be adjusted according to the type of mobile robot.

D. Policy Optimization

The reward function for each agent transition $R(O_t, A_t, O_{t+1})$ is designed to encourage imitation of the expert demonstration. The reward increases as the agent's pose P_{t+1} after taking action A_t closely aligns with the expected expert pose P_{t+1}^e.

To encourage smoothness in the agent's path and reduce abrupt directional changes and zig-zag trajectories, we incorporate a temporal smoothness regularization from Conditioning for Action Policy Smoothness (CAPS) [29], alongside a regularization based on expert actions A_{t-1}^e, A_t^e to maintain essential action variations, such as turning corners. The reward function is defined as:

$$R(O_t, A_t, O_{t+1}; \mathcal{D}) = \begin{cases} L_p - \lambda L_{caps} & \textit{if } L_p \geqq \gamma \\ 0 & \textit{otherwise}, \end{cases} \quad (5)$$

$$L_p = \frac{1}{1 + dist(P_t, P_t^e)}, \quad (6)$$

$$L_{caps} = dist(A_{t-1}, A_t) - dist(A_{t-1}^e, A_t^e), \quad (7)$$

where $dist(a, b)$ calculates the Euclidean Distance between positions a and b, and γ serves as a tolerance threshold for the permissible deviation from the expert pose within an episode, and the episode ends when $L_p < \gamma$ or the agent reaches the goal. $L_{caps} \geqq 0$ is the range of value.

In addition to the optimization of Q-value $Q(S, A)$ using the above reward function, we implement BC-SAC [30] to effectively learn a navigation policy that closely fits expert demonstrations and can adaptively return to the expert trajectory from states outside the training distribution. BC-SAC combines the soft-actor critic objective with behavioral cloning (BC) loss for the actor's objective function, expressed as follows:

$$\mathbb{E}_{S,A\sim\pi}[Q(S, A) + \mathcal{H}(\pi(\cdot|S))] + \lambda \mathbb{E}_{S,A\sim\mathcal{D}}[\log \pi(A|S)]. \quad (8)$$

where $\mathcal{H}$ is entropy.

E. Camera Configurations

EBIAN enhances policy flexibility to apply for changes in the robot's camera configurations without necessitating retraining. It is crucial to simulate variations in intrinsic and extrinsic parameters during the training phase with offsets.

Extrinsic: NeRF can simulate the free-viewpoint rendering and the policy can observe rendering results that take into account differences in camera mounting position. One example of taking advantage of this benefit is incorporating a height offset to develop a policy that is invariant to camera height. While the expected expert trajectory remains unaltered, an offset is integrated into the current pose for LRF rendering requests, such that $O_t = LRF(P_t, \omega_h)$, $P_t = (p_t^x, p_t^y, p_t^z + \epsilon, p_t^\phi, p_t^\theta, p_t^\psi)$, where ϵ is an offset parameter, ω_h is the horizontal FoV of the camera. In our method, a random z-axis offset is applied within a specific range in each episode, designed to operate without height information during testing, thus not included in the state representation.

Intrinsic: For intrinsic adaptability, we employ a camera-conditioned state representation, denoted as $S_t = CNN(O_t, C)$. The intrinsic parameter is provided to the LRF varying per episode in the range of $[\omega_{h,min}, \omega_{h,max}]$. An additional channel for the intrinsic condition $C = \frac{\omega_h - \omega_{h,min}}{\omega_{h,max} - \omega_{h,min}}$ is concatenated with O_t. Since intrinsic differences, like FoV, can influence policy outcomes—potentially causing incorrect actions such as premature or delayed turns at corners—they are incorporated as part of the state.

F. Implementation Details

Navigation Policy: Our experimental setup utilizes stable-baselines 3 [31]. The CNN architecture for feature extraction comprises three convolutional layers with kernel sizes of 8x8, 4x4, and 3x3, and strides of 4, 2, and 1, respectively, followed by an LSTM layer.

Offline Data for BC: Prior to the RL training phase, an offline dataset for BC is created. The preparation of this dataset incorporates rendering with offsets and the camera-conditioned state representation similarly.

Image Data Augmentation: Following NeRF2Real [9], our methodology applies image augmentations to enhance the diversity of the training data, including randomized adjustments to brightness, saturation, and hue. We refrain from employing geometric transformations, such as image rotation, to maintain the integrity of spatial relationships.

Motion Controller: During testing, a motion controller is essential for translating output actions by the policy into control commands suitable for the mobile robot's configuration. Our experimental setup employs a motion controller designed for a differential two-wheeled robot, issuing commands for linear and angular velocities to follow a dynamically specified target position (x, y, ψ) within the odometry frame.

Inference Interval: The inference interval during testing denotes the duration until the robot achieves a predetermined movement amount as dictated by the action output or reaches a timeout (set to 0.5 seconds for our experiments).

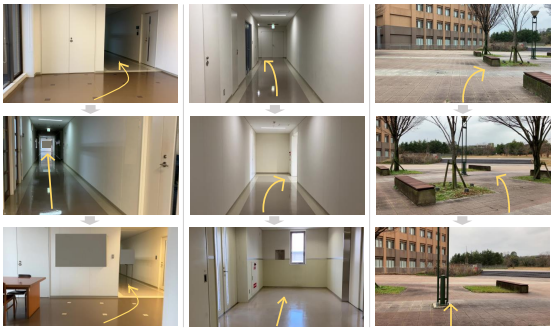

Fig. 3. The experimental environments were two indoor corridors and a courtyard in the university.

TABLE I

DATASET FOR EVALUATION

No.	Use	Device	Height	FoV
1	Training	iPhone	120 cm	60 deg.
2	Test	iPhone	70 cm	60 deg.
3	Test	iPhone	120 cm	60 deg.
4	Test	iPhone	170 cm	60 deg.

IV. EXPERIMENTS

We conducted experiments to reveal questions as follows:

- Can EBIAN offer enhanced robustness to variations in robot configurations? (Q1)
- Is the policy developed through EBIAN effective in navigating a real-world robot? (Q2)

To address Q1, we executed a series of closed-loop simulation experiments. In these, the agent was trained within a single NeRF-generated environment and subsequently deployed in other NeRF environments created from video sequences captured under varied conditions. The specifics of these experiments are detailed in Section IV-B. For Q2, we undertook real-world navigation trials using an actual mobile robot, as outlined in Section IV-C. These experiments aimed to assess the practical applicability and effectiveness of the EBIAN-trained policy in real-world settings.

A. Training Setup

For our experiments, we utilized servers equipped with NVIDIA A6000 GPUs to train the navigation policy. The selected image resolution for this process was 96×54. The range of extrinsic offsets applied was $[-0.25, 0.25]$ relative to the scale of the radiance fields, explicitly affecting the robot's height direction. The simulation predefined FoV, with this study selecting three variations: 50, 60, and 70 degrees (deg.). The stride between two poses within the expert trajectory was set to 0.25 on the scale of the radiance fields. The numbers of RL training steps were the same at each environment among the methods, ensuring adequate rewards. The numbers of BC epochs was the same among the methods, and models with sufficiently small losses were used.

B. Simulation

Setup: The simulation evaluation utilized experimental environments as depicted in Fig. 3. Data collection involved individuals navigating the course while recording with an iPhone SE3, whose FoV is approximately 60 deg., with the OSMO Mobile SE gymbal to minimize rotation around pitch and roll, at different heights: 120 centimeters (cm) for training and 70 cm, 120 cm (another set), and 170 cm for testing, as detailed in Table I. The recording height of the camera was fixed using a camera height adjuster. Each data was acquired during the same time period. This data was processed through LRF to generate radiance fields and camera trajectories, serving as ground truth for simulation evaluation. The objective was to assess the generalizability of the policy across different camera mounting heights and FoVs from the training data.

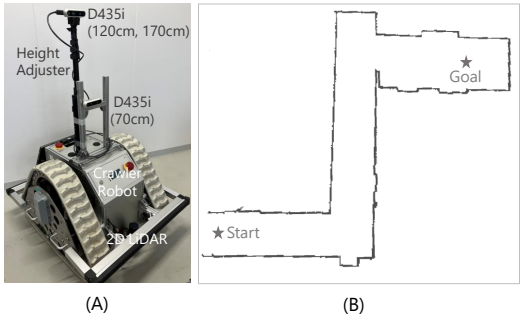

Fig. 4. (A) The crawler robot with the camera sensor (RealSense D435i). (B) The test course with the actual robot.

TABLE II

COMPARISON METHODS

Method	Image	Extrinsic	Intrinsic	SAC
BC (w/ Raw)	Raw	No	No	No
BC (w/ NeRF)	NeRF	No	No	No
EBIAN (-SAC)	NeRF	Yes	Yes	No
EBIAN (-ext)	NeRF	No	Yes	Yes
EBIAN (-int)	NeRF	Yes	No	Yes
EBIAN	NeRF	Yes	Yes	Yes

The policy, trained using radiance fields at a 120 cm camera height, was tested across fields at 70 cm, 120 cm (another set), and 170 cm heights. FoV variations were introduced by adjusting the LRF rendering FoV to 50 deg. and 70 deg. Based on the initial pose of the camera trajectory estimated by LRF, we conducted 25 trials with offsets ranging from -0.1 m to 0.1 m in 0.05 m increments in the left-right direction and -0.1 rad to 0.1 rad in 0.05 rad increments in the yaw direction. The success rates of these trials were then compared. Success is defined as maintaining a distance of within 1.0 m between the expected pose and the current pose for the entire duration of the total number of steps.

Comparison methods are shown in the TABLE II. Behavioral cloning (BC) which has the same network layer as the actor in EBIAN is set as the baseline. BC assumes expert trajectory acquisition via visual odometry, utilizing LRF-estimated trajectories as labels of training data. BC (w/ Raw) is trained by raw images as the observation, while BC (w/ NeRF) is trained by images rendered by NeRF. EBIAN (-SAC) is set to reveal that RL (SAC) is taking advantage of

TABLE III

SUCCESS RATES OF 25 TRIALS CONDUCTED UNDER DIFFERENT CAMERA HEIGHT CONDITIONS

	Corridor 1 (Indoor)			Corridor 2 (Indoor)			Courtyard (Outdoor)		
	60 deg.			60 deg.			60 deg.		
	70 cm	120 cm	170 cm	70 cm	120 cm	170 cm	70 cm	120 cm	170 cm
BC (w/ Raw)	0.00	0.00	0.00	0.08	0.00	0.00	0.12	0.00	0.00
BC (w/ NeRF)	0.44	0.44	0.00	0.00	0.00	0.00	0.76	0.64	0.00
EBIAN (-SAC)	0.24	0.28	0.04	0.08	0.00	0.00	0.64	0.48	0.68
EBIAN (-ext)	0.20	**1.00**	0.08	0.00	0.00	0.00	0.56	**1.00**	0.24
EBIAN	**0.52**	**1.00**	**0.96**	**0.84**	**0.32**	**0.12**	**0.88**	**1.00**	**0.92**

TABLE IV

SUCCESS RATES OF 25 TRIALS CONDUCTED UNDER DIFFERENT CAMERA FOV CONDITIONS

	Corridor 1 (Indoor)		Corridor 2 (Indoor)		Courtyard (Outdoor)	
	120 cm		120 cm		120 cm	
	50 deg.	70 deg.	50 deg.	70 deg.	50 deg.	70 deg.
BC (w/ Raw)	0.16	0.00	0.00	0.08	0.08	0.00
BC (w/ NeRF)	0.32	0.20	0.00	0.00	1.00	0.08
EBIAN (-SAC)	0.52	0.52	0.00	0.00	0.44	0.40
EBIAN (-int)	0.68	**1.00**	**0.68**	0.00	**1.00**	0.00
EBIAN	**1.00**	**1.00**	0.60	**0.24**	**1.00**	**1.00**

the ability to collect data in poses not included in the expert trajectory. EBIAN (-ext) is set to reveal the effectiveness of the offset simulation by subtracting extrinsic offsets, and the policy is trained only in 120 cm of capturing height. EBIAN (-int) is without intrinsic offsets, and the policy is trained only in 60 deg. FoV of iPhone.

Results: From the TABLE III, EBIAN demonstrated superior success rates across all environments and camera height conditions. EBIAN can generalize across all heights in all environments. Despite the diminished rendering quality from NeRF with increased extrinsic offsets, i.e., the further away from the estimated camera trajectory, the training using these images proved effective for generalizing across variations in camera height.

From the TABLE IV, EBIAN success rates show that the score is high in all but one case in corridor 2. It can be seen that EBIAN is also effective for the generalization of FoV conditions, however, in the corridor 2 environment, the success rates were lower compared to other environments, possibly due to lower reconstruction quality by NeRF in certain areas, affecting navigation learning accuracy. The cause analysis of failure is explained in Section IV-D.

From all scores shown in TABLE III and IV, we found that EBIAN could offer enhanced robustness to variations in robot configurations.

C. Real-World Robot Navigation

Setup: The real-world navigation experiments using EBIAN, utilized a two-wheeled mobile robot equipped with a RealSense D435i camera, as illustrated in Fig. 4 (A). As in the simulation experiment, the camera mounting height could be changed to 70 cm, 120 cm, and 170 cm conditions by the camera height adjuster. For the FoV adjustment, images from the D435i, whose FoV is approximately 70 deg., were cropped to 50 and 60 deg. FoV and used as observation. The robot's movement was governed by linear velocity and angular velocity, with the motion controller (depicted in Fig. 2) converting the output of RL policy (x, y) into linear velocity and angular velocity. The experiments were conducted using an NVIDIA Jetson AGX Xavier embedded within the robot. The scale value of the real environment corresponding to the radiance fields was approximately 2.6, and the stride between two poses within the expert trajectory was set to 0.25 m, therefore, the stride of the actual robot was approximately 0.65 m. The navigation course, shown in Fig. 4 (B), corresponds to corridor 2 from Fig. 3. Each condition underwent three trials, evaluating performance by the success rates and the average rate of progress, which was determined by the distance progressed over the total distance moved. Trials concluded upon reaching the goal or when a potential collision, approaching within about 10 cm from a wall, was anticipated. Failures attributed to immediate excessive turning post-start were considered outliers. Note that the 2D LiDAR in Fig. 4 (A) was for plotting the trajectory of movement and was not used for autonomous navigation, and depths from the D435i were also not used.

Results: TABLE V shows the success rates of real-world navigation at each condition. Numbers in parentheses indicate the average rate of progress. We found that the policy learned by EBIAN can navigate the actual robot, although the success rates have not reached 1.00.

Fig. 5 showcases the outcomes of the real-world robot navigation trials with EBIAN, highlighting two successful attempts on the left and an unsuccessful trial on the right. The successful trials demonstrate the robot's capability to navigate straight paths and execute turns at appropriate junctures under various camera height conditions. Notably, in Fig. 5 (1), the robot adeptly corrects its course from a non-expert state, such as facing a wall, indicative of the effectiveness of free-viewpoint rendering by NeRF in simulating unseen demonstration conditions. However, in Fig. 5 (2), at a camera mounting height of 120 cm and FoV of 50 deg., the robot failed to execute a timely turn at the second corner. The same behavior was observed at a camera mounting height of 120 cm and FoV of 70 deg. The cause analysis of failure is explained in Section IV-D. Successful trials were observed

TABLE V

SUCCESS RATES OF 3 TRIALS FOR REAL-WORLD NAVIGATION. (THE AVERAGE RATES OF PROGRESS)

	Corridor 2				
	70 cm	120 cm			170 cm
	60 deg.	50 deg.	60 deg.	70 deg.	60 deg.
EBIAN	0.67 (0.80)	0.00 (0.58)	0.33 (0.88)	0.00 (0.71)	0.00 (0.61)

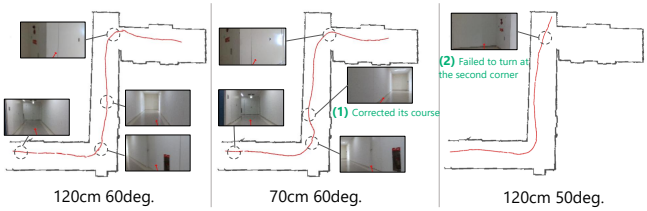

Fig. 5. Experimental results on actual robot navigation. The map is made by SLAM in advance just to record the trajectory of the movement, the map and pose on it do not affect the autonomous navigation. Red lines indicate the trajectory estimated by AMCL in each trial. The observed image of the vicinity is indicated by the dashed circle, and its action is superimposed with red arrows.

with 70 cm and 120 cm of camera mounting height and FoV of 60 deg. conditions, other conditions were none, which may have been influenced by different types of cameras. This is also explained in Section IV-D.

D. Cause analysis of failure

Failed results, noticeable in corridor 2 across both simulation and real-world experiments, are believed to stem from inaccuracies in NeRF reconstruction. Fig. 6 illustrates a simulation failure in corridor 2, where Fig. 6 (B) captures a significant camera pose jump mid-turn, marked by a dashed circle. This inaccuracy leads to a notable degradation in image quality when comparing the original (A) and the NeRF-rendered image (C), suggesting that navigating within such an inaccurately reconstructed environment could yield unstable outcomes due to the divergence from real-world appearances. Similar failures were observed in real-world navigation tests, as shown in Fig. 5 (2), where the robot failed to timely navigate a turn. This parallels the simulation issue (Fig. 6), underscoring the pivotal role of reconstruction precision of NeRF in the successful navigation of EBIAN.

The differences of camera specifications, except for FoV, also contribute to navigation failures. Additional tests by the trained policy in iPhone-generated radiance fields, within radiance fields generated from D435i-captured videos, indicated a performance drop compared to the results of the test in iPhone-generated radiance fields, as seen in TABLE VI. The use of D435i, differing from the iPhone, underscores the necessity for policies to accommodate camera-specific differences, including resolution and color, beyond FoV. While EBIAN accounts for intrinsic offsets, further refinement is needed for agents to discern and adapt to camera model variations, with potential solutions like [32] proposed to address these challenges.

Therefore, the enhancement of NeRF is closely linked to the progress in navigation capabilities through EBIAN.

TABLE VI

SUCCESS RATES OF 25 TRIALS SIMULATION COMPARING OTHER TYPES OF CAMERAS

	Corridor 2				
	70 cm	120 cm			170 cm
	60 deg.	50 deg.	60 deg.	70 deg.	60 deg.
EBIAN (w/ iPhone)	0.84	0.60	0.32	0.24	0.12
EBIAN (w/ D435i)	0.28	0.08	0.00	0.00	0.00

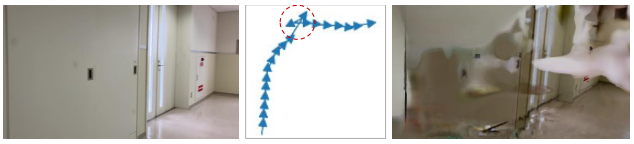

Fig. 6. (A) The original image. (B) The camera trajectory around the original image. (C) The rendered image is qualitatively low.

V. LIMITATION

Our approach demonstrated the capability of an RL agent to learn navigation policies without direct interaction with the real world. Nonetheless, as highlighted in Section III-B, addressing scale inconsistency, potentially through automated scale optimization based on keyframe matching, remains an area for improvement. Additionally, the agent's adaptability to dynamic environmental factors—such as changes in weather, time of day, seasons, and the presence of moving objects—could be enhanced by incorporating techniques like appearance embedding [32] and spatial change representation [33]. Improving the 3D reconstruction quality of NeRF is essential, as the navigation accuracy heavily depends on this aspect, as noted in Section IV-D. Thus, advancements in the reconstruction accuracy and rendering quality [34] are anticipated to further refine navigation precision. Future developments should also focus on bridging the gap between the cameras used during the training and testing phases.

VI. CONCLUSION

This work introduced a novel one-shot imitation learning framework leveraging NeRF to facilitate policy learning without necessitating real-world interaction by the robot. Our research particularly addressed the adaptability of this method to variations in the intrinsic and extrinsic parameters of camera sensors equipped on robots, demonstrating robust performance. Furthermore, we verified the effectiveness of our framework with an actual mobile robot in real-world navigation scenarios. While our approach primarily focused on simple policy learning, generating single-step motion actions from visual observations, there exists the potential for refinement, such as implementing strategies that output multi-step trajectories or actions aimed at minimizing deviations from expected imagery.

Although our evaluation was confined to a two-wheeled robot, the underlying policy might apply to other robotic platforms, including holonomic robots and drones with the single policy learned by EBIAN. Extending our evaluation to encompass these additional robot types constitutes a direction

for future research. Moreover, we aim to explore the comparative efficacy of our framework against traditional imitation learning and VT&R approaches, which rely on state-action pair data, in various types of simulated environments.

ACKNOWLEDGMENT

This work was partially supported by JSPS KAKENHI Grant Number JP21K18701 and a collaborative research with RICOH Co., Ltd.

REFERENCES

[1] M. Bojarski, D. D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba, "End to end learning for self-driving cars," 2016.

[2] A. Kendall, J. Hawke, D. Janz, P. Mazur, D. Reda, J.-M. Allen, V.-D. Lam, A. Bewley, and A. Shah, "Learning to drive in a day," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 8248–8254.

[3] A. Hu, G. Corrado, N. Griffiths, Z. Murez, C. Gurau, H. Yeo, A. Kendall, R. Cipolla, and J. Shotton, "Model-Based Imitation Learning for Urban Driving," Oct. 2022. [Online]. Available: https://arxiv.org/abs/2210.07729v2

[4] C. Finn, T. Yu, T. Zhang, P. Abbeel, and S. Levine, "One-shot visual imitation learning via meta-learning," in *Proceedings of the 1st Annual Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, S. Levine, V. Vanhoucke, and K. Goldberg, Eds., vol. 78. PMLR, 13–15 Nov 2017, pp. 357–368. [Online]. Available: https://proceedings.mlr.press/v78/finn17a.html

[5] F. Codevilla, M. Müller, A. López, V. Koltun, and A. Dosovitskiy, "End-to-end driving via conditional imitation learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 4693–4700.

[6] C.-H. Lin, W.-C. Ma, A. Torralba, and S. Lucey, "Barf: Bundle-adjusting neural radiance fields," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 5721–5731.

[7] A. Meuleman, Y.-L. Liu, C. Gao, J.-B. Huang, C. Kim, M. H. Kim, and J. Kopf, "Progressively optimized local radiance fields for robust view synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 16 539–16 548.

[8] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[9] A. Byravan, J. Humplik, L. Hasenclever, A. Brussee, F. Nori, T. Haarnoja, B. Moran, S. Bohez, F. Sadeghi, B. Vujatovic, and N. Heess, "Nerf2real: Sim2real transfer of vision-guided bipedal motion skills using neural radiance fields," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 9362–9369.

[10] Y. Liu, A. Gupta, P. Abbeel, and S. Levine, "Imitation from observation: Learning to imitate behaviors from raw video via context translation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 1118–1125.

[11] F. Torabi, G. Warnell, and P. Stone, "Generative adversarial imitation from observation," June 2019. [Online]. Available: http://nn.cs.utexas.edu/?ICML19a-torabi

[12] ——, "Recent advances in imitation learning from observation," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI)*, 2019, pp. 6325–6331. [Online]. Available: https://doi.org/10.24963/ijcai.2019/882

[13] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[14] M. Labbé and F. Michaud, "Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation," *Journal of Field Robotics*, vol. 36, no. 2, pp. 416–446, 2019.

[15] M. Mahdavian, K. Yin, and M. Chen, "Robust visual teach and repeat for ugvs using 3d semantic maps," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8590–8597, 2022.

[16] D. Dall'Osto, T. Fischer, and M. Milford, "Fast and Robust Bio-inspired Teach and Repeat Navigation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Prague, Czech Republic: IEEE, Sep. 2021, pp. 500–507.

[17] H. Karnan, G. Warnell, X. Xiao, and P. Stone, "Voila: Visual-observation-only imitation learning for autonomous navigation," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 2497–2503.

[18] M. Fehr, T. Schneider, M. Dymczyk, J. Sturm, and R. Siegwart, "Visual-inertial teach and repeat for aerial inspection," 2018.

[19] K. Rao, C. Harris, A. Irpan, S. Levine, J. Ibarz, and M. Khansari, "Rl-cyclegan: Reinforcement learning aware simulation-to-real," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[20] D. Ho, K. Rao, Z. Xu, E. Jang, M. Khansari, and Y. Bai, "Retinagan: An object-aware approach to sim-to-real transfer," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 10 920–10 926.

[21] X. Xu, M. You, H. Zhou, Z. Qian, and B. He, "Robot imitation learning from image-only observation without real-world interaction," *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 3, pp. 1234–1244, 2023.

[22] A. Bewley, J. Rigley, Y. Liu, J. Hawke, R. Shen, V.-D. Lam, and A. Kendall, "Learning to drive from simulation without real world labels," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 4818–4824.

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

[24] M. Wulfmeier, A. Bewley, and I. Posner, "Addressing appearance change in outdoor robotics with adversarial domain adaptation," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 1551–1558.

[25] "Luma AI," https://lumalabs.ai/.

[26] M. Tancik, E. Weber, E. Ng, R. Li, B. Yi, J. Kerr, T. Wang, A. Kristoffersen, J. Austin, K. Salahi, A. Ahuja, D. McAllister, and A. Kanazawa, "Nerfstudio: A modular framework for neural radiance field development," in *ACM SIGGRAPH 2023 Conference Proceedings*, ser. SIGGRAPH '23, 2023.

[27] P. Gesel, N. Sojib, and M. Begum, "Self-supervised visual motor skills via neural radiance fields," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 3712–3718.

[28] M. Adamkiewicz, T. Chen, A. Caccavale, R. Gardner, P. Culbertson, J. Bohg, and M. Schwager, "Vision-only robot navigation in a neural radiance world," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4606–4613, 2022.

[29] S. Mysore, B. Mabsout, R. Mancuso, and K. Saenko, "Regularizing action policies for smooth control with reinforcement learning," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 1810–1816.

[30] Y. Lu, J. Fu, G. Tucker, X. Pan, E. Bronstein, R. Roelofs, B. Sapp, B. White, A. Faust, S. Whiteson, D. Anguelov, and S. Levine, "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)*, 2023, pp. 7553–7560.

[31] 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

[32] R. Martin-Brualla, N. Radwan, M. S. M. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth, "Nerf in the wild: Neural radiance fields for unconstrained photo collections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 7210–7219.

[33] Z. Li, S. Niklaus, N. Snavely, and O. Wang, "Neural scene flow fields for space-time view synthesis of dynamic scenes," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 6494–6504.

[34] R. Wu, B. Mildenhall, P. Henzler, K. Park, R. Gao, D. Watson, P. P. Srinivasan, D. Verbin, J. T. Barron, B. Poole, and A. Holynski, "Reconfusion: 3d reconstruction with diffusion priors," 2023.