2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2025)
October 19-25, 2025. Hangzhou, China

2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 979-8-3315-4393-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/IROS60139.2025.11247117

C-TRAC: Terrain-Adaptive Control for Articulated Tracked Robots via Contact-Aware Reinforcement Learning

Hainan Pan Kaihong Huang Xieyuanli Chen Hongchuan Zhang Junfeng Shi
Chuang Cheng Bailiang Chen Huimin Lu*

Abstract— **Articulated tracked robots face significant challenges in maintaining stable locomotion over uneven terrain due to unknown contact points between tracks and ground, which are critical for dynamic control. Unlike legged robots, where contact locations can be predicted, tracked systems require real-time adaptation to varying terrains. This paper presents C-TRAC, a terrain-adaptive control framework that integrates reinforcement learning with a contact-modeling variational autoencoder (C-VAE) to enable robust obstacle traversal. We first train a C-VAE in simulation to reconstruct high-fidelity contact information (position and binary probability) from noisy sensor measurements. This model learns a latent representation of terrain contacts, capturing complex interactions between the robot's kinematics and environment. Subsequently, we employ an asymmetric Soft Actor-Critic (SAC) algorithm to optimize a control policy that leverages the predicted contact data for adaptive track control during locomotion. Extensive experiments validate C-TRAC in both simulated and real-world scenarios. In benchmark tests against state-of-the-art (SOTA) methods using RoboCup Rescue Robot League environments, our approach achieves superior obstacle traversal speed (up to 66.67% faster on** 45° **staircase) and stability (up to 47.53% more stable on the oblique terrace) compared to contact-agnostic RL baselines and model-based methods. Notably, zero-shot sim-to-real transfer demonstrates consistent performance in unstructured outdoor ruins, also confirming the framework's practicality.**

I. Introduction

Articulated tracked robots equipped with four rotatable flippers demonstrate exceptional terrain traversal capabilities in urban search and rescue (USAR) missions, enabling efficient navigation through complex uneven terrains [1]. However, the high-dimensional control freedom of the multi-flipper system presents significant operational challenges. Traditional model-based flippers control methods [2], [3], [4] depend on geometric and kinematic assumptions, struggling to adapt to unstructured environments. Deep reinforcement learning (DRL) methods [5] avoid explicit modeling by directly learning terrain interaction policies, such as stair climbing [6] and terrain elevation fusion frameworks [7], [8], [9]. However, existing DRL methods remain constrained by parameter sensitivity, sim-to-real transfer gaps [10], and limited generalization to 3D terrains. While techniques like intrinsic curiosity [11], [12] and domain randomization (DR) [13] improve exploration efficiency, their performance is still hindered by explicit terrain representations.

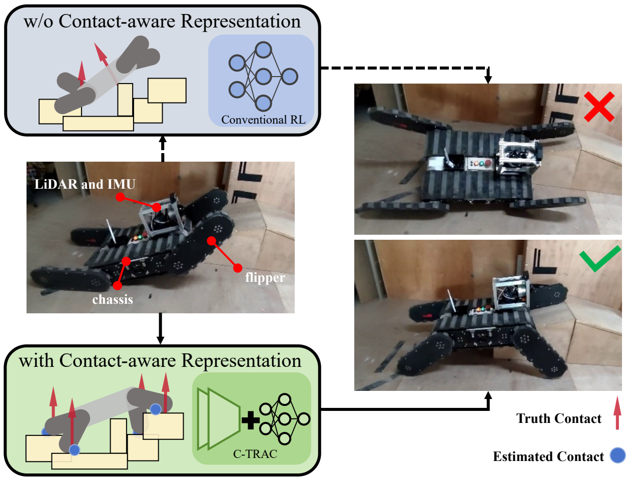

Fig. 1: Conventional RL methods struggle to capture dynamically varying contact interactions, resulting in suboptimal performance. We employ contact-aware representation learning to explicitly estimate physical contact relationships and thereby improve the overall RL performance.

Recent studies bridge sim-to-real gaps through implicit environmental representations by autoencoders to infer terrain properties (e.g., friction coefficients, height maps) from proprioception and employ asymmetric Actor-Critic architectures. Such approaches enable robust locomotion on complex terrains for legged robots, such as quadruped robots [14], [15], [16] and humanoid robots [17].

As shown in Fig. 1, the contact dynamics between the robot and the terrain dictate the locomotion performance for flipper-tracked robots. However, unlike the static contacts in legged robots (where contact points are localized at specific foot-end locations), such contact relationships are inherently nonlinear, dynamic, and challenging to model explicitly or measure in real-world scenarios, exacerbating sim-to-real difficulties.

Building on implicit representation learning advances [14], [17], this work proposes a novel VAE-enhanced RL framework that integrates implicit contact dynamics modeling for articulated track control in uneven terrains. Unlike conventional approaches constrained by predefined terrain models [6], [18] or explicit feature extraction pipelines [15], our two-stage approach first pre-trains an implicit contact-VAE to jointly learn obstacle-crossing dynamics from fused proprioception and terrain perception through denoising au-

All authors are with the College of Intelligence Science and Technology, and the National Key Laboratory of Equipment State Sensing and Smart Support, National University of Defense Technology, China.

This work has partially been funded by the National Science Foundation of China under Grant U22A2059, 62403478, and 62203460, and Young Elite Scientists Sponsorship Program by CAST (No. 2023QNRC001).

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on January 16,2026 at 15:48:09 UTC from IEEE Xplore. Restrictions apply.

toencoding, while explicitly estimating contact states. The latent representations then drive joint optimization with an asymmetric Soft Actor-Critic (SAC) architecture, enhancing obstacle traversal performance and zero-shot sim-to-real transfer capability.

In summary, the contributions of this work are threefold:

1) A novel autonomous obstacle-crossing learning two-stage framework, C-TRAC, for flipper-tracked robots is proposed. It integrates proprioceptive data and local terrain perception to learn obstacle-crossing features implicitly. Pre-trained C-VAE is jointly optimized with an asymmetric SAC to develop articulated track control policies, enabling stable and efficient traversal of uneven terrains.
2) A new contact estimation module that incorporates dynamic masking and a hybrid loss function with geometric constraints is introduced. This module estimates contact relationships from multi-frame observations, addressing the challenge of inaccessible contact information in tracked robots while enhancing the extraction of implicit obstacle-crossing features.
3) Extensive evaluations in diverse environments demonstrate superior traversal performance compared to SOTA methods such as [4], [9]. Outdoor zero-shot testing validates C-TRAC's superior sim-to-real transfer and generalization capabilities.

II. Related Work

Articulated tracked robots with flippers have demonstrated potential in complex terrains like disaster relief scenarios. Although traditional model-based control approaches relying on manually crafted geometric models [4], [18] or kinematic formulations [2], [3], [19] retain rigorous reliability through explicit stability margins, their analytical nature fundamentally restricts deployment to structured environments with known terrain properties. Recent advances leverage RL to address non-linear terrain interactions without manual modeling. Mitriakov et al. [6] trained RL policies for stair traversal, while Zimmermann et al. [8] and Pecka et al. [7] integrated terrain elevation maps and safety constraints into RL frameworks. In addition, some RL efforts focus on open and standardized simulation platforms for tracked robots with flippers to advance the development of adaptive control strategies [9], [20]. However, RL-based methods face challenges, including parameter sensitivity, data inefficiency, and sim-to-real gaps [10]. Recent innovations in DR technique [13] have improved exploration and generalization.

Recent studies emphasize learning latent environmental representations to bridge sim-to-real gaps. EstimatorNet [15] trains a policy and state estimator jointly, using a deterministic auto-encoder to reduce foot clearance errors and improve robustness on slippery surfaces. DreamWaQ [14] employs an asymmetric Actor-Critic architecture with a context-aided estimator network (CENet) and a β-VAE to infer terrain properties from proprioception, enabling robust locomotion on diverse terrains. The CENet jointly optimizes body velocity estimation and observation reconstruction via a hybrid loss, enhancing generalization [15]. Similarly, RL2AC [16] integrates a VAE-based encoder-decoder to estimate latent dynamics and adaptively compensates for torque uncertainties through composite adaptive control, achieving rapid online adaptation in unstructured environments. For humanoid robots, DWL [17] introduces a denoising world model combining Gated Recurrent Unit (GRU-based) encoders and VAEs to reconstruct full states from noisy observations. By masking privileged information and applying DR, DWL achieves zero-shot sim-to-real transfer on complex terrains.

III. Preliminaries

A. Problem Definition

The NuBot-Rescue robot[1] features a dual-tracked configuration augmented with four independently controllable flippers. In addition, it is equipped with a LiDAR sensor and an inertial measurement unit (IMU) for real-time local mapping and accurate pose estimation. As shown in Fig. 1, this platform has exceptional traversability and transport capabilities.

This paper aims to fully exploit the obstacle-crossing potential of flipper-tracked robots by developing control policies that dynamically adapt flipper rotation angles and linear velocity in real-time, enabling straight-path traversal across unstructured terrains.

B. RL preliminaries

Following [14], we formulate the environment as an infinite-horizon partially observable Markov decision process (POMDP), characterized by the tuple $M = (S, \mathcal{O}, \mathcal{A}, d_0, P, r, \gamma)$. The full state $\mathbf{s} \in S$, partial observation $\mathbf{o} \in \mathcal{O}$, and action $\mathbf{a} \in \mathcal{A}$ are all continuous. The environment is initialized with an initial state distribution $d_0(\mathbf{s}_0)$, evolves according to the state transition probability $P(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$, and yields rewards governed by the reward function $r : S \times \mathcal{A} \to \mathcal{R}$. The discount factor is denoted by $\gamma \in [0, 1)$. A temporal observation at time t over a total number of H past measurements is denoted as $\mathbf{o}_t^H = [\mathbf{o}_t, \mathbf{o}_{t-1}, \cdots \mathbf{o}_{t-H}]$.

IV. Our Approach

The overall C-TRAC architecture consists of two stages: C-VAE network pre-training and control policy training. In the first stage, C-VAE is trained on a dataset of successful locomotion trajectories performed by terrain-specific(including stairs and ramps in simulation) flipper control policies to learn the implicit representation of contact dynamics more stably. In the second stage, the pre-trained C-VAE is integrated into an asymmetric SAC network and jointly optimized across diverse simulation scenarios. In this section, we will first introduce the learning process of a flipper control policy based on asymmetric SAC in Stage II and then the implicit model learning process of C-VAE in Stage I.

[1] https://tdp.robocup.org/tdp/2023-tdp-nubot-rescue-robocuprescue-robot

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on January 16,2026 at 15:48:09 UTC from IEEE Xplore. Restrictions apply.

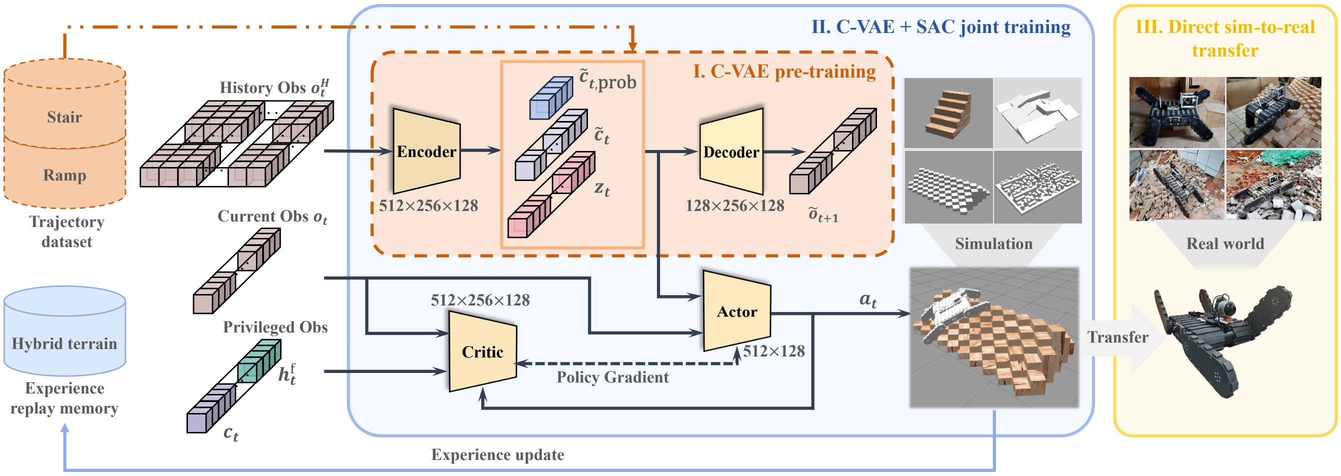

Fig. 2: The overall C-TRAC architecture consists of two stages: "I. C-VAE network pre-training" and "II. C-VAE + SAC joint training". The training alternates between SAC policy updates (1 iteration) and C-VAE contact model refinements (5 iterations) to balance exploration and representation learning.

A. *Control Policy Learning with Privileged Observation*

In the Actor-Critic algorithm, the interplay between the policy and value networks is sufficient to learn a robust locomotion strategy that can implicitly imagine privileged observations given partial temporal observations [14]. This study employs an asymmetric Actor-Critic architecture to implicitly imagine the contact relationship between the robot and the terrain. Given that the strong exploratory nature of SAC is beneficial for obstacle-crossing training of flipper-tracked robots [9], the policy is optimized using the SAC algorithm.

*1) **State Space**:* The partial observation $\mathbf{o}_t$ used to input the Actor includes the proprioceptive robot state and the perceived height map $\mathbf{h}_t^l$ of the terrain in front of the robot:

$$\mathbf{o}_t = [v_t \quad \theta_t^{\mathbf{fl}} \quad \theta_t^{\mathbf{rl}} \quad \theta_t^{\mathbf{rr}} \quad \theta_t^{\mathbf{fr}} \quad \mathbf{b}_t \quad \mathbf{p}_t \quad \mathbf{h}_t^{\mathrm{l}}]^T, \tag{1}$$

where v_t is the desired linear velocity along the X-axis, $\theta_t^{\mathbf{fl}}$, $\theta_t^{\mathbf{rl}}$, $\theta_t^{\mathbf{rr}}$, and $\theta_t^{\mathbf{fr}}$ are the rotated angles of the four flippers, $\mathbf{b}_t$ denotes the Euler angles of the robot's chassis, $\mathbf{p}_t$ denotes the relative position of the robot about the goal in the XY-plane. Height map $\mathbf{h}_t^l$ is centered on the robot and spans a length range of $[0.4\,\mathrm{m}, 1.0\,\mathrm{m}]$ and a width range of $[-0.5\,\mathrm{m}, 0.5\,\mathrm{m}]$.

The full state $\mathbf{s}_t$ given by the simulator, which includes additional privileged observation, can be defined as follows:

$$\mathbf{s}_t = [\mathbf{o}_t \quad \mathbf{c}_t \quad \mathbf{h}_t^{\mathrm{f}}]^T, \tag{2}$$

where $\mathbf{h}_t^f$ is a larger terrain height map of range $[-1.0\,\mathrm{m}, 1.4\,\mathrm{m}]$, and $\mathbf{c}_t$ is a vector containing contact point positions information.

*2) **Action Space**:* The action space consists of the desired velocity of the chassis v_t and the delta rotation angles of the four flippers, which are defined as

$$\mathbf{a}_t = [v_t \quad \Delta\theta_t^{\mathbf{fl}} \quad \Delta\theta_t^{\mathbf{rl}} \quad \Delta\theta_t^{\mathbf{rr}} \quad \Delta\theta_t^{\mathbf{fr}}]^T. \tag{3}$$

*3) **Actor-Critic Architecture**:* The actor is a neural network $\pi_\psi(\mathbf{a}_t|\mathbf{o}_t, \mathbf{z}_t, \tilde{\mathbf{c}}_t, \tilde{\mathbf{c}}_t^{\mathrm{prob}})$ with parameter ψ that produces an action $\mathbf{a}_t$ based on a partial observation $\mathbf{o}_t$, latent vector $\mathbf{z}_t$, contact points $\tilde{\mathbf{c}}_t$ and the binary probability of contact existence $\tilde{\mathbf{c}}_t^{\mathrm{prob}}$. Here, $\tilde{\mathbf{c}}_t$, $\tilde{\mathbf{c}}_t^{\mathrm{prob}}$ and $\mathbf{z}_t$ are estimated by C-VAE. The critic employs the Q-value function $Q_\phi(\mathbf{s}_t, \mathbf{a}_t)$ to estimate state values, utilizing the complete state-action pair $(\mathbf{s}_t, \mathbf{a}_t)$ that includes privileged observations.

B. *Reward Formulation*

The reward function aims to guide the robot to reach the target position quickly, maintain a stable orientation angle, and ensure necessary contact, enabling robust locomotion in rugged and challenging terrain environments.

1) Progress Penalty: Based on the potential-based reward shaping (PBRS) [21], a negative reward for the distance on the X-axis from the target position is designed as the rapidity reward r_{v}:

$$r_{\mathrm{v}} = p_t^x - 1, \tag{4}$$

where $r_{\mathrm{v}} \in [-1, 0]$. p_t^x is the value of $\mathbf{p}_t$ on the X axis, representing the locomotion progress of the robot.

2) Posture Swing Penalty: To address the issue of the robot's orientation angle swing during obstacle-crossing, we introduce the short-term swing and the long-term average swing of the pitch and roll angles, which are defined as follows:

$$\begin{aligned} \Delta|\mathbf{b}_{i,t}| &= |\mathbf{b}_{i,t+1}| - |\mathbf{b}_{i,t}|, \\ \Delta\mathbf{b}_{i,t}^k &= \frac{1}{k-1}\sum_{j=t}^{t+k-1} |\mathbf{b}_{i,j+1} - \mathbf{b}_{i,j}|, \end{aligned} \tag{5}$$

where $\forall i \in \{\mathrm{p}, \mathrm{r}\}$ denotes the orientation angle pitch and roll, k is the time step of long-term stability. The absolute value of the angular variation $\Delta|\mathbf{b}_{i,t}|$ quantifies the instantaneous posture stability of the robot and indicates whether the orientation angle is currently increasing ($\Delta|\mathbf{b}_{i,t}| > 0$) or decreasing ($\Delta|\mathbf{b}_{i,t}| < 0$). The average angular variation $\Delta\mathbf{b}_{i,t}^k$ captures the long-term posture stability of the robot throughout the obstacle-crossing process. The reward of

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on January 16,2026 at 15:48:09 UTC from IEEE Xplore. Restrictions apply.

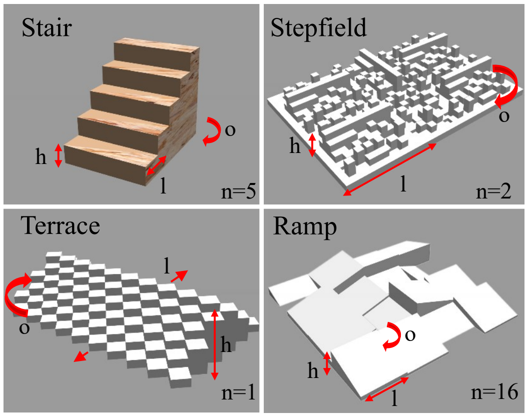

Fig. 3: The schematic diagram of terrain curriculum parameters. The l, h, o, and n represent, respectively, the length, height, orientation, and number of the whole or smallest unit of a given terrain.

shaking reduction r_s is defined:

$$\begin{aligned} r_\mathrm{s} &= \sum_{i=\mathrm{p,r}} \alpha_i \cdot r_{\mathrm{s},i}, \\ r_{\mathrm{s},i} &= \begin{cases} -1, & \text{if } |\mathbf{b}_{i,t}| > \mathbf{b}_{i,t}^{\max} \text{ and } \Delta|\mathbf{b}_{i,t}| > 0, \\ -|\Delta \mathbf{b}_{i,t}^{k}|, & \text{otherwise,} \end{cases} \end{aligned} \tag{6}$$

where $\alpha_\mathrm{p} = 1.5$ and $\alpha_\mathrm{r} = 1$. When the robot approaches a tipping threshold $\mathbf{b}_{\mathrm{p},t}^{\max} = 15°$ and $\mathbf{b}_{\mathrm{r},t}^{\max} = 2°$, it becomes critical to suppress further deterioration of the orientation angle.

3) Stabilization Penalty: To evaluate stabilization during obstacle traversing, the Normalized Energy Stability Margin (NESM) [3] E_nesm is adopted as a penalty metric r_c to encourage the robot to explore flipper motions with greater stability margins. We choose the minimum margin $E_\mathrm{nesm}^{\min}$ as the metric of penalty r_c:

$$r_\mathrm{c} = \begin{cases} -1, & \text{if } E_\mathrm{nesm} \nexists, \\ \mathrm{Norm}(E_\mathrm{nesm}^{\min}) - 1, & \text{otherwise.} \end{cases} \tag{7}$$

4) Termination Reward: When the distance between the robot and the target point approaches a certain distance, the agent will receive a positive reward. The agent will be penalized if the robot tips over or takes too large a time step. The termination reward r_T can be represented as follows:

$$r_\mathrm{T} = \begin{cases} 150, & \text{if reached;} \\ -150, & \text{if tipover;} \\ -225, & \text{if } t \geq 200. \end{cases} \tag{8}$$

C. Implicit Contact-VAE Model Learning

While RL enables versatile skill acquisition in simulation, real-world deployment faces challenges from dynamic contact interaction uncertainties during obstacle negotiation, particularly the difficulty of obtaining accurate contact states in unstructured environments.

We develop the C-VAE to simultaneously estimate contact states and infer latent dynamics representations during obstacle negotiation. Inspired by implicit terrain reasoning principles, the encoder-decoder architecture employs a denoising autoencoder mechanism to learn robot-terrain interaction dynamics from noisy multi-frame interaction data. Crucially, the latent space explicitly models contact state distributions, enabling joint optimization of dynamic feature extraction and contact-aware state estimation.

TABLE I: Parameter setting of the terrain during training.

Terrain	length [m]	height [m]	orientation [rad]	number	frequency
Stair	$\mathcal{U}(0.2, 0.5)$	$\mathcal{U}(0.15, 0.2)$	$\mathcal{U}(-\frac{\pi}{4}, \frac{\pi}{4})$	$\mathcal{U}(3, 7)$	0.2
Stepfield	1.2	$\mathcal{U}(0, 0.3)$	$\mathcal{U}(-\frac{\pi}{4}, \frac{\pi}{4})$	$\mathcal{U}(1, 3)$	0.25
Terrace	1.7	0.6	$\mathcal{U}(-\frac{\pi}{4}, \frac{\pi}{4})$	$\mathcal{U}(1, 2)$	0.25
Ramp	0.6	$\mathcal{U}(0, 0.4)$	$\mathcal{U}(-\pi, \pi)$	$\mathcal{U}(8, 16)$	0.3

The C-VAE employs a β-VAE architecture consisting of a single encoder and a multi-head decoder, as shown in Fig. 2. It can extract the latent feature vector $\mathbf{z}_t$, the estimated contact states $\tilde{\mathbf{c}}_t$ and $\tilde{\mathbf{c}}_t^\mathrm{prob}$ during obstacle crossing from the noisy historical multi-frame observations $\mathbf{o}_t^H$. The first head estimates $\tilde{\mathbf{c}}_t$ and $\tilde{\mathbf{c}}_t^\mathrm{prob}$, while the second head reconstructs and denoises the observation $\tilde{\mathbf{o}}_{t+1}$. The model can be expressed as follows:

$$P(\tilde{\mathbf{s}}_t) = \mathbb{E}_{\mathbf{o}_t^H} \left[\int_\mathbf{z} P_\mathrm{de}(\tilde{\mathbf{s}}_t | \mathbf{z}_t) \cdot P_\mathrm{en}(\mathbf{z}_t | \mathbf{o}_t^H) \right], \tag{9}$$

where $P(\tilde{\mathbf{s}}_t)$ denotes the estimation of the raw state distribution $P(\mathbf{s}_t)$ at time t. The encoder captures the conditional distribution P_en of these latent variables given the noisy historical observations $\mathbf{o}_t^H$, while the decoder P_de reconstructs the state from $\mathbf{z}_t$, $\mathbf{c}_t$, and $\mathbf{c}_t^\mathrm{prob}$. The C-VAE is optimized using a hybrid loss function, defined as follows:

$$\mathcal{L}_\mathrm{IC} = \mathcal{L}_\mathrm{VAE} + \mathcal{L}_\mathrm{C}, \tag{10}$$

where $\mathcal{L}_\mathrm{VAE}$ and $\mathcal{L}_\mathrm{C}$ are the VAE loss and the contact state estimation, respectively. The training loss of the β-VAE follows as [14], [16], and the loss $\mathcal{L}_\mathrm{C}$ of the contact state is defined as a hybrid function as follows:

$$\mathcal{L}_\mathrm{C} = \mathcal{L}_\mathrm{prob} + \mathcal{L}_\mathrm{est} + \mathcal{L}_\mathrm{geo}. \tag{11}$$

To address the intermittent nature of contact points during dynamic locomotion, we introduce a Binary Cross-Entropy (BCE) loss function, $\mathcal{L}_\mathrm{prob}$, which evaluates the discrepancy between the ground-truth probability of contact point existence $\mathbf{c}_i^\mathrm{prob}$ and the estimated probability $\tilde{\mathbf{c}}_i^\mathrm{prob}$. The loss function $\mathcal{L}_\mathrm{prob}$ is defined as follows:

$$\mathcal{L}_\mathrm{prob} = -\frac{1}{4} \sum_{i=1}^{4} \left[\mathbf{c}_i^\mathrm{prob} \log(\tilde{\mathbf{c}}_i^\mathrm{prob}) + (1 - \mathbf{c}_i^\mathrm{prob}) \log(1 - \tilde{\mathbf{c}}_i^\mathrm{prob}) \right] \tag{12}$$

Additionally, we propose a novel dynamic mask-weighted Mean Squared Error (MSE) loss function $\mathcal{L}_\mathrm{est}$, to ensure that the model only supervises the estimated contact points $\tilde{\mathbf{c}}_t$ against the ground-truth contact points $\mathbf{c}_t$ derived from simulation when contact points are present. The loss function

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on January 16,2026 at 15:48:09 UTC from IEEE Xplore. Restrictions apply.

$\mathcal{L}_{\text{est}}$ is formulated as:

$$\mathcal{L}_{\text{est}} = \sum_{i=1}^{4} m_i \ \mathrm{MSE}(\tilde{\mathbf{c}}_i, \mathbf{c}_i), \qquad m_i = \frac{\mathbf{c}_i^{\text{prob}}}{\sum_{j=1}^{4} \mathbf{c}_j^{\text{prob}}} \tag{13}$$

where m is the dynamic mask-weighted. Based on the geometric constraints Ω, which ensures contact points are located on the robot, we construct a loss function $\mathcal{L}_{\text{geo}}$ with spatial feasibility penalty for contact point estimation in supervised learning:

$$\mathcal{L}_{\text{geo}} = \sum_{i=1}^{4} m_i \ \mathbb{I}(\tilde{\mathbf{c}}_i, \Omega), \qquad \mathbb{I}(\mathbf{c}, \Omega) = \begin{cases} 0 & \mathbf{c} \in \Omega; \\ d(\mathbf{c}, \partial\Omega) & \mathbf{c} \notin \Omega \,. \end{cases}, \tag{14}$$

where $\mathbb{I}(\cdot)$ is the penalty function, $d(\mathbf{c}, \partial\Omega)$ denotes the Euclidean distance from $\mathbf{c}_t$ to region boundary $\partial\Omega$.

V. Experimental Evaluation

A. *Training Procedure*

1) Training Setting: To achieve realistic physical simulation and contact modeling, we developed our framework within the 3D Gazebo simulation engine based on an open-source simulation platform [20]. The implementation leverages the SAC framework from the Stable-Baselines3 RL library [22], enhanced with our proposed asymmetric SAC network architecture and C-VAE. The main hyperparameters of SAC and VAE include the entropy coefficient (1.5), target network update rate (0.01), discount factor (0.99), KL divergence weight (1.0), and gradient clipping threshold (0.5). All neural networks employ LeakyReLU activation functions and are optimized using the Adam optimizer with a learning rate of 3×10^{-4}. The training was conducted on a PC with an Intel Core i7-11700F CPU and NVIDIA RTX 4090 GPU.

2) Terrain Curriculum: The terrain curriculum uses standardized complex terrains from the RoboCup Rescue League, including Stair, Stepfield, Terrace, and Ramp, to facilitate autonomous obstacle-crossing in challenging rescue scenarios. The frequency of terrain occurrence is modulated by difficulty, with higher frequencies corresponding to increased complexity. The terrain curriculum parameters are set as shown in Fig. 3 and Tab. I.

3) Domain Randomization: To enhance the generalizability of obstacle-crossing strategies, we implement systematic randomization in the proprioceptive and exteroceptive perception domains, including random initialization of the robot's flippers and starting positions and injecting Gaussian noise during obstacle-crossing. Specifically, perturbations are applied to the robot's orientation, linear velocity, flipper motions, and terrain perception with the $\mathcal{N}(0, 0.1^2)$ rad, $\mathcal{N}(0, 0.05^2)$ m/s, $\mathcal{N}(0, 0.1^2)$ rad and $\mathcal{N}(0, 0.08^2)$ m, respectively, simulating real-world sensor uncertainties. Furthermore, as shown in Tab. I, terrain parameters such as types, lengths, heights, orientations, and numbers are also randomized by uniform distribution during training.

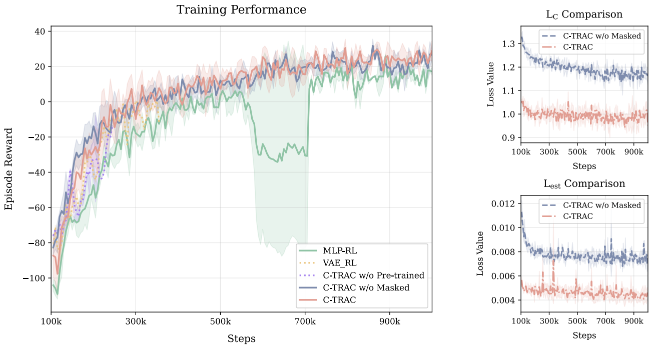

Fig. 4: The training curves schematic diagram of episode reward and contact points estimation loss. The curves and shaded regions indicate the mean and standard deviation of the reward and loss. All methods were trained three times under identical conditions, including the same terrain curriculum, noise intensity, reward function, and fixed initial random seed.

B. *Ablation Verification*

We conduct the following ablation experiments on the proposed C-VAE module:

- Baseline (MLP-RL): An asymmetric SAC policy network that replaces our C-VAE mechanism with an MLP of the same size without estimated contact states or privileged contact state inputs.
- VAE-RL: The baseline is augmented with a conventional VAE network (difference from C-VAE) to extract latent features, but still excludes estimated or privileged contact states.
- C-TRAC w/o Pretrained: Our proposed method, but the training process lacks the two-stage pre-training framework.
- C-TRAC w/o Masked: Our proposed method, but the contact loss function lacks the dynamic masking mechanism.
- C-TRAC: The complete method.

The experimental results shown in Fig. 4 demonstrate that MLP-RL, which replaces the C-VAE with an MLP, exhibits pronounced reward oscillations and lower convergence performance than C-TRAC, confirming the C-VAE module's essential role in stabilizing RL training. Direct integration of a standard VAE architecture unexpectedly led to consistent mid-training failures across multiple trials, necessitating the proposed two-stage pretraining framework to ensure reliable convergence. Analysis reveals this instability arises from volatile gradient dynamics during the VAE's denoising process and compounding errors in obstacle-contact estimation. The two-stage framework addresses these challenges by pretraining C-VAE to reduce encoding error. Incorporating dynamic masked weighting further reduces contact estimation loss. Based on these findings, the C-TRAC method with dynamic masking is adopted as the baseline for subsequent obstacle negotiation experiments due to its balanced performance and robustness.

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on January 16,2026 at 15:48:09 UTC from IEEE Xplore. Restrictions apply.

TABLE II: Quantitative evaluation results of different algorithms on various tasks in the simulation. The bold number indicates the best performance, and the underlined number indicates the second-best performance.

Metrics	45° stair-ascent				45° stair-descent				0.3 m unilateral step			terrace(ordinary)		terrace(oblique)	
	Ours	FTR	FTRm	Chen's	Ours	FTR	FTRm	Chen's	Ours	FTR	FTRm	Ours	Chen's	Ours	Chen's
max speed [m/s]↑	**0.25**	**0.25**	**0.25**	0.15	**0.25**	0.1	0.1	0.15	**0.25**	0.2	0.2	**0.25**	0.15	**0.25**	0.15
t_{cost} [s]↓	**18.341**	19.845	21.051	29.960	**16.052**	28.155	29.490	25.630	23.320	**22.420**	27.560	**29.395**	34.160	**33.474**	37.240
APS [rad/s]↓	0.126	0.150	0.220	**0.123**	0.124	0.125	0.134	**0.120**	**0.071**	0.112	0.109	0.213	**0.176**	**0.231**	0.249
MAS [rad/s]↓	1.960	3.269	4.278	**1.826**	**1.654**	4.793	6.176	2.592	**2.303**	5.263	6.006	**2.177**	2.824	**2.292**	4.368

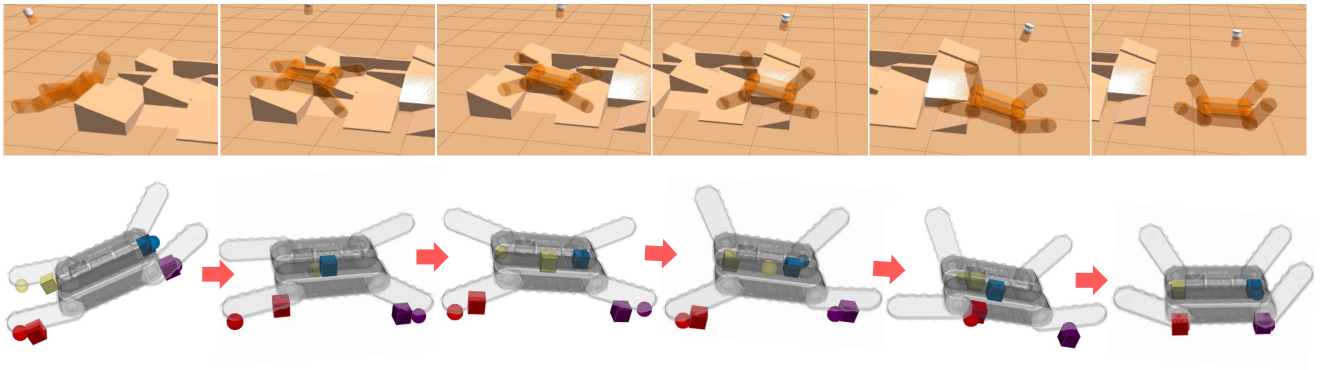

Fig. 5: The process of the robot passing through random ramps and the corresponding contact estimation results. The squares represent the estimated contact points, the dots represent the ground truth of contact points, and the four colors represent the four areas of the contact. In the simulation, we adopted a low-cost method to realize the perception of the robot's surroundings and subjacent terrain by installing a LiDAR on the top of the robot and hiding the visual mark of URDF (Unified Robot Description Format).

C. Evaluation of the Performance in Simulation

Following the metrics established in prior works [4], [9], [23], which have been proven effective for evaluating obstacle-crossing performance. We adopt the following metrics, including:

- Average Pitch/Roll Swing (APS/ARS): the average of the absolute pitch/roll angular velocity.
- Maximum of the Angular Swing (MAS): the sum of the maximum of the orientation angular velocity.
- t_{cost}: the cost time of obstacle-crossing process.

To validate the superiority of our proposed obstacle-crossing algorithm, we compare it with the following SOTA methods in simulation across diverse terrains:

- Chen's method [4]: Method of pose prediction combined with dynamic programming.
- FTR [9]: The SAC strategy of a specific terrain.
- FTRm [9]: The SAC strategy variant trained on mixed terrains.

1) Obstacle-crossing Performance: The obstacle-crossing performance comparison in Tab. II is tested across four terrains through 10 repeated trials: 45° ascending/descending stairs, a 0.3m unilateral step, and ordinary/oblique terrace environments. Our algorithm performs with superior speed, achieving the highest max speed and shortest task completion time t_{cost} while maintaining stable obstacle crossing. It also effectively balances agility against stability with near-optimal APS and MAS metrics.

The 45° staircases serve as a demanding vertical obstacle challenge, where our method achieves near-optimal stability metrics [4] at the fastest obstacle-crossing speed on stairs. It significantly reduces traversal time while maintaining high stability. In V-shaped valleys, where robots are prone to jamming, our approach achieves lower peak shaking (MAS) than conventional symmetric-modeling methods [4] on ordinary terraces, albeit with slightly reduced orientation stability (APS). Notably, it outperforms competitors in asymmetric terrains, demonstrating strong adaptability to irregular configurations. These results highlight our algorithm's ability to maintain stability while prioritizing speed in rugged environments.

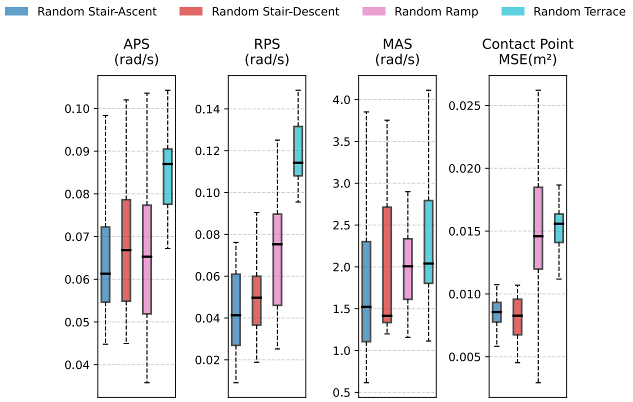

Fig. 6: The boxline schematic diagram of quantitative evaluation results on various random terrains in the simulation.

2) Contact Estimation Performance: As visualized in the obstacle-crossing process on the random ramp terrain Fig. 5, C-TRAC achieves smooth crossing while estimating contact states in unstructured scenarios. The flipper contact estimates exhibit high precision due to localized point-contact dynam-

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on January 16,2026 at 15:48:09 UTC from IEEE Xplore. Restrictions apply.

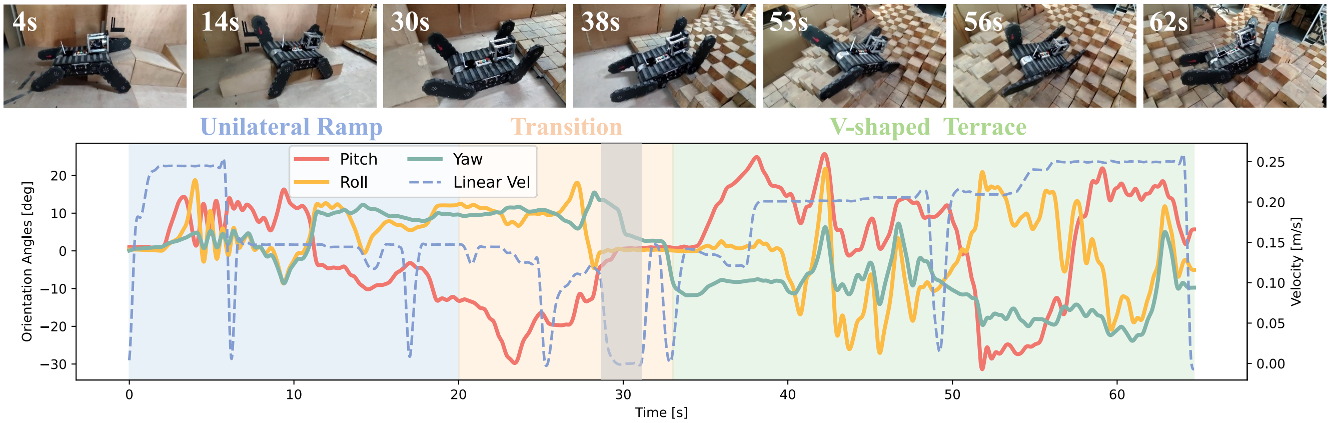

Fig. 7: The process of the robot passing through the hybrid unstructured terrain and the corresponding orientation angles and linear velocity curves.

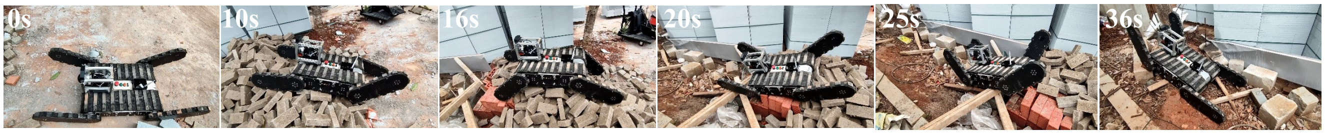

Fig. 8: The process of the robot passing through the stack of bricks outdoors.

ics. In contrast, the chassis estimates show larger errors primarily stemming from the inherent complexity of modeling surface-level interactions compared to discrete point contacts. Despite these challenges, the overall estimation accuracy remains operationally viable, excelling in critical flipper positioning while maintaining acceptable error bounds for chassis surface contacts.

The obstacle-crossing and contact estimation performance of C-TRAC is evaluated on the terrain generated by four programs, and the quantitative results are shown in Fig. 6. The average value of the stability metric under each type of terrain is low. The result with a large standard deviation appears in the more difficult terrain (Ramp or Terrace with random yaw angle distribution). The changing trend of the metric is consistent with the progress of terrain difficulty. Each terrain type is generated using a fixed random seed to ensure robustness testing, thus achieving repeatable but diverse configurations.

D. *Evaluation of the Performance in Real World*

We conducted sim-to-real experiments in challenging hybrid terrains and outdoor construction site scenarios to verify the proposed algorithm's robustness and generalization. To ensure the accuracy of both the terrain and the robot pose information, we employed a surroundings terrain map created using the reliable ALOAM map building algorithm [24], which is gradually constructed during its traversal as the robot progresses.

1) Obstacle-crossing in Hybrid Terrain: The 7.4m hybrid terrain integrates a 0.4m high unilateral ramp and a V-shaped terrace to test the generalization capability of a single traversal policy across multiple unstructured scenarios. The obstacle-crossing performance, corresponding orientation angles, and linear velocity curves are shown in Fig. 7. The algorithm controls the two right-side flippers to stabilize the robot during the (0 $\sim$ 20s) while traversing the unilateral ramp, followed by retracting the flippers for transition between (21 $\sim$ 33s). Upon entering the V-shaped valley, the algorithm coordinates flipper retraction with increased linear velocity to ensure unimpeded passage. The linear velocity adaptively modulates between higher-speed locomotion on flat regions and lower-speed or stationary states during potential instability. It should be noted that the robot's yaw is highly susceptible to influence when crossing single-sided obstacles, and the grey marker indicates manual intervention to adjust the yaw.

TABLE III: Quantitative evaluation results of our algorithm on various tasks in the real world. The number in brackets indicates the standard deviation of metrics.

Metrics	Unilateral Ramp	V-shaped Terrace	Hybrid Terrain	Stack of Bricks
t_{cost} [s]↓	31.981(1.315)	36.470(3.992)	61.120(3.129)	32.127(3.646)
APS [rad/s]↓	0.138(0.021)	0.137(0.010)	0.140(0.004)	0.123(0.012)
ARS [rad/s]↓	0.154(0.025)	0.152(0.020)	0.177(0.015)	0.152(0.038)
MAS [rad/s]↓	2.982(0.441)	2.289(0.238)	2.493(0.416)	2.358(0.821)

Quantitative metrics for unstructured terrain traversal are summarized in Tab. III. The slightly lower Average Postural Stability (APS) and Angular Rate Stability (ARS) in hybrid terrain compared to standalone unilateral ramp or V-shaped terrace tests reflect increased difficulty due to compounded yaw disturbances. The Maximum Amplitude of Shaking (MAS) falls between these two scenarios, indicating the peak oscillations primarily originate from unilateral obstacles. Overall, the algorithm satisfies mission requirements for stability and agility in hybrid terrain.

2) Obstacle-crossing in Building Site: Outdoor zero-shot transfer scenarios present substantial challenges for reinforcement learning algorithms. Field tests were performed in authentic construction sites featuring two representative

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on January 16,2026 at 15:48:09 UTC from IEEE Xplore. Restrictions apply.

scenarios: a stack of bricks and debris-filled ruins to evaluate obstacle-crossing capabilities in post-disaster environments.

Quantitative and qualitative results from brick stack obstacle-crossing trials are illustrated in the Tab. III and Fig. 8 respectively, while qualitative demonstrations in ruin environments are provided in the supplementary video. The proposed algorithm achieves rapid and stable traversal across these unstructured terrains, partially demonstrating its ability to bridge the sim-to-real gap. This success confirms the method's strong generalization capacity and robustness in real-world applications. Notably, the flipper locomotion strategy effectively adapts to unmodeled terrain properties such as variable brick surface friction and irregular debris geometry, maintaining consistent performance without additional training.

VI. Conclusion

This paper presents the C-TRAC framework, a novel approach for enabling articulated tracked robots to autonomously traverse unstructured terrains by addressing sim-to-real transfer challenges rooted in dynamic contact uncertainties by integrating a two-stage training paradigm that pre-trains an implicit contact-aware VAE to model latent contact dynamics and joint optimization with an asymmetric SAC policy. Our method achieves robust adaptation to real-world contact variations. C-VAE's denoising multi-frame coding mechanism reduces the inaccuracy of contact estimation. At the same time, the asymmetric Actor-critical architecture ensures that SAC can experience richer contact information and terrain environment from the simulation, thus improving sim-to-real capability.

Experiments demonstrate superior performance over SOTA methods, significantly improving traversal speed, stability, and zero-shot sim-to-real generalization. The success of the framework in real-world deployment highlights its potential for disaster response and exploration applications. At the same time, future work will extend it to dynamic environments and multimodal sensing for enhanced contact resolution.

References

[1] J. Liu, Y. Wang, B. Li, and S. Ma, "Current research, key performances and future development of search and rescue robots," *Chinese Journal of Mechanical Engineering*, vol. 42, no. 12, pp. 1–12, 2006.

[2] K. Nagatani, A. Yamasaki, K. Yoshida, T. Yoshida, and E. Koyanagi, "Semi-autonomous traversal on uneven terrain for a tracked vehicle using autonomous control of active flippers," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2008, pp. 2667–2672.

[3] Y. Okada, K. Nagatani, K. Yoshida, S. Tadokoro, T. Yoshida, and E. Koyanagi, "Shared autonomy system for tracked vehicles on rough terrain based on continuous three-dimensional terrain scanning," *Journal of Field Robotics (JFR)*, vol. 28, no. 6, pp. 875–893, 2011.

[4] B. Chen, K. Huang, H. Pan, H. Ren, X. Chen, J. Xiao, W. Wu, and H. Lu, "Geometry-based flipper motion planning for articulated tracked robots traversing rough terrain in real-time," *Journal of Field Robotics (JFR)*, vol. 40, no. 8, pp. 2010–2029, 2023.

[5] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[6] A. Mitriakov, P. Papadakis, J. Kerdreux, and S. Garlatti, "Reinforcement learning based, staircase negotiation learning: Simulation and transfer to reality for articulated tracked robots," *IEEE Robotics and Automation Magazine (RAM)*, vol. 28, no. 4, pp. 10–20, 2021.

[7] M. Pecka, V. *S*alanský, K. Zimmermann, and T. Svoboda, "Autonomous flipper control with safety constraints," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2016, pp. 2889–2894.

[8] K. Zimmermann, P. Zuzánek, M. Reinstein, T. Pet*rí*cek, and V. Hlavá*c*, "Adaptive traversability of partially occluded obstacles," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2015, pp. 3959–3964.

[9] H. Zhang, J. Ren, J. Xiao, H. Pan, H. Lu, and X. Xu, "Ftr-bench: Benchmarking deep reinforcement learning for flipper-track robot control," *Journal of Field Robotics (JFR)*, 2025.

[10] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2018, pp. 3803–3810.

[11] J. Gottlieb, P.-Y. Oudeyer, M. Lopes, and A. Baranes, "Information-seeking, curiosity, and attention: Computational and neural mechanisms," *Trends in Cognitive Sciences*, vol. 17, no. 11, pp. 585 – 593, 2013.

[12] H. Pan, X. Chen, J. Ren, B. Chen, K. Huang, H. Zhang, and H. Lu, "Deep reinforcement learning for flipper control of tracked robots in urban rescuing environments," *Remote Sensing*, vol. 15, no. 18, 2023.

[13] 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 *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017, pp. 23–30.

[14] I. M. Aswin Nahrendra, B. Yu, and H. Myung, "Dreamwaq: Learning robust quadrupedal locomotion with implicit terrain imagination via deep reinforcement learning," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023, pp. 5078–5084.

[15] G. Ji, J. Mun, H. Kim, and J. Hwangbo, "Concurrent training of a control policy and a state estimator for dynamic and robust legged locomotion," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 2, pp. 4630–4637, 2022.

[16] S. Lyu, X. Lang, H. Zhao, H. Zhang, P. Ding, and D. Wang, "Rl2ac: Reinforcement learning-based rapid online adaptive control for legged robot robust locomotion," in *Proc. of Robotics: Science and Systems (RSS)*, 2024.

[17] X. Gu, Y.-J. Wang, X. Zhu, C. Shi, Y. Guo, Y. Liu, and J. Chen, "Advancing humanoid locomotion: Mastering challenging terrains with denoising world model learning," 2024.

[18] Y. W. Li, S. R. Ge, X. Wang, and H. B. Wang, "Steps and stairs-climbing capability analysis of six-tracks robot with four swing arms," *Applied Mechanics and Materials*, vol. 397, pp. 1459–1468, 2013.

[19] K. Ohno, S. Morimura, S. Tadokoro, E. Koyanagi, and T. Yoshida, "Semi-autonomous control system of rescue crawler robot having flippers for getting over unknown-steps," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2007, pp. 3012–3018.

[20] A. Mitriakov, P. Papadakis, and S. Garlatti, "An open-source software framework for reinforcement learning-based control of tracked robots in simulated indoor environments," *Advanced Robotics*, vol. 36, no. 11, pp. 519–532, 2022.

[21] A. Y. Ng, D. Harada, and S. Russell, "Policy invariance under reward transformations: Theory and application to reward shaping," in *Proc. of the Int. Conf. on Machine Learning (ICML)*, vol. 99, 1999, pp. 278–287.

[22] 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.

[23] T. Azayev and K. Zimmermann, "Autonomous state-based flipper control for articulated tracked robots in urban environments," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 3, pp. 7794–7801, 2022.

[24] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time." in *Proc. of Robotics: Science and Systems (RSS)*, vol. 2, no. 9. Berkeley, CA, 2014, pp. 1–9.

Authorized licensed use limited to: National Univ of Defense Tech. Downloaded on January 16,2026 at 15:48:09 UTC from IEEE Xplore. Restrictions apply.