Research on Whole-Body Coordinated Motion of Humanoid Robots Based on LSTM-Integrated Reinforcement Learning

Tianyu Yuan[1], Chaoyi Dong[1,2,3,*], Ge Tai[1], Shuai Xiang[1], Haoda Yan[1],
Zhifeng Kong[1], Chenzhe Zhang[1] and Xiaoyan Chen[1,2,3]

Abstract— **This paper addresses the issue of stiffness in the upper body and lack of coordination between the upper and lower body during humanoid robot walking. An improved humanoid robot reinforcement learning algorithm incorporating an LSTM framework is proposed to optimize full-body coordinated movement. Based on the Humanoid-Gym framework, a novel reward mechanism is designed, taking into account the detailed evaluation of arm movement and the collaborative control between the arms and thighs. The reinforcement learning model adopts an Actor-Critic architecture, integrating the LSTM framework into the network to enhance the feature extraction and dynamic modeling capabilities. Finally, experiments were conducted using the Hi ROBOT humanoid platform to validate the proposed model. The proposed LSTM network algorithm is compared with the original network, GRU, CNN, and other networks, demonstrating the superiority of the model. Compared with other networks, the performance improves by approximately 3.4% in terms of reward metric, and the model reaches the performance level of the original network after 40k training steps, as opposed to 60k steps. It also maintains a fast convergence rate. Additionally, the optimized algorithm results in better gait arm swing and leg coordination, with smoother and more coordinated movement, closest to the human walking pattern.**

I. INTRODUCTION

In recent years, with the rapid development of artificial intelligence and robotics, the potential applications of humanoid robots in fields such as services, healthcare, education, and rescue have garnered widespread attention. However, achieving full-body coordinated movement in humanoid robots remains a critical technical challenge. During humanoid robot walking, insufficient coordination between the upper and lower body movements often leads to issues such as jerky motion and instability.

Reinforcement Learning (RL), as a data-driven optimization method, has shown great potential in robot motion control. The Proximal Policy Optimization (PPO) algorithm proposed by Schulman et al. excels in policy optimization [1], while Heess et al. demonstrated the use of reinforcement learning to learn motion behaviors in complex environments [2], providing theoretical support for humanoid robot coordination. Additionally, Yu et al. explored symmetrical and low-energy motion strategies for humanoid robots based on deep reinforcement learning [3], further advancing the field. However, traditional reinforcement learning methods primarily focus on single tasks or local optimization, making them insufficient for addressing the complex needs of full-body humanoid robot coordination.

To overcome this challenge, researchers in recent years have combined deep learning models with reinforcement learning, proposing various improved methods. This paper addresses the issue of poor coordination between the upper and lower body in humanoid robots by proposing an improved reinforcement learning algorithm. A novel reward mechanism function is designed, and the LSTM network is integrated into the traditional Actor-Critic architecture to significantly enhance the ability to model complex dynamic environments. The following sections will provide a detailed introduction to the methods used, related work, experimental design, and research conclusions.

II. RELATED WORK

A. Actor-Critic Framework and PPO Algorithm

The Actor-Critic architecture in Reinforcement Learning is a framework that combines policy learning (Actor) and value estimation (Critic). The Actor-Critic method generates action policies through the policy network, while the value network evaluates the value function of the current state. By interacting these two components, it improves the stability and efficiency of policy optimization.

In the Actor-Critic framework, PPO algorithm has gained significant attention due to its stability and efficiency in policy updates. PPO was proposed by Schulman et al. in 2017 [1], introducing a clipped loss function to restrict the magnitude of policy updates, thereby avoiding performance fluctuations caused by excessively large policy gradient updates. This method is particularly suitable for handling continuous action spaces, making it widely applied in robot motion control tasks.

In recent years, PPO has been extensively used in reinforcement learning tasks for humanoid robots, and Heess et al.'s research demonstrates PPO's performance in complex multi-task environments [3]. However, traditional PPO algorithms still have limitations in optimizing full-body coordinated movements, such as limited capability in fully extracting dynamic features of the whole body. Therefore, this paper proposes integrating LSTM into the PPO Actor-Critic architecture to further enhance the model's ability to learn complex motion patterns.

[1] College of Electric Power, Inner Mongolia University of Technology, Hohhot 010080, China.

[2] Intelligent Energy Technology and Equipment Engineering Research Centre of Colleges and Universities in Inner Mongolia Autonomous Region, Hohhot 010080, China.

[3] Engineering Research Center of Large Energy Storage Technology, Ministry of Education, Hohhot 010051, China.

[*]Corresponding author: Chaoyi Dong, email:dongchaoyi@imut.edu.con

B. Humanoid-Gym Framework

Humanoid-Gym is a reinforcement learning simulation environment specifically designed for humanoid robot motion control. Built on the Isaac-Gym framework, it provides flexible humanoid robot modeling, dynamic simulation, and reinforcement learning environment integration. By simulating the motion characteristics of the upper and lower limbs of a real robot, Humanoid-Gym can efficiently generate training data and support complex optimization in continuous action spaces [4]. Compared to traditional simulation platforms, Humanoid-Gym offers more customizable features, especially the flexible design of the reward mechanism, which is crucial for optimizing full-body coordinated motion in this study.

This paper selects Humanoid-Gym as the simulation environment due to its balanced efficiency and flexibility, support for reward mechanisms, and strong community integration. By designing innovative rewards and combining them with an improved Actor-Critic architecture on Humanoid-Gym, it effectively enhances humanoid robots' full-body coordination, validating the method's feasibility and superiority.

C. LSTM

In reinforcement learning tasks, modeling temporal features is key to achieving full-body coordinated motion. Long Short-Term Memory (LSTM) networks [5] are widely used in dynamic behavior prediction due to their ability to model long-term dependencies [6]. Compared to traditional RNN structures, LSTM effectively addresses the vanishing gradient problem by introducing gating mechanisms, enabling more accurate capture of temporal sequence features in humanoid robot motion.

D. Design of the Reward Mechanism

In reinforcement learning, the design of the reward mechanism is crucial for achieving full-body coordinated motion in humanoid robots [7]. A well-designed reward function can guide the robot to learn the desired behavior patterns and promote coordination between the upper and lower body. For example, recent research has proposed a training method based on fractal noise [8], simplifying the design of the reward function and successfully achieving dynamic motion for humanoid robots on complex terrains. Additionally, a research team from Carnegie Mellon University developed a real-time human-to-humanoid robot full-body teleoperation system, using a reinforcement learning framework to enable humanoid robots to mimic human movements [9]. Therefore, designing an effective reward mechanism is essential for enhancing the motion coordination and adaptability of humanoid robots in complex environments.

When designing the reward mechanism, various factors need to be considered to ensure the robot learns the desired behavior. First, the reward function should reflect the task objectives, such as stability, speed, and energy consumption during walking. Second, the reward function should include penalties for undesired behaviors, such as falling or deviating from the planned path. Additionally, the reward function's design should consider the coordination between the robot's upper and lower body, encouraging natural human-like movement patterns [10].

However, designing an effective reward function is not easy. An overly complex reward function may lead to instability during the learning process, while an overly simple reward function may fail to capture complex motion patterns. Therefore, researchers need to strike a balance between task requirements and learning efficiency [11].

In conclusion, the design of the reward mechanism plays a critical role in reinforcement learning-driven humanoid robot full-body coordination. Through reasonable reward function design, robots can achieve more natural and efficient movement in complex environments.

III. METHODS

A. Modifications to the Actor-Critic Network

In traditional reinforcement learning, the Actor-Critic architecture typically uses simple linear layers or fully connected layers to construct the policy network (Actor) and the value network (Critic). While these networks can handle some basic tasks, they still have certain limitations. Specifically, traditional network structures struggle to effectively capture temporal and global features, leading to poor performance in full-body coordinated motion optimization.

To overcome these limitations, this paper introduces the following improvements to the Actor-Critic network:

Introduction of LSTM Modules,The motion of humanoid robots is a typical time-series problem, and the coordination between the upper and lower body requires effective modeling of temporal features. LSTM networks are a special type of recurrent neural network (RNN) that can effectively handle long-term dependency problems and capture temporal features in sequence data. LSTM modules are introduced into both the Actor and Critic networks. Specifically, input data is first processed through the LSTM module to extract temporal features, which are then passed on to the subsequent fully connected layers.

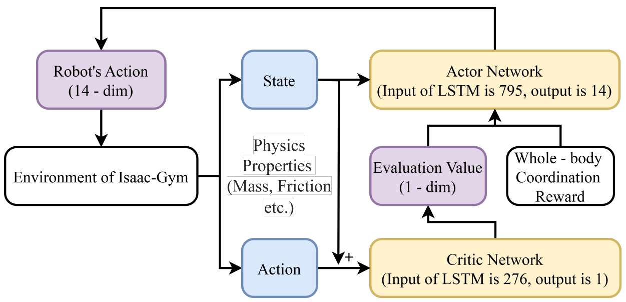

Fig. 1. Joint framework diagram of Long Short-Term Memory (LSTM) and Reinforcement Learning)

B. Design of the Reward Mechanism

In this study, we aim to design a comprehensive reward mechanism to promote the coordinated motion of both the upper and lower body of the robot. This mechanism includes a fine-grained evaluation of arm movements as well as the coordinated control between the arms and thighs, with the goal of guiding the robot to achieve smooth, efficient, and coordinated full-body motion through appropriate reward signals. 1. Arm Movement Reward Design:

TABLE I
EXPERIMENTAL PARAMETERS

No.	Symbol	Meaning	Unit
1	$v_{\text{arm}}, v_{L,\text{arm}}, v_{R,\text{arm}}$	Arm linear velocity	m/s
2	$\theta_{L,\text{arm}}, \theta_{R,\text{arm}}$	Arm position	degrees
3	$\theta_{L,\text{leg}}, \theta_{R,\text{leg}}$	Thigh position	degrees
4	$\omega_{L,\text{arm}}, \omega_{R,\text{arm}}$	Arm angular velocity	rad/s
5	$\omega_{L,\text{leg}}, \omega_{R,\text{leg}}$	Thigh angular velocity	rad/s
6	R	Represents the reward value	-
7	λ	Weight param	-
8	α, β	Adjustment params	-

We first design a general reward function for arm movements, which focuses on accurately evaluating the quality of arm motion in a specific direction. The speed thresholds $v_{\min}$, $v_{\max}$ and penalty factor C are parameters, and v_{arm} represents the arm's speed.

(1) Activation Function Design:

We introduce a smooth activation function to construct $Active$, which is used to measure whether the arm's speed falls within a reasonable range.

$$Active(v) = \sigma \cdot [\alpha_1(v - v_{\min})] \times \{1 - \sigma[\beta_1(v - v_{\max})]\} \quad (1)$$

Where σ represents the sigmoid function, which maps the arm's speed to the range [0, 1]. When the speed is within the threshold range, the activation value approaches 1; otherwise,it approaches 0.

(2) Speed Perturbation and Constraints:

Considering the disturbances and uncertainties in the real-world environment, we introduce a perturbation term into the reward function:

$$R_{arm} = Active(v_{arm}) \cdot (1 - C|v_{arm} - v_{opt}|) + \gamma \cdot \xi \quad (2)$$

Where v_{opt} is the desired speed, C is the penalty factor for deviations from the ideal speed, and γ controls the strength of the perturbation, determining the impact of random factors on the reward. $\xi \sim N(0, \sigma^2)$.

2. Full-Body Coordination Reward Design:

To achieve coordinated motion of the upper and lower limbs, we design the $R_{total\_symmetry}$ function, which comprehensively evaluates the synchronization and consistency of the arms and thighs.

(1) Symmetry Penalties:

- Speed Symmetry Penalty: Constrains the motion speed of the left and right arms, ensuring symmetry.

$$R_{speed} = -\lambda_1 \|v_{L,arm} - v_{R,arm}\| \quad (3)$$

- Position Symmetry Penalty: Ensures that the relative positions of the left and right arms are symmetric in space.

$$R_{position} = -\lambda_2 \|\theta_{L,arm} - \theta_{R,arm}\| \quad (4)$$

- Inverse Motion Penalty: If the left and right arms move in the same direction, a penalty is applied to encourage a reasonable swinging pattern.

$$R_{opposite} = -\lambda_3 \|v_{L,arm} + v_{R,arm}\| \quad (5)$$

- Thigh Symmetry Penalty: Constrains the angles of the left and right thighs, ensuring coordination during the gait process.

$$R_{thigh} = -\lambda_4 |\theta_{L,leg} - \theta_{R,leg}| \quad (6)$$

(2) Directional Reward Introduction:

We calculate the reverse dot product of the left and right arm velocities to encourage symmetric motion.

$$R_{direction} = \lambda_5 \cdot [v_{L,arm} \cdot (-v_{R,arm})] \quad (7)$$

(3) Gait Coordination Constraints (Arm and Thigh Interaction):

In a natural walking pattern, when the left arm swings, the right leg swings forward; when the right arm swings, the left leg swings forward, forming a cross-coordination relationship between the arm and the opposite leg. To model this, we introduce the following constraints:

- Arm-Thigh Angle Coordination: Encourages the arm swing angle to match the opposite thigh's angle.

$$R_{h1} = \lambda_6 \left(|\theta_{L,arm} - \alpha_2 \theta_{R,leg}| + |\theta_{R,arm} - \alpha_3 \theta_{L,leg}|\right) \quad (8)$$

- Arm-Thigh Angular Velocity Direction Consistency: Encourages the angular velocity direction of the arm swing to align with that of the opposite thigh.

$$R_{h2} = \lambda_7 \left[\omega_{L,arm} \cdot (\beta_2 \omega_{R,leg}) + \omega_{R,arm} \cdot (\beta_3 \omega_{L,leg})\right] \quad (9)$$

IV. EXPERIMENTS

This experiment was conducted on an Ubuntu 20.04 system equipped with an Nvidia 2080Ti GPU (11GB GDDR6 memory). The robot model used is an existing one in our laboratory, based on the Hi robot from HIGH TORQUE, a leading manufacturer of high-performance humanoid robots (Figure 1 illustrates the robot model). The simulation environment is based on Nvidia Isaac-Gym, and reinforcement learning training is carried out using Humanoid-Gym. The experiment employs the Proximal Policy Optimization (PPO) algorithm, combined with the Actor-Critic architecture for

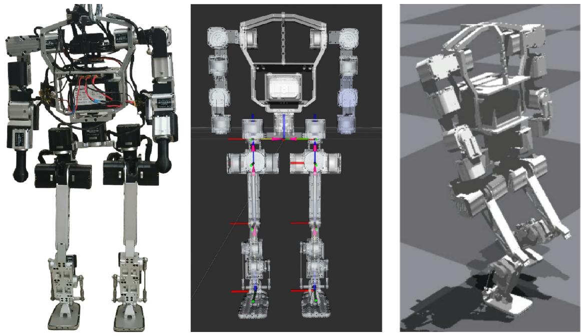

Fig. 2. Robot Model (from left to right: the physical robot, URDF and simulation environment model)

TABLE II

Experimental parameters

Optimized parameters for training	Values
Number of Environments	4096
Number Training Epochs	2
Batch size	4096
Discount Factor	0.994
GAE discount factor	0.95
Entropy Regularization Coefficient	0.001
Learning rate	1×10^{-5}
Hidden Units/Units in the LSTM Layer	128
Hidden Units/Units in the GRU Layer	128
Filter Size	3
The input dimension of the actor	795
The output dimension of the actor	14
The input dimension of the critic	276
The output dimension of the critic	1

TABLE III

Model Comparison

Model	Original	GRU	CNN1D	LSTM
Reward	172.67881	60.75419	42.93547	178.73331
Episode	2359.635	1690.709	1131.572	2399.2
FPS	71584.6	13574.8	70229.4	70321.3
Time	3.11445	6.00941	2.99472	3.09412
Value	0.0011296	0.0060712	0.0030067	-0.0000735
Surrogate	0.0008810	0.0023006	0.0031957	0.00043785

training. Based on this, we designed a series of comparative experiments to verify the impact of different network structures on the humanoid robot's motion coordination.

A. *Comparison Experiments*

In this implementation, the parameters used are listed in Table 2.

$$Q(s_t, a_t) \leftarrow (1-\alpha)Q(s_t, a_t)+\alpha r_{t+1}+\alpha\gamma \max_{a} Q(s_{t+1}, a) \quad (10)$$

In reinforcement learning, the Q - learning update rule (Equation (10)) is: Here, s_t and a_t are the state and action at time step t. The learning rate α (Table 2, 1×10^{-5}) and discount factor γ (Table 2, 0.994) regulate update extent and future reward importance, respectively. r_{t+1} is the immediate reward, and $\max_a Q(s_{t+1}, a)$ estimates the optimal future action's Q - value. This rule iteratively optimizes $Q(s_t, a_t)$ to find the optimal policy.

To evaluate the effectiveness of the proposed method, we tested the following four models:

- Original (Baseline Model): Only a fully connected layer (MLP) is used as the policy and value network.
- GRU: The LSTM is replaced with a GRU to compare its performance in time series modeling.
- CNN: A CNN is used to extract temporal features, and the policy is output through a fully connected layer.
- LSTM (Proposed Method): An LSTM layer is introduced into the Actor-Critic framework to enhance the modeling capability of temporal features.

B. *Result Analysis*

The parameters in Table 3 represent the peak values of the curves shown in Figures 2 to 4, where the data in the first and second rows correspond to the left and right sides of Figure 2, respectively; the data in the third and fourth rows correspond to the left and right sides of Figure 3, respectively; and the data in the fifth and sixth rows correspond to the left and right sides of Figure 4, respectively.

As illustrated in Figure 2, LSTM surpasses other models at 40k training steps, reaching a peak reward of 178.73—3.39% higher than the Original model (172.68). In contrast, GRU and CNN1D peak at 60.75 and 42.94, respectively, 66.04% and 76.03% below LSTM's performance, with GRU performing poorly in early training. The Original model requires over 60k steps to match LSTM's 40k-step performance, highlighting LSTM's faster feature capture and optimization.

Figure 3 shows that LSTM (3.09412s) and the Original model (3.11445s) have comparable training times. GRU takes 6.00941 seconds—1.94 times longer than LSTM—due to higher computational complexity. CNN1D's 2.99472s training time is marginally shorter but its low reward reflects poor time-series modeling. Regarding FPS, the Original model (71584.6) slightly outperforms LSTM (70321.3), while GRU's 13574.8 is 80.6% lower, indicating inefficiency.

Figure 4 reveals that LSTM and the Original models maintain low loss values, with LSTM's -0.0000735 significantly lower than GRU's 0.0060712 and CNN1D's 0.0030067. Similarly, LSTM's Surrogate value of 0.00043785 is much lower than GRU's 0.0023006 and CNN1D's 0.0031957, confirming its optimization efficiency and superiority in time-series modeling.

Overall, although LSTM has a slightly slower convergence speed compared to the Original model, it shows superior performance in reward improvement, with a peak increase of 3.39%. It also demonstrates significant advantages in training efficiency, loss minimization, and convergence speed. As the innovation point of this study, LSTM exhibits far superior time-series modeling ability compared to traditional fully connected network models.

C. *Motion Analysis*

To further analyze the robot's motion patterns under different methods, we have organized the keyframes of the motion in Figure 5. From left to right, the following observations can be made:

- LSTM:The gait of the LSTM model is the closest to human walking patterns, with good coordination between arm swinging and leg movement. The motion is smooth, and the coordination is high. The robot is also able to maintain good stability in complex environments.
- Original: The Original model has poorer robustness in its motion pattern. There is a lack of coordination between the height of the foot and the amplitude of the arm swing. The gait is relatively stiff, and the robot is prone to losing balance, especially during high-speed motion.
- GRU: The gait of the model is the least natural, with the movement rhythm of the arms and legs not being well-coordinated. This results in an abnormal overall

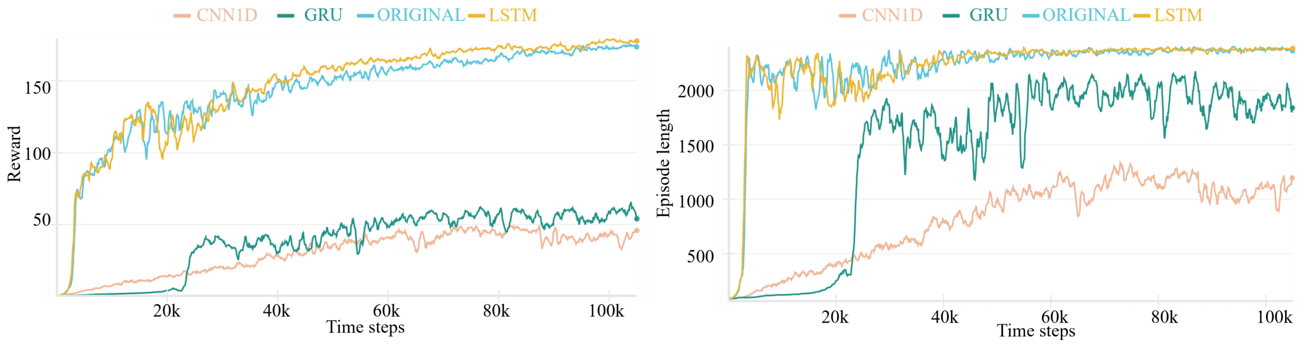

Fig. 3. Comprehensive Robot Reward Comparison Chart (Left: Total Reward, Right: Mean Episode Length)

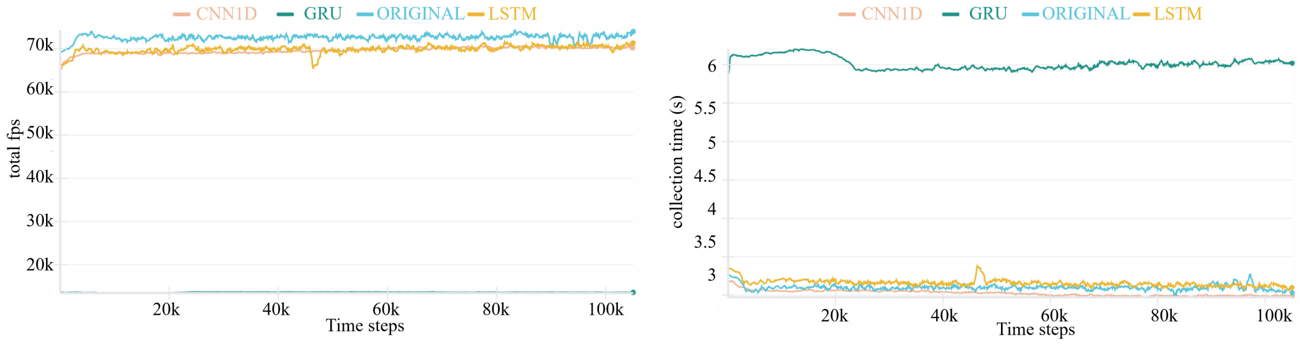

Fig. 4. Reinforcement Learning Performance Comparison Chart (Left: Total Collection Time, Right: FPS)

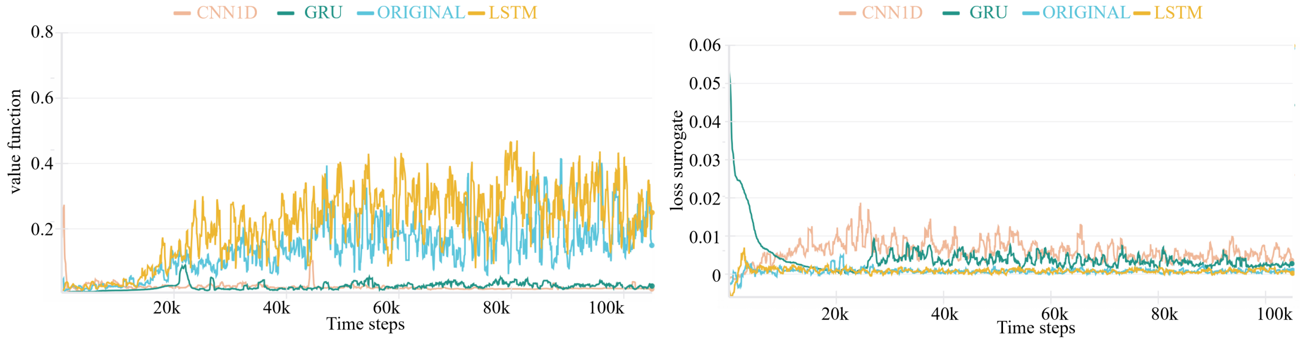

Fig. 5. Reinforcement Learning Model Loss Comparison Chart (Left: Value Function Loss, Right: Surrogate Loss)

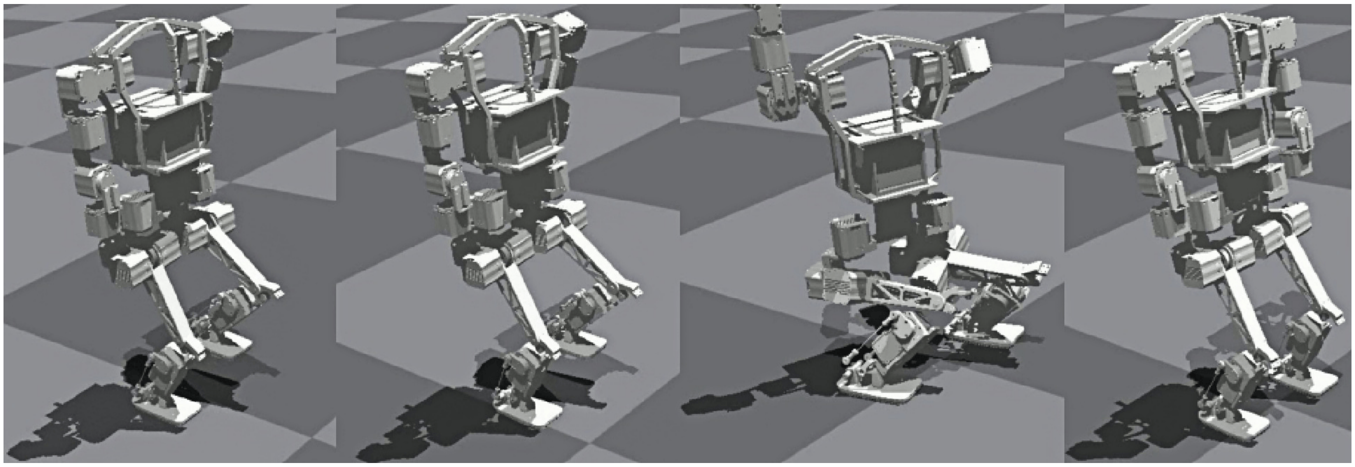

Fig. 6. The keyframe results of robot motion (from left to right: LSTM model, Original model, GRU model, and CNN1D model)

walking posture, which deviates significantly from the actual human walking pattern.

- CNN: The CNN model can maintain standing, but the overall posture is unstable, with slight wobbling during motion. The robot is prone to tilting when facing complex terrains, indicating insufficient dynamic balance control.

The newly introduced reward function has effectively improved the coordination between the robot's upper and lower limbs. By incentivizing synchronized movement between the arms and legs, the robot's overall motion has become more fluid and natural. This enhancement allows the robot to maintain better balance and stability, especially during dynamic movements and in complex environments, showcasing the effectiveness of the reward function in promoting seamless limb coordination.

V. CONCLUSIONS

This study experimentally validates the effectiveness of LSTM in reinforcement learning-driven humanoid robot motion control. Compared to traditional MLP, GRU, and CNN, LSTM demonstrates superior performance in gait coordination, convergence speed, and stability. Its strong ability to model temporal sequences allows the robot to achieve more natural movement patterns, enhancing gait balance and adaptability.

In the comparative analysis of different models, LSTM performs most stably, coordinating the robot's upper and lower limb movements effectively, and resulting in a smoother and more natural gait. The Original model, similar to LSTM in reward, lacks the ability to model temporal information, causing poor gait coordination and instability. CNN mainly focuses on local feature extraction and fails to adequately model the overall movement pattern, only maintaining a standing position without stable gait control. The GRU model has high training loss, slow convergence, and inferior gait performance compared to others, with a disorganized overall movement pattern.

Future research will optimize existing methods by integrating the LSTM structure with Lite Vision Transformer (LVT) [12] and MobileVIT's attention mechanisms [**?**]. Traditional Transformers, powerful for long-range dependencies, have high computational costs and large sizes, posing challenges to the real-time operation of humanoid robots. In contrast, LVT [12] and MobileVIT [13] are lightweight: LVT reduces complexity via optimized attention, and MobileVIT combines the advantages of CNN and Transformer for efficient, low-memory feature extraction, enabling real-time processing, enhancing long-term temporal modeling, and improving the adaptability of humanoid robots in complex environments.

The reward mechanism will be refined to optimize gait and energy control, enhancing robotic performance on various terrains and tasks. The proposed system has practical potential, such as in disaster relief where humanoid robots can navigate rubble for rescues, in healthcare for patient care, and in smart homes to support the elderly and disabled. Finally, environment modeling will be improved by fusing high-precision simulation and real-world data, strengthening the model's generalization in real-world scenarios.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (61364018 and 61863029), Inner Mongolia Natural Science Foundation (2016JQ07, 2020MS06020, and 2021MS06017), Inner Mongolia Scientific and Technological Achievements Transformation Project (CGZH2018129), Industrial Technology Innovation Program of IMAST (2023JSYD01006), and Science and Technology Plan Project of Inner Mongolia Autonomous Region (2021GG0264 and 2020GG0268).

REFERENCES

[1] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[2] Nicolas Heess, Dhruva Tb, Srinivasan Sriram, Jay Lemmon, Josh Merel, Greg Wayne, Yuval Tassa, Tom Erez, Ziyu Wang, SM Eslami, et al. Emergence of locomotion behaviours in rich environments. *arXiv preprint arXiv:1707.02286*, 2017.

[3] Wenhao Yu, Greg Turk, and C Karen Liu. Learning symmetric and low-energy locomotion. *ACM Transactions on Graphics (TOG)*, 37(4):1–12, 2018.

[4] Xinyang Gu, Yen-Jen Wang, and Jianyu Chen. Humanoid-gym: Reinforcement learning for humanoid robot with zero-shot sim2real transfer. *arXiv preprint arXiv:2404.05695*, 2024.

[5] Alex Graves and Alex Graves. Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, pages 37–45, 2012.

[6] Yunlong DING, Minchi KUANG, Jihong ZHU, Jingyu ZHU, and Zhi QIAO. Intelligent decision making and target assignment of multi-aircraft air combat based on the lstm–ppo algorithm. *Chinese Journal of Engineering*, 46(7):1179–1186, 2024.

[7] Tairan He, Zhengyi Luo, Wenli Xiao, Chong Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. Learning human-to-humanoid real-time whole-body teleoperation. *arXiv preprint arXiv:2403.04436*, 2024.

[8] Jingkang Wang, Yang Liu, and Bo Li. Reinforcement learning with perturbed rewards. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 6202–6209, 2020.

[9] Chong Zhang, Wenli Xiao, Tairan He, and Guanya Shi. Wococo: Learning whole-body humanoid control with sequential contacts. *arXiv preprint arXiv:2406.06005*, 2024.

[10] Tairan He, Wenli Xiao, Toru Lin, Zhengyi Luo, Zhenjia Xu, Zhenyu Jiang, Jan Kautz, Changliu Liu, Guanya Shi, Xiaolong Wang, et al. Hover: Versatile neural whole-body controller for humanoid robots. *arXiv preprint arXiv:2410.21229*, 2024.

[11] Yang Yang, Zheng Li, Liuliu He, and Ruilian Zhao. A systematic study of reward for reinforcement learning based continuous integration testing. *Journal of Systems and Software*, 170:110787, 2020.

[12] Chenglin Yang, Yilin Wang, Jianming Zhang, He Zhang, Zijun Wei, Zhe Lin, and Alan Yuille. Lite vision transformer with enhanced self-attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11998–12008, 2022.

[13] Sachin Mehta and Mohammad Rastegari. Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer. *arXiv preprint arXiv:2110.02178*, 2021.