EMPOWER HUMANOID MOTION CONTROL VIA LATENT DYNAMIC REPRESENTATION

Jincheng Yang
ACM Honor Class, Zhiyuan College
Shanghai Jiao Tong University
Shanghai 200240, China
modem514@sjtu.edu.cn

ABSTRACT

This paper implements and utilizes recent results for controlling humanoid robots using a structured motion representation learned through Fourier Latent Dynamics (FLD) Li et al. (2024) and a policy model trained with Proximal Policy Optimization (PPO). The experimented method addresses the limitation of pure Reinforcement Learning (RL) methods in controlling humanoid robots and leverages the benefits of learning a structured motion representation for improved performance. The pipeline includes motion retargeting from human motion clips using inverse kinematic technologies and training on multiple types of humanoid robots, including MIT humanoid Chignoli et al. (2021) and Unitree H1. The paper also presents an ablation study on the effectiveness of motion retargeting and compares the performance of the policy model on different types of humanoid robots. The results show that the proposed method achieves improved performance and demonstrates the contributions of the paper to the field of humanoid robotics automation.

1 INTRODUCTION

Humanoid robots with the ability to operate autonomously in various environments have the potential to alleviate labor shortages in factories, assist the elderly at home, and explore new planets. However, traditional controllers for humanoid robots have limitations in terms of generalization and adaptability to new environments. To address these challenges, two main approaches have been developed for obtaining robotic motion models: reinforcement learning (RL)-based methods and learning-based methods.

Reinforcement learning-based approaches involve training robots through trial and error, where they learn to perform tasks by receiving feedback from their actions in the form of rewards or penalties. These methods are particularly advantageous in dynamic and unpredictable environments, as they enable robots to learn complex behaviors autonomously without explicit programming. RL-based models can achieve high levels of performance and adaptability, as they continuously improve their policies based on interactions with the environment. However, these methods often require substantial computational resources and extensive training times to converge to an optimal policy. Additionally, the learned behaviors might be sensitive to the specific training conditions, leading to challenges in generalizing to new and unseen scenarios.

On the other hand, learning-based methods, which include supervised and unsupervised learning techniques, involve training robots using pre-collected datasets. These methods can leverage large amounts of labeled data to learn accurate motion models, often resulting in faster training times compared to RL-based methods. Learning-based approaches can provide robust solutions for specific tasks with high precision. However, they might lack the flexibility to adapt to new environments without retraining, as they rely heavily on the quality and diversity of the training data.

In this paper, we focus on the arising trend of hybrid approaches. Policy controlling over Fourier Latent Dynamics (FLD) Li et al. (2024), for example, is one of the hybrid approaches that produce high-quality results. Hybrid approaches similar to FLD typically consist of two stages: a latent space dynamic feature learning stage, followed by a RL-based stage to manipulate the robotics. The ability of this type of models relies heavily on the structure of the latent motion space. FLD Li et al.

(2024), for example, makes certain assumptions on the motion sequence to ensure the learnability of Fourier latent. In this paper, we focus on easing such limitations on the motion sequence, and hence enable the pipeline to receive a wider range of motion training data.

In conclusion, our contributions can be summarized as:

- We implement and utilize the FLD Li et al. (2024) pipeline to learn a structured motion representation, which benefits the policy model to control the motion of humanoid robots. We ultimately manage to control full-body motion for several types of humanoids (including MIT humanoid Chignoli et al. (2021) and Unitree H1) in IsaacGym simulation environment.
- We further make some refinement on the existing FLD Li et al. (2024) pipeline, resolving some limitation on the training motion sequence.
- We implement a motion retargeting pipeline that retargets motion data in joint space from human motion clips released by DeepMimic Peng et al. (2018). This pipeline helps to collect the necessary training data for FLD Li et al. (2024) representation.

2 METHOD

In this section, we will first go through the basic approach of FLD Li et al. (2024), which is then followed by our refinement on the input motion sequence.

2.1 PRELIMINARIES

FLD Li et al. (2024) is inspired by and largely based on PAE Starke et al. (2022). To begin, we introduce the necessary notations used by both PAE and FLD. PAE tackles the challenges associated with learning the structure of the motion space—such as data sparsity and the highly nonlinear nature of the space—by focusing on the periodicity of motions in the frequency domain. The structure of PAE is illustrated in Fig. 1(a).

We denote trajectory segments of length H in the d-dimensional state space preceding time step t as $s_t = (s_{t-H+1}, \dots, s_t) \in \mathbb{R}^{d \times H}$, which serves as the input to PAE Starke et al. (2022). The autoencoder structure decomposes the input motions into c latent channels, producing a lower-dimensional embedding $\mathbf{z}_t \in \mathbb{R}^{c \times H}$ of the motion input. A subsequent differentiable Fast Fourier Transform extracts the frequency f_t, amplitude a_t, and offset b_t vectors of the latent trajectories, while the phase vector ϕ_t is computed using a separate fully connected layer. This parameterization process is denoted by p, and we have:

$$\mathbf{z}_t = \mathbf{enc}(s_t), \quad \phi_t, f_t, a_t, b_t = p(\mathbf{z}_t) \tag{1}$$

where $\phi_t, f_t, a_t, b_t \in \mathbb{R}^c$.

Next, the reconstructed latent trajectory segments $\hat{\mathbf{z}}_t \in \mathbb{R}^{c \times H}$ are computed using sinusoidal functions parameterized by the latent vectors:

$$\hat{\mathbf{z}}_t = \hat{p}(\phi_t, f_t, a_t, b_t) = a_t \sin\left(2\pi(f_t \mathcal{T} + \phi_t)\right) + b_t, \tag{2}$$

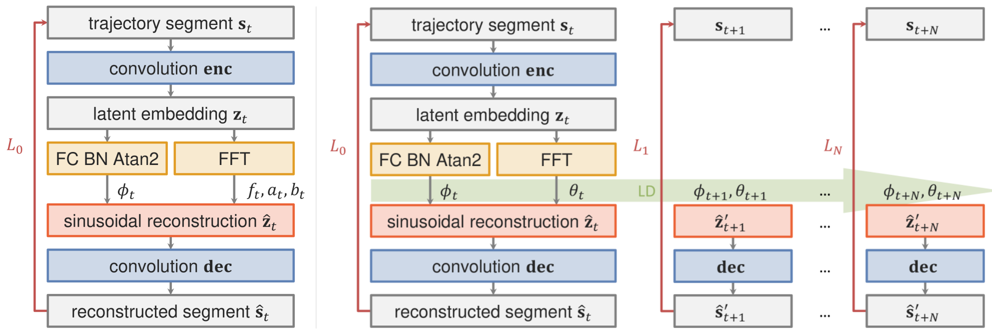
where $\hat{p}$ denotes the reconstruction process, and $\mathcal{T}$ represents the time window corresponding to the state transition horizon H. Finally, the network decodes the reconstructed latent trajectories $\hat{\mathbf{z}}_t$ back to the original motion space, and the reconstruction error is computed with respect to the original input:

$$\hat{s}_t = \mathbf{dec}(\hat{\mathbf{z}}_t), \quad L_0 = \mathrm{MSE}(\hat{s}_t, s_t) \tag{3}$$

where $\hat{s}_t \in \mathbb{R}^{d \times H}$, and MSE denotes the Mean Squared Error. For more details, we refer to the original work .

2.2 FOURIER LATENT DYNAMICS

Problem Formation We define the state space $\mathcal{S}$ and describe a motion sequence $\tau = (s_0, s_1, \dots)$ as a trajectory of consecutive states $s \in \mathcal{S}$ drawn from a reference dataset $\mathcal{M}$. FLD aims to develop a physics-based learning controller that can replicate the motions from the reference dataset and



(a) PAE structure (b) PAE structure with FLD extension

Figure 1: PAE Starke et al. (2022) structure and FLD training pipeline. During training, latent dynamics are enforced to predict proceeding latent states and parameterizations. The prediction loss is computed in the original motion space with respect to the ground truth future states.

generate new motions in response to novel target inputs. This approach enhances the controller's generality, allowing it to handle a wide range of motions beyond those in the dataset.

To achieve this, FLD employs a two-stage training pipeline. The first stage involves training an efficient representation model on the reference dataset, resulting in a continuously parameterized latent space. This latent space enables the synthesis of novel motions by sampling latent encodings. In the second stage, FLD develops a robust learning algorithm designed to track the diverse generated target trajectories. Throughout both stages, FLD emphasizes the importance of recognizing periodic or quasi-periodic patterns in the temporal progression of motions, which are characteristic of robotic motor skills.

Latent Dynamics Learning FLD's Li et al. (2024) analysis of the latent trajectories for periodic and quasi-periodic motions encoded by the PAE reveals that the frequency, amplitude, and offset vectors are nearly constant over time. This finding leads us to propose the quasi-constant parameterization assumption:

A latent trajectory $\mathbf{z} = (\mathbf{z}_t, \mathbf{z}_{t+1}, \ldots)$ *can be approximated by* $\hat{\mathbf{z}} = (\hat{\mathbf{z}}_t, \hat{\mathbf{z}}_{t+1}, \ldots)$ *with a bounded error* $\delta = ||\mathbf{z} - \hat{\mathbf{z}}||$*, where* $\hat{\mathbf{z}}_{t'} = \hat{p}(\phi_{t'}, f, a, b), \forall t' \in \{t, t+1, \ldots\}$.

FLD formalize the latent dynamics of FLD and its training process, as shown in Fig. 1(b). For a motion segment $s_t = (s_{t-H+1}, \ldots, s_t)$ with latent trajectory $\mathbf{z}_t$ parameterized by ϕ_t, f_t, a_t, and b_t, we predict the subsequent segment s_{t+1} using $\hat{s}_{t+1}$, decoded from i-step forward propagation $\mathbf{z}'_{t+i}$:

$$\mathbf{z}'_{t+i} = \hat{p}(\phi_t + if\Delta t, f_t, a_t, b_t), \quad \hat{s}'_{t+i} = \mathrm{dec}(\mathbf{z}'_{t+i}), \tag{4}$$

where Δt denotes the time step. Assuming locally constant latent parameters, FLD compute the prediction loss at time $t+i$. The local reconstruction in PAE can be viewed as zero-step forward regression using latent dynamics. FLD extend this to multi-step forward prediction and define the total loss for FLD with a maximum propagation horizon N and decay factor α:

$$L_N^{\mathrm{FLD}} = \sum_{i=0}^{N} \alpha^i L_i, \quad L_i = \mathrm{MSE}(\hat{s}_{t+i}, s_{t+i}), \tag{5}$$

where MSE denotes Mean Squared Error.

Motion Learning Given reference trajectories, physics-based motion learning algorithms train a control policy that actuates the joints of the simulated character or robot and reproduces the instructed motion trajectories.

At the beginning of each episode, a set of latent parameterizations $\theta_0 \in \mathbb{R}^{3c}$ is sampled from a skill sampler p_θ (e.g., a buffer of offline reference motion encodings). The latent state $\phi_0 \in \mathbb{R}^c$ is

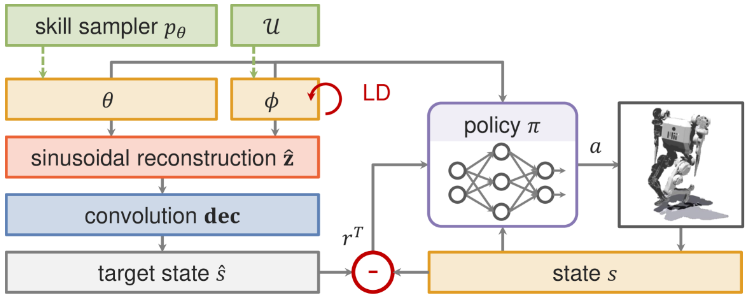


Figure 2: System overview. During training, the latent states propagate under the latent dynamics and are reconstructed to policy tracking targets $\hat{s}$ at each step. The tracking reward r^T is computed as the distance between the target $\hat{s}$ and the measured states s

uniformly sampled from a fixed range $\mathcal{U}$. The step update of the latent vectors follows the dynamics in Eq. 4:

$$\theta_t = \theta_{t-1}, \quad \phi_t = \phi_{t-1} + f_t \Delta t. \tag{6}$$

At each step, the latent state and parameterization are used to reconstruct a motion segment:

$$\hat{s}_t = (\hat{s}_{t-H+1}, \ldots, \hat{s}_t) = \text{dec}(\mathbf{z}_t) = \text{dec}(\hat{p}(\phi_t, \theta_t)), \tag{7}$$

where the most recent state $\hat{s}_t$ serves as a tracking target for the learning environment at the current time step. The tracking reward encourages alignment with the target.

The latent state and parameterization are provided to the observation space to inform the policy about the motion and the specific frame it should be tracking. Fig. 2 provides a schematic overview of the training pipeline.

2.3 Motion refinement

Due to the assumption described in Section. 2.2, FLD Li et al. (2024) motion representation approximates well only in cases that the input motion sequence are periodic or quasi-periodic. While in a survey on popular motion clip datasets, we found that a significant number of motions samples are non-periodic or are periodic but too short to process. We address this by applying an pre-processing onto the motion sequence data, which is described in Section. 3.1.

3 Experiments

3.1 Dataset and Resources

The latent dynamics learning stage of FLD requires a joint space motion data of the certain humanoid robotics for supervised learning. We finally collect for each type of humanoid robot a dataset of 10 reference motion data including running, raising arms, side-jogging, etc. The experiment proves that dataset of these quatity is sufficient to learn a good dynamic motion

Humanoid Resources We carry out experiments on two types of humanoids, which are MIT Humanoid Chignoli et al. (2021) and Unitree H1. We use the official released URDF description and meshes for training and simulation.

Motion retargeting Given the lack of existing motion data of these humanoids, we use motion retargeting techniques to obtain referenced robotics data from motion captures dataset of human beings. In this case, we use the dataset collected by Peng et al. in their work. With the URDF descrption of humanoid robot shape, we achieve this by inverse kinematics in joint space, implemented using Pybullet IK toolkit.

Case	MSE (vel)↓	MSE (arm)↓	MSE (leg)↓
M.H. w/o FLD	**0.36**	0.25	0.23
M.H. w/ FLD	3.55	-	-
H1 w/o FLD	0.47	0.72	0.39
H1 w/ FLD	4.31	-	-

Table 1: Motion prediction evaluation. M.H. stands for MIT Humanoid robot Chignoli et al. (2021), and H1 stands for Unitree H1. The velocity MSE has unit $\mathrm{m/s}$. The arm and leg rotation has unit rad.

Motion wrapping and smoothing While in a survey on popular motion clip datasets, we found that a significant number of motions samples are non-periodic or are periodic but too short to process. To address these isssues, we apply the following pre-processing onto the collected motion clips.

- For non-periodic motion clips, we wrap them by concatenating its end with its begining, and repeat for at least 5 times.
- For periodic motion clips that are shorter than 30 frames, we repeat them for at least 3 times.

3.2 Implementation Details

We use PPO as motion learning prior and use Isaac Gym Makoviychuk et al. (2021) for physical-aware simulation and visualization.

3.3 Results

Motion prediction and synthesis We carry out quantified experiments on humanoid motion prediction to evaluate model's ability. For each experiment, we warmup motion control model for $t = 30$ timesteps, and after that, we use our model to prediction the following motion sequence and randomly sample $n = 50$ predicted timesteps to compute their MSE with respect to the ground truth data. The result is shown in Tab. 1

Motion simulation We also carry motion control experiments in the Isaac Gym Makoviychuk et al. (2021) simulator, as shown in Fig. 3. For video demos, please check `https://drive.google.com/drive/folders/1OQbhcQAh_m5SoiOqcgzmUJ35NSI69nOF?usp=sharing`

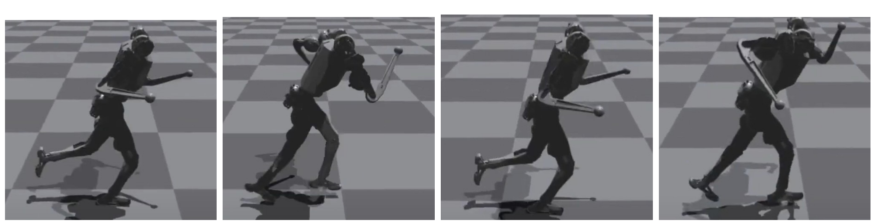


Figure 3: Motion demo. MIT Humanoid robot Chignoli et al. (2021) motion prediction results.

4 Conclusion

In this paper, we implement and utilize the FLD Li et al. (2024) pipeline to learn a structured motion representation, which benefits the policy model to control the motion of humanoid robots. We ultimately manage to control full-body motion for several types of humanoids (including MIT humanoid Chignoli et al. (2021) and Unitree H1) in Isaac Gym simulation environment Makoviychuk et al. (2021). We further make some refinement on the existing FLD pipeline, resolving some limitation on the training motion sequence. We also implement a motion retargeting pipeline that retargets motion data in joint space from human motion clips released by DeepMimic Peng et al. (2018). This pipeline helps to collect the necessary training data for FLD representation.

ACKNOWLEDGMENTS

We would like to express our gratitude to Xialin He, a senior undergraduate student at Shanghai Jiao Tong University, for his guidance on this project. We are also thankful for the advice provided by Yichao Zhong.

REFERENCES

Matthew Chignoli, Donghyun Kim, Elijah Stanger-Jones, and Sangbae Kim. The mit humanoid robot: Design, motion planning, and control for acrobatic behaviors. In *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, pp. 1–8. IEEE, 2021.

Chenhao Li, Elijah Stanger-Jones, Steve Heim, and Sangbae Kim. Fld: Fourier latent dynamics for structured motion representation and learning. *arXiv preprint arXiv:2402.13820*, 2024.

Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel Van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions On Graphics (TOG)*, 37(4):1–14, 2018.

Sebastian Starke, Ian Mason, and Taku Komura. Deepphase: Periodic autoencoders for learning motion phase manifolds. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022.