Skeleton2Humanoid: Animating Simulated Characters for Physically-plausible Motion In-betweening

Yunhao Li[*]
Institute of Image Communication
and Network Engineering
Shanghai Jiao Tong University
Shanghai, China
lyhsjtu@sjtu.edu.cn

Zhenbo Yu[*]
Shanghai Jiao Tong University
Shanghai, China
yuzhenbo@sjtu.edu.cn

Yucheng Zhu
Institute of Image Communication
and Network Engineering
Shanghai Jiao Tong University
Shanghai, China
zyc420@sjtu.edu.cn

Bingbing Ni
Shanghai Jiao Tong University
Shanghai, China
nibingbing@sjtu.edu.cn

Guangtao Zhai[†]
Institute of Image Communication
and Network Engineering
Shanghai Jiao Tong University
Shanghai, China
zhaiguangtao@sjtu.edu.cn

Wei Shen[†]
MoE Key Lab of Artificial Intelligence,
AI Institute
Shanghai Jiao Tong University
Shanghai, China
wei.shen@sjtu.edu.cn
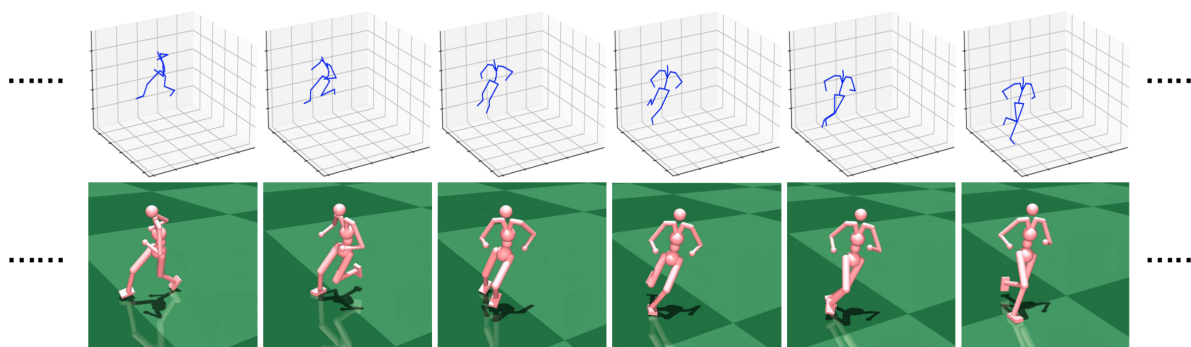
Figure 1: Our Skeleton2Humanoid system can directly synthesize a complete humanoid character transition motion in a physics simulator (Bottom) given past keyframes and a future keyframe (Top). Our system can produce both accurate and physically-plausible character motions.

ABSTRACT

Human motion synthesis is a long-standing problem with various applications in digital twins and the Metaverse. However, modern deep learning based motion synthesis approaches barely consider the physical plausibility of synthesized motions and consequently they usually produce unrealistic human motions. In order to solve this problem, we propose a system "Skeleton2Humanoid" which performs physics-oriented motion correction at test time by regularizing synthesized skeleton motions in a physics simulator. Concretely, our system consists of three sequential stages: (I) test time motion synthesis network adaptation, (II) skeleton to humanoid matching and (III) motion imitation based on reinforcement learning (RL). Stage I introduces a test time adaptation strategy, which improves the physical plausibility of synthesized human skeleton motions by optimizing skeleton joint locations. Stage II performs an analytical inverse kinematics strategy, which converts the optimized human skeleton motions to humanoid robot motions in a physics simulator, then the converted humanoid robot motions can be served as reference motions for the RL policy to imitate. Stage III introduces a curriculum residual force control policy, which drives the humanoid robot to mimic complex converted reference motions in accordance with the physical law. We verify our system on a typical human motion synthesis task, motion-in-betweening. Experiments on the challenging LaFAN1 dataset show our system can outperform prior methods significantly in terms of both physical plausibility and accuracy. Code will be released for research purposes at: https://github.com/michaelliyunhao/Skeleton2Humanoid.

arXiv:2210.04294v1 [cs.CV] 9 Oct 2022

CCS CONCEPTS

• **Computing methodologies** → **Motion capture**.

KEYWORDS

3D motion in-betweening; inverse kinematics; reinforcement learning; 3D animation

1 INTRODUCTION

Synthesizing both accurate and realistic virtual human motions has been a widely explored but challenging task in computer vision and graphics [48, 49] with various applications in digital twins and the Metaverse. Recently, deep learning sheds light onto a way to generate accurate human motions and has been applied to various motion synthesis tasks, such as human motion prediction [2–4, 11, 16, 17, 53–55, 60], human motion completion [31, 58, 59] and human motion in-betweening [1, 47, 56, 57]. Although they have shown great performance on synthesizing accurate human body motions with small skeleton joint errors comparing with ground truth motions, they fail to model the motions under the physics laws. Consequently, the synthesized motions are usually physically implausible. For example, the synthesized feet often penetrate the ground, the body joints are rotated with impossible angles, the whole body motions are unsmooth, the synthesized feet slide back and forth while they should be in static and touch the ground. These synthesized artifacts significantly limits the application of motion synthesis on the virtual human animation and the incoming Metaverse because they easily make humans feel unrealistic.

Utilizing humanoid characters in a physics simulator to optimize motions is a promising solution because the physics simulator can guarantee the physical plausibility of the generated motions. Prior works [39, 40, 52] utilized reinforcement learning (RL) to actuate the humanoid character to imitate various reference mocap data for creating physical character animation. Inspired by them, Recent works [8, 29] also attempted to utilize RL to imitate motions synthesized by deep neural networks, in the format of skeletons or SMPL [9] models, aiming at producing physically-plausible motions for 3D pose estimation. However, these methods are only validated on simple motions such as walking and talking in the Human3.6m dataset and cannot generalize well to complex motions or irregular motions. In addition, RL based imitation requires transferring synthesized human skeleton motions to humanoid motions, where a humanoid character should be carefully designed to exactly match the human skeletons in terms of both shapes and the kinematics tree. This limits RL based imitation to transfer motions between skeleton and humanoid with different shapes and kinematics trees.

To address these issues, we propose Skeleton2Humanoid, a novel system which is able to improve the physical plausibility of the motions synthesized from motion synthesis networks, though the transfer from human skeleton motions to humanoid character motions. Our Skeleton2Humanoid system consists of three sequential stages: **(I) Test Time Motion Synthesis Network Adaptation:** We adapt the motion synthesis network with a few gradients on the test data using two new self-supervised losses, a foot contact consistency loss and a motion smoothness loss, which can improve the physical plausibility of the predicted motions. **(II) Skeleton to Humanoid Matching:** We match the synthesized human skeleton motions to humanoid character motions by a novel general analytical inverse kinematic method. Inverse kinematics is able to convert human skeleton motions to humanoid motions even when the body structure is different from the human skeleton. **(III) Motion Imitation base on RL:** Finally, we animate the humanoid character to imitate various synthesized motions. Specifically, based on recent work [26, 29], we propose a curriculum residual force control humanoid control policy (CRP) by introducing a curriculum learning paradigm that dynamically adjusts a residual force scale during RL training, which can improves asymptotic RL performance on imitating various synthesized motions. To verify the effectiveness of our Skeleton2Humanoid system, we select "motion in-betweening" task, as it is a recent proposed challenging motion prediction task [1, 47] for evaluation. Motion in-betweening aims at predicting the transition motions between the past given keyframes and a provided future keyframe. Experiments on challenging LaFAN1 dataset show the superiority of our Skeleton2Humanoid system.

The main contributions of this paper are as follows: **(1)** We present Skeleton2Humanoid, a new system that converts human skeleton motions to humanoid character motions to produce physical plausible motions. **(2)** Our proposed test time adaptation stage can further improve the prediction accuracy and physical plausibility on large mocap dataset LaFAN1 for the motion in-betweening task. With test time adaptation, we achieve a new benchmark accuracy on the motion in-betweening task. **(3)** Our proposed curriculum residual force control policy enables finer character control and outperforms prior arts on motion imitation. **(4)** Our whole Skeleton2Humanoid system significantly improves the performance of human in-betweening motions on physical plausibility and achieves comparable motion prediction accuracy.

2 RELATED WORK

Human/character motion synthesis: Motion synthesis is a general term which contains several tasks including motion prediction, in-betweening and completion. Motion prediction aims at predicting future human motions given past motions. Deterministic motion prediction estimates a single accurate motion and prior works used various network architectures including recurrent neural network [2–4], graph convolution network [61] or transformer [16] to model human motions. Stochastic motion prediction produces diverse future human motions by utilizing generative model such as VAE [6, 17, 55, 66], GAN [12, 14, 65]. Motion completion and in-betweening aim at filling gaps of motion with predefined keyframe constraints. Current works utilized convolution networks [31, 57, 59, 62], recurrent networks [1, 63] or transformers [47] to synthesize accurate and consistent results. For instance, Harvey et al. [1] proposed a transition generation technique based on recurrent neural networks for motion in-betweening task. Duan et al. [47] utilized transformer architecture to model human motions in a sequence-to-sequence manner for the motion in-betweening task.
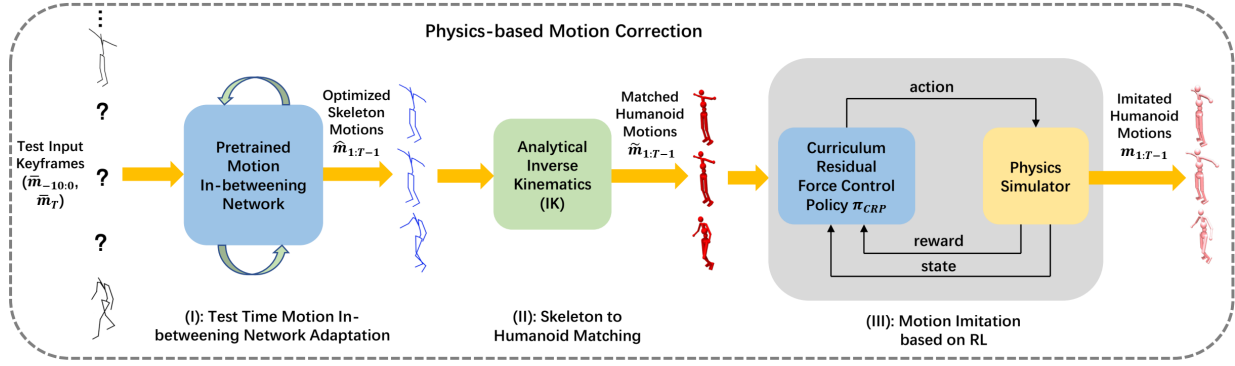
Figure 2: An overview of our Skeleton2Humanoid system on the motion in-betweening task. Given the test data containing past keyframes $\overline{m}_{-9:0}$ and a future keyframe $\overline{m}_T$, Stage I optimizes skeleton joint locations by test time adaptation and produces more plausible skeleton motions $\hat{m}_{1:T-1}$. Stage II converts the optimized skeleton motions $\hat{m}_{1:T-1}$ to humanoid motions $\tilde{m}_{1:T-1}$ in the physics simulator by analytical inverse kinematics. Stage III finally drives the humanoid to mimic converted skeleton motions $\tilde{m}_{1:T-1}$ to produce physically-plausible humanoid motions $m_{1:T-1}$.
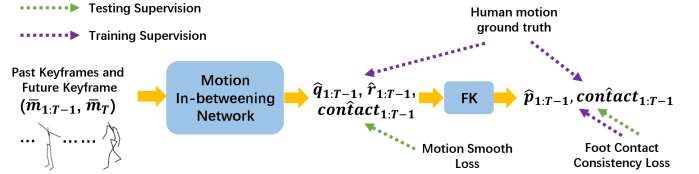
Test Time Adaptation: Test time adaptation is a recently proposed method that utilize the self-supervised distribution information from the test data presented at test time to quickly adapt models with a few gradient steps [32, 33, 36], which can further improve the model performance on test data. The first work [32] introduced test time adaptation by proposing an auxiliary branch with self-supervised rotation prediction loss to adapt the classification model. Wang et al. [36] minimized the predicted entropy of classification model on test data to improve the performance. Recently, more works start to utilize test time adaptation on the 2D/3D human pose related task [34, 35]: For instance, Guan et al. [35] proposed an online bilevel adaptation framework for 3D human mesh reconstruction which greatly improves model generalization. In contrast to other works, Our approach is the first one to study test time adaptation on the human motion in-betweening task.

Reinforcement Learning for Humanoid Character Control: Deep RL is a promising approach for learning character control policies [37–41] to help character perform various motions. Peng et al. [39] first utilize hand craft rewards to imitate a single sequence of human poses. Recently, some works [8, 29, 42, 43, 51] used RL to produce simple human motions from egocentric videos for ego-pose estimation or 3d human pose estimation. Yuan et al. [26] proposed to add external residual forces and help characters to better imitate agile single reference motions. In addition, some works [44–46] utilized deep RL to learn a interactive controllable policies from large motion capture data for character animation. However, Prior works mostly focused on learning control policies on motion capture data, while we learn a policy to imitate synthesized motions. We propose a curriculum residual force control policy (CRP) that can better imitate diverse motions.

3 APPROACH

3.1 System Overview

The human motion in-betweening task can be formulated as: given the past 10 human skeleton poses $\overline{m}_{-9:0}$ and a future skeleton keyframe $\overline{m}_T$ at time T, we want to recover the ground truth

Figure 3: Details of our test time adaptation method. $\hat{q}$, $con\hat{t}act$, FK and $\hat{p}$ represent the predicted root positions, the contact prediction of feet joints, the forward kinematic process and 3d joint positions for human skeleton, respectively.

transition motions $\overline{m}_{1:T-1}$. Given a pretrained typical motion in-betweening network [1], our Skeleton2Humanoid performs a physics-oriented motion correction consists of 3 stages as presented in Fig. 2 to optimize synthesized in-betweening motions. Stage I optimizes the pretrained motion in-betweening network at test time to predict more physically-plausible skeleton transition motions $\hat{m}_{1:T-1}$. Then Stage II transfers the optimized skeleton motions $\hat{m}_{1:T-1}$ to humanoid motions $\tilde{m}_{1:T-1}$ through analytical inverse kinematics. Finally, Stage III learns a curriculum residual force control policy to imitate the transferred humanoid motions $\tilde{m}_{1:T-1}$ to produce physically-plausible humanoid motions $m_{1:T-1}$.

In our Skeleton2Humanoid framework, $\overline{m}_t$ and $\hat{m}_t$ are skeleton motions, and $\overline{m}_t$ is represented by $\overline{m}_t \triangleq (\overline{q}_t, \overline{r}_t, \overline{p}_t)$, where $\overline{q}_t$ and $\overline{r}_t$ denote body joint angles in quaternions and root translation, $\overline{p}_t$ denotes 3d joint positions calculated by forward kinematics. Similarly, $\hat{m}_t \triangleq (\hat{q}_t, \hat{r}_t, \hat{p}_t)$. In addition, $\tilde{m}_t$ and m_t are humanoid motions, $\tilde{m}_t$ is represented by $\tilde{m}_t \triangleq (\tilde{q}_t, \tilde{r}_t, \tilde{p}_t)$, where $\tilde{q}_t, \tilde{r}_t$ and $\tilde{p}_t$ denote joint angles in euler angles, root translation and 3d joint positions of the reference humanoid motions. Similarly, $m_t \triangleq (q_t, r_t, p_t)$.

3.2 Test Time Motion In-betweening Network Adaptation

3.2.1 Adaptation for Physically-plausible Skeleton Motion. Previous human motion in-betweening model [1] has shown great performance on synthesizing accurate human motions. However, it

still suffers from implausibility on testing data because of the domain shift between training data and testing data. For example, the synthesized human motions usually have the foot sliding problem which makes motions look strange and it is hard for humanoid in a physics simulator to imitate this motion. In addition, the root joint velocity of the synthesized human motion sometimes changes drastically and results in on unsmooth motion. Inspired by these observations, we propose a test time adaptation method which updates the motion in-betweening network on test data and design two new self-supervised losses for test time adaptation.

Foot contact consistency loss: We encourage that the feet should be in static contact with the ground when performing various motions. The loss is:

$$L_{contact} = \sum_{t=0}^{t=T-2} \sum_{foot \in F} \left| \hat{p}_{t+1}^{foot} - \hat{p}_t^{foot} \right| \cdot \hat{contact}_{t+1}^{foot}, \quad (1)$$

where $\hat{p}_t^{foot}$ is the predicted 3D position of the foot joint at time t, F is a subset of joints which contains all foot-related joints, i.e. "left ankle", "right ankle", "left front feet" and "right front feet". $\hat{contact}_{t+1}^{foot}$ represents the predicted contact probability for the foot joint at time t.

Motion smoothness loss: We encourage the root joint positions along the time axis to be smooth. The motion smoothness loss is:

$$L_{smooth} = \sum_{t=0}^{T-2} \|\hat{r}_{t+1} - \hat{r}_t\|_2, \quad (2)$$

where r_t is the 3D position of root joint at time t.

By utilizing these two losses, we can optimize the motion in-betweening network which is trained on the training set given the test set. In the following section, we will introduce our method in detail.

3.2.2 Implementation Detail. Now, we describe the training phase and the testing phase of our test time adaptation method, respectively.

During training, as shown in Fig. 3, we select the typical motion in-betweening network [1] as our baseline to generate transition human skeleton motions. The training process is the same as [1]. The training loss is formulated as:

$$L_{train} = L_{inbetween} + \alpha L_{contact}, \quad (3)$$

where $L_{inbetween}$ is the sum of all the training losses used in [1], $L_{contact}$ is the foot contact consistency loss, and α is a constant coefficients.

During testing, we only use the self-supervised losses to update the motion in-betweening model for several epochs on the whole test data. The testing loss is formulated as:

$$L_{test} = L_{contact} + \beta L_{smooth}, \quad (4)$$

where β is a constant coefficients. After test time adaptation, the motion in-betweening network is able to produce optimized plausible skeleton motions $\hat{m}_{1:T-1}$.

3.3 Skeleton to Humanoid Matching

After acquiring optimized skeleton motions $\hat{m}_{1:T-1}$ from stage I, we then transfer the skeleton motions $\hat{m}_{1:T-1}$ to corresponding humanoid motions $\tilde{m}_{1:T-1}$ which can be imitated by RL. To achieve
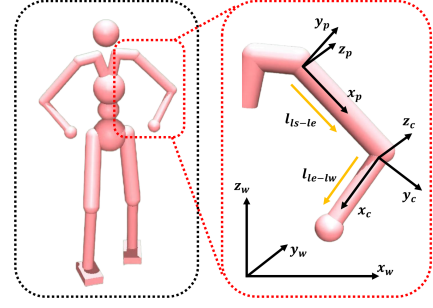
Figure 4: Illustration of skeleton to humanoid matching process in left elbow. (x_w, y_w, z_w), (x_c, y_c, z_c) and (x_p, y_p, z_p) represent the world coordinate, the child coordinate(left elbow) and the parent coordinate(left shoulder) respectively.

this, recent works [8, 29] constructed a humanoid model which is exactly the same as the skeleton or the SMPL model in shapes and kinematics tree, hence it is easy to transfer the motions to humanoid motions with no error because they can directly transfer the axis-angles of the SMPL model to the corresponding euler angles of humanoid. However, we emphasize that it is not suitable for all cases in practice.

First, this method is not general and cumbersome because it needs to carefully construct a humanoid model and it is not possible to flexibly transfer motions between human skeleton and humanoid with different shapes and kinematics trees. Second, constructing an exactly same humanoid model is an ideal solution because the pose of humanoid $\tilde{m}_t$ in the physics simulator is represented by euler angles $\tilde{q}_t$ with specific ranges which corresponds body joints constraints. For example, the euler angle of the shoulder joint in z axis is usually between -90 and 90 degrees. However, transferring the quaternions of the skeleton to the corresponding euler angles of humanoid in a specific range is actually an ill-posed problem. The range of the euler angle in each axis is between -180 and 180 degrees, hence one rotation can be represented by many different euler angles. For human skeletons with strange poses or strange kinematics trees, it is difficult to transfer the quaternions to the corresponding euler angles in a specific range.

To solve this problem, inverse kinematics is a solution that can convert the 3d joint positions of skeleton into the euler angles of humanoid. Analytical Inverse kinematics calculates the rotation vectors at the end of a kinematic chain in a given position and can match motions between skeletons and humanoids with different shapes and kinematics trees. Previous works use analytical inverse kinematics [18, 19] to convert the 3d joint positions of skeleton to the axis-angles of the SMPL model. We extend their works to convert 3d joint positions of skeleton to the euler angles of humanoid via specific matching equations. Concretely, we select a suitable axis definition for each joint of the skeleton that can compute the corresponding euler angle of the humanoid model. We take the left elbow for instance to describe the detailed matching process in Fig. 4. Following [19], we consider the skeleton and the humanoid as articulated bodies with pairs of parent-children joints. Suppose the parent and the child coordinate system of joints in 3d skeleton is

denoted as $[x_p, y_p, z_p]$ and $[x_c, y_c, z_c]$, then the parent and child coordinate system of the left elbow is calculated by Eqn. 24,

$$\begin{cases} [x_p, y_p, z_p] = \left[\dfrac{l_{ls-le}}{||l_{ls-le}||}, \dfrac{z_p \otimes x_p}{|z_p \otimes x_p|}, \dfrac{-l_{ls-le} \otimes l_{le-lw}}{||l_{ls-le} \otimes l_{le-lw}|} \right], \\ [x_c, y_c, z_c] = \left[\dfrac{l_{le-lw}}{||l_{le-lw}||}, \dfrac{z_c \otimes x_c}{|z_c \otimes x_c|}, z_p \right] \end{cases} \quad (5)$$

where each item (e.g., x_c) is a 3*1 vector in world coordinate system. I_{a-b} indicates the vector pointing from joint a to joint b and subscripts ls, le, lw denote left shoulder, left elbow and left wrist, respectively. Specifically, l_{le-lw} means the vector pointing from the left elbow joint to the left wrist joint. Then we can calculate the euler angle E of the left elbow of the humanoid via Eqn. 6,

$$\begin{cases} \vec{N} = [n_0, n_1, n_2] = [x_c \cdot x_p^T, x_c \cdot y_p^T, x_c \cdot z_p^T] \\ \vec{O} = [o_0, o_1, o_2] = [y_c \cdot x_p^T, y_c \cdot y_p^T, y_c \cdot z_p^T] \\ \vec{A} = [a_0, a_1, a_2] = [z_c \cdot x_p^T, z_c \cdot y_p^T, z_c \cdot z_p^T] \\ E_z = atan2(\dfrac{n_1}{n_0}) \\ E_y = atan2(\dfrac{-n_2}{n_0 * cos(E_z) + n_1 * sin(E_z)}) \\ E_x = atan2(\dfrac{-a_1 * cos(E_z) + a_0 * sin(E_z)}{o_1 * cos(E_z) - o_0 * sin(E_z)}) \\ E = [E_z, E_y, E_x] \end{cases} \quad (6)$$

where E is the euler angle of the humanoid model in z-y-x rotation order. After applying the similar matching operation on the other joints of the humanoid, the whole human skeleton motions $\hat{m}_{10:T-1}$ can be converted to humanoid motions $\tilde{m}_{10:T-1}$. In addition, because the matching process is applied frame by frame, the values of the converted euler angles usually flip and become discontinued due to the $atan2$ process in equation 6. We correct the values of euler angles by adding a temporally checking process, suppose E_z^t is a euler angle of a joint in z axis at time t,

$$\begin{cases} E_z^{t+1} = E_z^{t+1} + 2\pi, \ \text{if } E_z^t > 0 \text{and} |E_z^{t+1} - E_z^t| > \lambda \\ E_z^{t+1} = E_z^{t+1} - 2\pi, \ \text{if } E_z^t <= 0 \text{and} |E_z^{t+1} - E_z^t| > \lambda \end{cases}, \quad (7)$$

where λ is a threshold value for checking if the euler angles are discontinued or not and we set λ to 5. In the end, we can convert the synthesized skeleton motions $\hat{m}_{1:T-1}$ to humanoid motions $\tilde{m}_{1:T-1}$.

3.4 Motion Imitation based on RL

After getting the converted humanoid motions $\tilde{m}_{1:T-1}$, they are still physically implausible due to lack of physical constraint. Motivated by [8, 26, 29], we utilize reinforcement learning to learn a curriculum residual force control policy π_{CRP} which can imitate $\tilde{m}_{1:T-1}$ to generate physically-plausible humanoid motions $m_{1:T-1}$. The motion imitation problem is usually formulated as a Markov decision process. Given a reference motion and current state s_t, the humanoid agent interacts with the physics simulator environment by action a_t and receive reward r_t. The action is produced by a policy $\pi_{CRP}(a_t|s_t)$ conditioned on state s_t. The reward encourages the agent to act like the reference motion. When an action is taken,
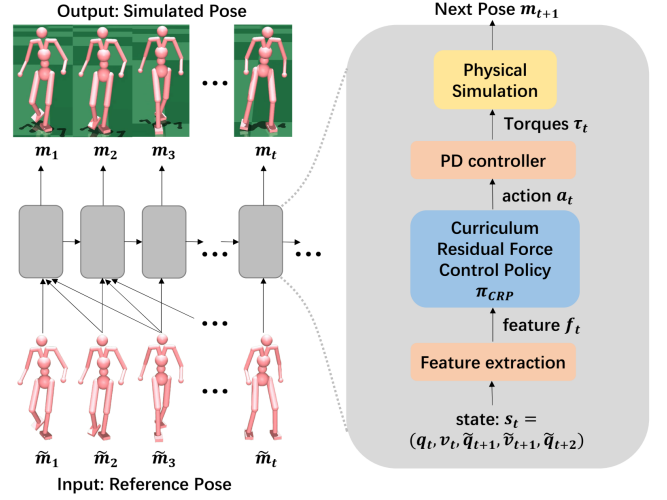
Figure 5: Overview of a curriculum residual force control policy π_{CRP}. Our π_{CRP} can actuate the humanoid to iteratively imitate the reference pose in next one frame precisely given the reference pose in next two frames.

the current agent state s_t changes to next state s_{t+1}. The goal of the motion imitation is to learn a policy that can maximizes the average cumulative rewards.

3.4.1 Motion Imitation Paradigm. Here we detail the states, actions, rewards and the training process.

State. Previous work [8, 26, 39] construct the humanoid state s_t which contains the information of the character's current pose and the information of the reference pose in the next frame. We utilize the information of the reference pose in the next two frames that can learn a finer control policy. Specifically, the state $s_t \triangleq (q_t, v_t, \tilde{q}_{t+1}, \tilde{v}_{t+1}, \tilde{q}_{t+2})$ and we apply a hand-craft feature extraction to obtain informative features f_t as the input of π_{CRP} from s_t. We formulate $f_t \triangleq (q_t, v_t, j_t, g_t, (\tilde{q}_{t+1} - q_t), (\tilde{j}_{t+1} - j_t), (\tilde{g}_{t+1} - g_t), (\tilde{v}_{t+1} - v_t), (\tilde{q}_{t+2} - q_t), (\tilde{j}_{t+2} - j_t), (\tilde{g}_{t+2} - g_t), (\tilde{v}_{t+2} - v_t))$, where q_t is the euler angle of the simulated humanoid, v_t is the joint velocity computed by finite difference, j_t and g_t are the 3d joint position in local head coordinates and global coordinates, $\tilde{q}_{t+1}$ is the euler angle of reference pose in time t+1, $\tilde{v}_{t+1}$ the joint velocity of the reference pose in time t+1, $\tilde{j}_{t+1}$ and $\tilde{g}_{t+1}$ are the 3d joint position of the reference pose in local head coordinates and global coordinates.

Action. Following prior works [8, 26, 29], we utilize proportional derivative (PD) controller [28] at each non-root joint to generate torques. Under this setting, the action a_t is the target joint angles u_t of the PD controllers. The joint torques τ_t are specifically calculated by:

$$\tau_t = k_p \circ (u_t - q_t) - k_d \circ \dot{q}_t, \quad (8)$$

where k_p and k_d are manually defined parameters and $\circ$ is element-wise multiplication. The PD controllers act like damped springs that drive joints to target angles u_t, while k_p and k_d are the stiffness and damping of the springs. Prior works [26, 29] also allowed the policy to apply external residual forces $\eta_t = k_r \circ S_r$ to joints to improve the motion imitation performance, S_r is a predefined residual force

scale and $k_r \in [0, 1]$ is a weighting parameter learned by policy . In section 3.4.2, we will introduce our curriculum residual force learning that can dynamically adjust the residual force scale S_r in RL training. Overall, our action is $a_t \triangleq (u_t, \eta_t)$.

Reward. Following [26, 29], we implement the same reward function that encourages the policy to mimic the reference pose.

3.4.2 Curriculum Learning for Residual Forces. The residual forces τ_t proposed by [26] aims at utilizing additional forces to enable finer control of the humanoid character. Recent works [8, 26] used it to perform humanoid motions with extra interaction such sitting on the chair even there is no chair in the physics simulator. However, large scale residual forces will make humanoids perform some unnatural motions or overly energetic movements. Meanwhile, removing the residual forces will cause the low converge speed of RL and a bad imitation performance. To balance the trade-off between the motion naturalness and the imitation performance, we propose a curriculum learning strategy that gradually adjusts the residual force scale in the training of reinforcement learning.

Starting from a large residual forces, we gradually decrease the residual force scale in RL training. Suppose i is the training iteration of PPO RL algorithm, S_r is residual force scale which is adjusted by:

$$S_r = \begin{cases} S_r^{init}, & \text{if } 0 \le i \le i_{start} \\ S_r^{init} - \alpha * (i - i_{start}), & \text{if } i_{start} \le i \le i_{end} , \\ S_r^{init} - \alpha * (i_{end} - i_{start}), & \text{if } i \ge i_{end} \end{cases} \quad (9)$$

where i_{start} and i_{end} are the start and end iterations of the curriculum, and α is a hyper-parameter indicating the changing rate of S_r. We empirically set $S_r^{init} = 220, i_{end} = 1300, i_{start} = 100, \alpha = 0.1$ and train RL for 1500 iterations. By introducing curriculum learning for residual forces, the policy can quickly learn to imitate the reference motion precisely with the help of large residual forces and learn to maintain the imitation performance while getting rid of large residual forces.

4 EXPERIMENTS

In this section, we perform experimental evaluations of our Skeleton2Humanoid system on the motion realism and accuracy, then we analyze the properties of the test time adaptation stage, the skeleton to humanoid matching stage and the motion imitation stage based on RL individually.

Dataset. To demonstrate the effectiveness of our Skeleton2Humanoid system on the motion in-betweening task, We employ the following datasets in our experiments: (1) **LaFAN1** [1], which is a widely used large scale dataset for evaluating motion in-betweening performance. LaFAN1 dataset contains various complex motions such as fighting, crawling and climbing obstacles. To be noticed, We only evaluate the test time adaptation stage of our Skeleton2Humanoid system on LaFAN1 dataset and we will explain it later. (2) **Noobjects LaFAN1** is a subset of LaFAN1 dataset for evaluating the physical plausibility of our synthesized in-betweening motions. LaFAN1 dataset contains many human object interaction motions such as climbing obstacles and we can not construct objects in the physics simulator due to the absence of the ground truth of

3d objects, hence we remove these motions and only consider human motions performing on the ground. We evaluate our whole Skeleton2Humanoid system on this dataset.

Evaluation Metrics. We use the following metrics including physics-based and accuracy-based metrics for all following evaluations: (1) **FP**, which is the foot penetration that computes the mean penetration distance of feet joints and ground following [64]. (2) **FQ**, which computes the frequency of the penetrations between the feet and ground penetrations. (3) **JQ**, which similarly computes the frequency of penetrations between all the joints and the ground. (4) **SM**, which represents motion smoothness that computes the average L2 distance between body joints within two adjacent frames. (5) **MPJPE**, which is the mean per joint position error for evaluating the mismatch between poses. (6) **L2Q** and **L2P**, where L2Q denotes the the average L2 distances of the global quaternions between the predicted motions and their ground truth. Similarly, L2P denotes the average L2 distance of the global positions. We follow standard evaluations from [1] to compute them.

4.1 Evaluation of Skeleton2humanoid System

Experiment Setting. We evaluate our whole Skeleton2Humanoid system on the Noobjects LaFAN1 dataset. We select the typical state-of-the-art motion in-betweening approach Harvey et al. [1] as a baseline and validate our approach based on it. The baseline method is trained with various transition lengths (5 frames to 30 frames) between past keyframes and a future keyframe and is evaluated when transition lengths are 5 frames, 15 frames and 30 frames.

Results and Discussion. The results on the Noobjects LaFAN1 dataset are summarized in table 1. First, it is evident that our Skeleton2Humanoid outperforms the baseline in all physics-based metrics and achieves comparable performance in accuracy-based metrics. In particular, Skeleton2Humanoid achieve significantly lower FP, FQ, JQ and SM, which demonstrates that utilizing the simulated humanoid character can effectively reduce the physical artifacts such as the penetration problem between human bodies and the ground and can produce physically-plausible motions. Second, we can observe that our test time adaptation stage (TTA) outperforms the baseline in both physics-based metrics and accuracy-based metrics, demonstrating that our TTA can optimize the motion in-betweening model to produce not only plausible but also accurate motions. Third, we find that applying the analytical inverse kinematics and the curriculum residual force control policy (CRP) cause worse L2P. The reason is that our efficient analytical inverse kinematics strategy can not perfectly matching the skeleton and humanoid even its matching error is already really low (MPJPE: 11.29mm) as discussed in section. 4.2.2, and our state-of-the-art motion imitation policy CRP as shown in section. 4.2.3 still struggle to imitate various complex in-betweening motions perfectly. Such a phenomenon is also observed in [29] when they utilize RL to imitate human motions for ego pose estimation. We emphasize that current motion in-betweening method should not only consider the accuracy of synthesized motions but also consider the physical plausibility.

We also provide qualitative results in Fig. 6, we can see that the imitated humanoid motions can accurately synthesize joint contact with the ground and without penetration. Our state-of-the-art RL policy is also able to control the humanoid to perform complex
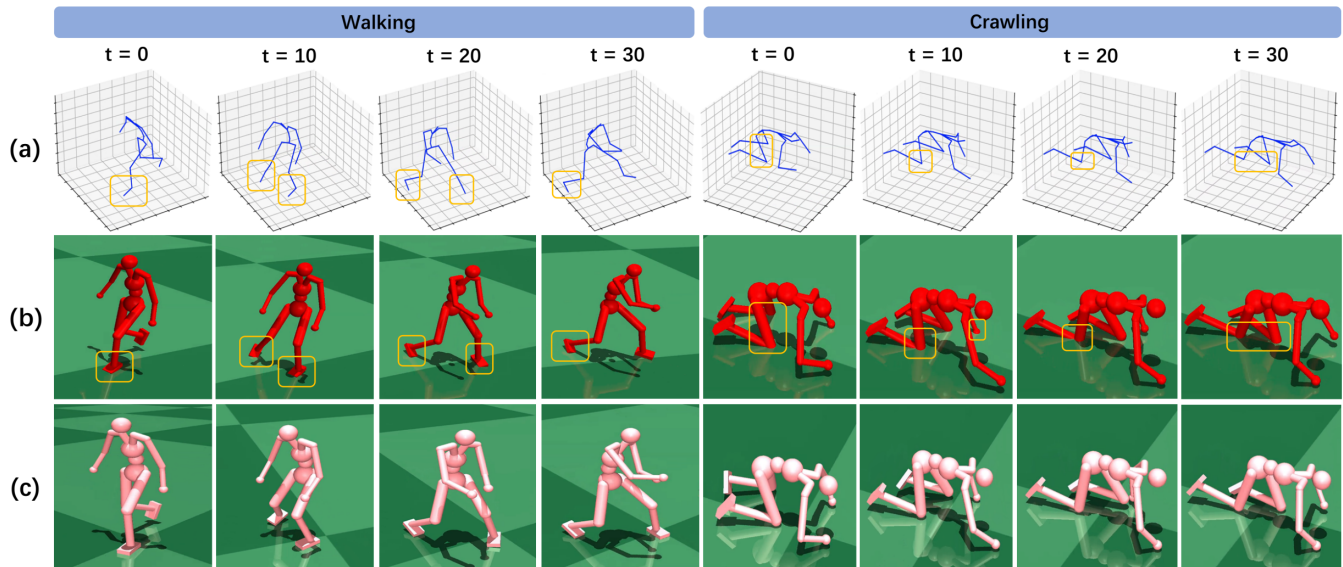
Figure 6: Qualitative Results of motions synthesized by our Skeleton2Humanoid system. The top row shows the skeleton motions (blue) after our test time adaptation stage (TTA), the middle row shows the matched humanoid motions (red) by our analytical inverse kinematics method (IK), the bottom row shows our final imitated humanoid motions (pink) by our curriculum residual force control policy π_{CRP}. Ground penetration is highlighted with boxes.

	FP			FQ			JQ			SM			L2P		
length	5	15	30	5	15	30	5	15	30	5	15	30	5	15	30
baseline[1]	-0.47	-0.43	-1.43	1.04%	4.12%	8.37%	0.68%	1.25%	2.08%	20.8	20.8	20.2	0.19	0.55	1.16
baseline w/ TTA	**-0.04**	-0.29	-1.13	**0.89%**	2.91%	6.57%	**0.07%**	1.08%	1.79%	20.6	20.6	19.9	**0.16**	**0.53**	**1.13**
baseline w/ TTA,IK	-0.24	-0.63	-1.73	6.28%	7.70%	11.3%	1.49%	1.80%	2.53%	20.6	20.6	20.0	0.23	0.56	1.16
baseline w/ TTA,IK,CRP	-0.06	**-0.04**	**-0.08**	1.48%	**0.73%**	**0.80%**	0.35%	**0.16%**	**0.16%**	**20.5**	**20.5**	**19.8**	0.34	0.68	1.27

Table 1: Evaluation of our Skeleton2Humanoid *i.e.* baseline w/ TTA,IK,CRP on Noobjects LaFAN1 dataset. "TTA", "IK" and "CRP" represent the test time adaptation, analytical inverse kinematics and curriculum residual force control policy respectively.

motions such as crawling. We provide more qualitative results in supplementary material for demonstrating the physical plausibility of our synthesized humanoid motions.

4.2 Evaluation of Each Stage

In this section, we concretely analyze the effect of each stage in our Skeleton2Humanoid system.

4.2.1 Evaluation of Test Time Motion In-betweening Network Adaptation Stage. We evaluate our test time adaptation stage on the test set of the whole LaFAN1 dataset [1]. Since no domain generalization approach or test time adaptation approach has been proposed for motion synthesis tasks, we compare against the state-of-the-art motion in-betweening approaches including Kaufmann et al. [31], duan et al. [47] and our implemented baseline Harvey et al. [1]. We select L2Q, L2P, FP, FQ, JQ, SM as evaluation metrics.

Results and discussion The results are shown in Table 3 and Table 2. First, As shown in Table 2, we observe that our test time adaptation stage (TTA) outperforms the baseline approach regarding all physics-based metrics. It shows that TTA can help reduce the physical artifacts and produce more physically-plausible motions for the humanoid character to imitate. Second, we can see that with

our test time adaptation stage, our adapted motion in-betweening network significantly outperforms comparison approaches regarding all accuracy-based metrics in Table 3. We also achieve a new state-of-the-art performance in the L2Q and the L2P metrics. The ablation (w/o ablation, i.e. w/o smooth loss) shows that the motion smooth loss is also useful for test time adaptation.

4.2.2 Evaluation of Skeleton to Humanoid Matching Stage. To evaluate our analytical inverse kinematics (IK) method, we report the matching performance on the synthesized in-betweening motions (transition lengths: 30 frames) by TTA on the test set of Noobject LaFAN1 dataset by computing the mean per joint position error (MPJPE) and inference time. Our analytical inverse kinematics method achieves excellent performance in terms of precision (MPJPE: 11.29mm) and speed (0.122s/sequence). It is worth noting that even our method precisely transfers the skeleton motions to humanoid motions, it still causes a slight performance degradation problem in the L2P metric as shown in table 1 (baseline w/ TTA, IK). This shows that current state-of-the-art motion in-between

	FP			FQ			JQ			SM		
length	5	15	30	5	15	30	5	15	30	5	15	30
baseline [1]*	-0.039	-0.351	-1.186	0.884%	3.49%	7.04%	0.538%	1.02%	1.71%	20.7	20.9	20.6
baseline w/ TTA	**-0.036**	**-0.235**	**-0.930**	**0.775%**	**2.42%**	**5.52%**	**0.527%**	**0.861%**	**1.46%**	**20.5**	**20.5**	**20.0**

Table 2: Evaluation of test time adaptation stage in terms of physics-based metrics on LaFAN1 dataset. "*" means self-implementation (better results than the original paper). "TTA" represents the test time adaptation.

	L2Q			L2P		
length	5	15	30	5	15	30
Zero-Vel	0.56	1.10	1.51	1.52	3.69	6.60
Interp	0.22	0.62	0.98	0.37	1.25	2.32
Kaufmann et al. [31]	0.49	0.60	0.78	0.84	1.07	1.53
duan et al. [47]	0.18	0.37	**0.61**	0.23	0.56	**1.06**
baseline [1]	0.17	0.42	0.69	0.23	0.65	1.28
baseline [1]*	0.138	0.373	0.667	0.183	0.548	1.167
baseline w/ ablation	0.134	0.367	0.662	0.162	0.527	1.142
baseline w/ TTA	**0.133**	**0.367**	0.661	**0.162**	**0.526**	1.140

Table 3: Evaluation of test time adaptation stage in terms of accuracy-based metrics on LaFAN1 dataset. "TTA" represents our test time adaptation. "ablation" represents applying test time adaptation without motion smoothness loss.

Method	L2P (↓)	MPJPE (↓)
UHC [8]	1.307	31.68mm
CRP w/o TwoFrameFeats	1.281	29.16mm
CRP w/o CurForce	1.292	29.95mm
CRP (Ours)	**1.272**	**27.63mm**

Table 4: Evaluation of our curriculum residual force control policy π_{CRP} on Noobjects LaFAN1 dataset. "CurForce" represents applying curriculum learning for residual forces. "TwoFrameFeats" represents utilizing state information from target poses of next two frames

methods can already predict transition motions with very high accuracy, hence a slight difference in poses will sensitively influence the prediction accuracy.

4.2.3 Evaluation of Motion Imitation based on RL Stage. Now we detailedly study the performance of our proposed curriculum residual force control policy π_{CRP}, we evaluate the motion imitation performance on the synthesized humanoid motions with transition lengths of 30 frames by our test time adaptation and skeleton to humanoid matching on the test set of Noobject LaFAN1 dataset. For baseline method, we select universal humanoid controller (UHC) [29] which is a state-of-the-art motion imitation approach based on residual force control [26] for comparison. Since our framework uses different humanoid characters, we use an in-house implementation of UHC [29]. We adopt the same residual force scale and state feature in each frame for fair comparison. All the policies are trained in 1500 iterations and we report the L2P and the MPJPE metrics.

The results are shown in Table 4. First, we can see that our π_{CRP} can imitate the synthesized motion with high accuracy, and significantly outperform the baseline approach. Second, from the ablation study, we can observe that the curriculum learning of residual force
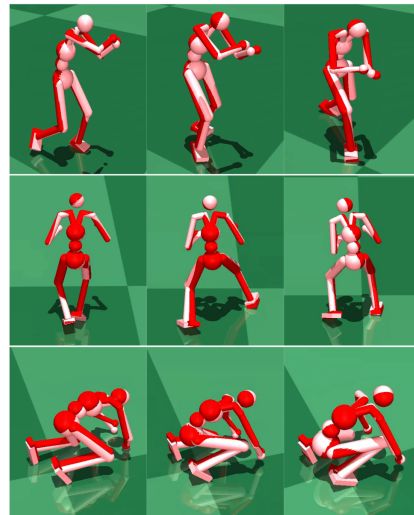
Figure 7: Visualization of the reference motions (red humanoids) and imitated motions (pink humanoids) by our curriculum residual force control policy π_{CRP}.

contributes to better motion imitation accuracy as shown by the corresponding ablations (w/o CurForce and w/o TwoFrameFeats). With the curriculum learning of residual force, the policy can easily learn a optimal control policy with large residual force and gradually learn to control the character precisely while getting rid of the large residual force. We also observe that utilizing the feature of reference poses in next two frames is instrumental, because it extracts informative features from more frames to learn control that advances the character to the next pose.

We also provide qualitative results in Fig. 7, we can see that our curriculum residual force control policy can imitate many complex motions accurately. More qualitative results can be seen in supplementary material.

5 CONCLUSION

In this paper, we proposed a system "Skeleton2Humanoid" which performs physics-oriented motion correction at test time by regularizing synthesized skeleton motions in a physics simulator. Concretely, our system consists of three sequential stages: (I) test time motion synthesis network adaptation, (II) skeleton to humanoid matching and (III) motion imitation based on reinforcement learning (RL). We verified our system on the motion-in-betweening task and showed our method can significantly improve the physical plausibility of synthesized motions. For future work, we plan to transfer the system to other tasks and support more complex human

motions including human-object interactions or human-human interactions.

ACKNOWLEDGMENTS

REFERENCES

[1] F. G. Harvey, M. Yurick, D. Nowrouzezahrai, and C. Pal, "Robust motion in-betweening," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, pp. 60–1, 2020.

[2] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2891–2900.

[3] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-rnn: Deep learning on spatio-temporal graphs," in *Proceedings of the ieee conference on computer vision and pattern recognition*, 2016, pp. 5308–5317.

[4] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4346–4354.

[5] M. Petrovich, M. J. Black, and G. Varol, "Action-conditioned 3d human motion synthesis with transformer vae," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 10 985–10 995.

[6] C. Guo, X. Zuo, S. Wang, S. Zou, Q. Sun, A. Deng, M. Gong, and L. Cheng, "Action2motion: Conditioned generation of 3d human motions," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2021–2029.

[7] P. Yu, Y. Zhao, C. Li, J. Yuan, and C. Chen, "Structure-aware human-action generation," in *European Conference on Computer Vision*. Springer, 2020, pp. 18–34.

[8] Y. Yuan, S.-E. Wei, T. Simon, K. Kitani, and J. Saragih, "Simpoe: Simulated character control for 3d human pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7159–7169.

[9] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM transactions on graphics (TOG)*, vol. 34, no. 6, pp. 1–16, 2015.

[10] Z. Cao, H. Gao, K. Mangalam, Q.-Z. Cai, M. Vo, and J. Malik, "Long-term human motion prediction with scene context," in *European Conference on Computer Vision*. Springer, 2020, pp. 387–404.

[11] W. Mao, M. Liu, M. Salzmann, and H. Li, "Learning trajectory dependencies for human motion prediction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 9489–9497.

[12] L.-Y. Gui, Y.-X. Wang, X. Liang, and J. M. Moura, "Adversarial geometry-aware human motion prediction," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 786–803.

[13] Q. Cui, H. Sun, and F. Yang, "Learning dynamic relationships for 3d human motion prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6519–6527.

[14] E. Barsoum, J. Kender, and Z. Liu, "Hp-gan: Probabilistic 3d human motion prediction via gan," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 1418–1427.

[15] E. Corona, A. Pumarola, G. Alenya, and F. Moreno-Noguer, "Context-aware human motion prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6992–7001.

[16] W. Mao, M. Liu, and M. Salzmann, "History repeats itself: Human motion prediction via motion attention," in *European Conference on Computer Vision*. Springer, 2020, pp. 474–489.

[17] Y. Yuan and K. Kitani, "Dlow: Diversifying latent flows for diverse human motion prediction," in *European Conference on Computer Vision*. Springer, 2020, pp. 346–364.

[18] J. Li, C. Xu, Z. Chen, S. Bian, L. Yang, and C. Lu, "Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3383–3393.

[19] Z. Yu, J. Wang, J. Xu, B. Ni, C. Zhao, M. Wang, and W. Zhang, "Skeleton2mesh: Kinematics prior injected unsupervised human mesh recovery," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 8619–8629.

[20] S. R. Buss and J.-S. Kim, "Selectively damped least squares for inverse kinematics," *Journal of Graphics tools*, vol. 10, no. 3, pp. 37–49, 2005.

[21] C. W. Wampler, "Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16, no. 1, pp. 93–101, 1986.

[22] C. A. Klein and C.-H. Huang, "Review of pseudoinverse control for use with kinematically redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 2, pp. 245–250, 1983.

[23] M. Kallmann, "Analytical inverse kinematics with body posture control," *Computer animation and virtual worlds*, vol. 19, no. 2, pp. 79–91, 2008.

[24] D. Tolani, A. Goswami, and N. I. Badler, "Real-time inverse kinematics techniques for anthropomorphic limbs," *Graphical models*, vol. 62, no. 5, pp. 353–388, 2000.

[25] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 5026–5033.

[26] Y. Yuan and K. Kitani, "Residual force control for agile human behavior imitation and extended motion synthesis," *arXiv preprint arXiv:2006.07364*, 2020.

[27] Z. Yu, J. Wang, J. Xu, B. Ni, C. Zhao, M. Wang, and W. Zhang, "Skeleton2mesh: Kinematics prior injected unsupervised human mesh recovery," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 8619–8629.

[28] J. Tan, K. Liu, and G. Turk, "Stable proportional-derivative controllers," *IEEE Computer Graphics and Applications*, vol. 31, no. 4, pp. 34–44, 2011.

[29] Z. Luo, R. Hachiuma, Y. Yuan, and K. Kitani, "Dynamics-regulated kinematic policy for egocentric pose estimation," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[30] A. Gopalakrishnan, A. Mali, D. Kifer, L. Giles, and A. G. Ororbia, "A neural temporal model for human motion prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 116–12 125.

[31] M. Kaufmann, E. Aksan, J. Song, F. Pece, R. Ziegler, and O. Hilliges, "Convolutional autoencoders for human motion infilling," in *2020 International Conference on 3D Vision (3DV)*. IEEE, 2020, pp. 918–927.

[32] Y. Sun, X. Wang, Z. Liu, J. Miller, A. Efros, and M. Hardt, "Test-time training with self-supervision for generalization under distribution shifts," in *International Conference on Machine Learning*. PMLR, 2020, pp. 9229–9248.

[33] P. Pandey, M. Raman, S. Varambally, and P. Ap, "Generalization on unseen domains via inference-time label-preserving target projections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 924–12 933.

[34] Y. Li, M. Hao, Z. Di, N. B. Gundavarapu, and X. Wang, "Test-time personalization with a transformer for human pose estimation," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[35] S. Guan, J. Xu, Y. Wang, B. Ni, and X. Yang, "Bilevel online adaptation for out-of-domain human mesh reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 472–10 481.

[36] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," *arXiv preprint arXiv:2006.10726*, 2020.

[37] L. Liu and J. Hodgins, "Learning to schedule control fragments for physics-based characters using deep q-learning," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 3, pp. 1–14, 2017.

[38] ——, "Learning basketball dribbling skills using trajectory optimization and deep reinforcement learning," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.

[39] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne, "Deepmimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.

[40] X. B. Peng, A. Kanazawa, J. Malik, P. Abbeel, and S. Levine, "Sfv: Reinforcement learning of physical skills from videos," *ACM Transactions On Graphics (TOG)*, vol. 37, no. 6, pp. 1–14, 2018.

[41] J. Ho and S. Ermon, "Generative adversarial imitation learning," *Advances in neural information processing systems*, vol. 29, 2016.

[42] Y. Yuan and K. Kitani, "3d ego-pose estimation via imitation learning," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 735–750.

[43] ——, "Ego-pose estimation and forecasting as real-time pd control," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 10 082–10 092.

[44] K. Bergamin, S. Clavet, D. Holden, and J. R. Forbes, "Drecon: data-driven responsive control of physics-based characters," *ACM Transactions On Graphics (TOG)*, vol. 38, no. 6, pp. 1–11, 2019.

[45] S. Park, H. Ryu, S. Lee, S. Lee, and J. Lee, "Learning predict-and-simulate policies from unorganized human motion data," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–11, 2019.

[46] J. Won, D. Gopinath, and J. Hodgins, "A scalable approach to control diverse behaviors for physically simulated characters," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, pp. 33–1, 2020.

[47] Y. Duan, Y. Lin, Z. Zou, Y. Yuan, Z. Qian, and B. Zhang, "A unified framework for real time motion completion," 2022.

[48] L. Kovar, M. Gleicher, and F. Pighin, "Motion graphs," in *ACM SIGGRAPH 2008 classes*, 2008, pp. 1–10.

[49] J. Min, Y.-L. Chen, and J. Chai, "Interactive generation of human animation with deformable motion models," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 1, pp. 1–12, 2009.

[50] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[51] K. Gong, B. Li, J. Zhang, T. Wang, J. Huang, M. B. Mi, J. Feng, and X. Wang, "Posetriplet: Co-evolving 3d human pose estimation, imitation, and hallucination under self-supervision," *arXiv preprint arXiv:2203.15625*, 2022.

[52] T. Wang, Y. Guo, M. Shugrina, and S. Fidler, "Unicon: Universal neural controller for physics-based character motion," *arXiv preprint arXiv:2011.15119*, 2020.

[53] J. Butepage, M. J. Black, D. Kragic, and H. Kjellstrom, "Deep representation learning for human motion prediction and classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6158–6166.

[54] Z. Li, Y. Zhou, S. Xiao, C. He, Z. Huang, and H. Li, "Auto-conditioned recurrent networks for extended complex human motion synthesis," *arXiv preprint arXiv:1707.05363*, 2017.

[55] Y. Zhang, M. J. Black, and S. Tang, "We are more than our joints: Predicting how 3d bodies move," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3372–3382.

[56] Y. Cai, Y. Wang, Y. Zhu, T.-J. Cham, J. Cai, J. Yuan, J. Liu, C. Zheng, S. Yan, H. Ding *et al.*, "A unified 3d human motion synthesis model via conditional variational auto-encoder," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 645–11 655.

[57] Y. Zhou, J. Lu, C. Barnes, J. Yang, S. Xiang *et al.*, "Generative tweening: Long-term inbetweening of 3d human motions," *arXiv preprint arXiv:2005.08891*, 2020.

[58] J. Wang, H. Xu, J. Xu, S. Liu, and X. Wang, "Synthesizing long-term 3d human motion and interaction in 3d scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9401–9411.

[59] A. Hernandez, J. Gall, and F. Moreno-Noguer, "Human motion prediction via spatio-temporal inpainting," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7134–7143.

[60] E. Aksan, M. Kaufmann, and O. Hilliges, "Structured prediction helps 3d human motion modelling," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7144–7153.

[61] M. Li, S. Chen, Y. Zhao, Y. Zhang, Y. Wang, and Q. Tian, "Dynamic multiscale graph neural networks for 3d skeleton based human motion prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 214–223.

[62] S. Yan, Z. Li, Y. Xiong, H. Yan, and D. Lin, "Convolutional sequence generation for skeleton-based action synthesis," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4394–4402.

[63] H. Cai, C. Bai, Y.-W. Tai, and C.-K. Tang, "Deep video generation, prediction and completion of human action sequences," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 366–382.

[64] D. Rempe, T. Birdal, A. Hertzmann, J. Yang, S. Sridhar, and L. J. Guibas, "Humor: 3d human motion model for robust pose estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 488–11 499.

[65] Z. Song, D. Wang, N. Jiang, Z. Fang, C. Ding, W. Gan, and W. Wu, "Actformer: A gan transformer framework towards general action-conditioned 3d human motion generation," *arXiv preprint arXiv:2203.07706*, 2022.

[66] M. Petrovich, M. J. Black, and G. Varol, "Action-conditioned 3d human motion synthesis with transformer vae," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 985–10 995.

[67] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

A ANALYTICAL INVERSE KINEMATICS

Here we concretely introduce our analytical inverse kinematics approach used in skeleton to humanoid matching stage, which is able to convert human skeleton motions to humanoid motions even when the body structure is different from the human skeleton.

We first construct a humanoid model whose body structure (bone length, joint number) is identical to the skeleton except for the kinematics tree. Concretely, the bone length and the joint number of the humanoid and the skeleton are the same, but the initial pose of the humanoid and the skeleton are different and each joint of the humanoid and the skeleton rotates in a different order. We visualize the humanoid model and the skeleton in Fig. 8, each model contains 20 joints, and we only calculate the rotations of 15 joints from the positions of all 20 joints. We describe the detailed matching process for 15 calculated joints:

(1) left elbow:

$$
\begin{cases}
[\mathrm{x}_p, \mathrm{y}_p, \mathrm{z}_p] = \left[\dfrac{l_{ls-le}}{||l_{ls-le}|}, \dfrac{\mathrm{z}_p \otimes \mathrm{x}_p}{|\mathrm{z}_p \otimes \mathrm{x}_p|}, \dfrac{-l_{ls-le} \otimes l_{le-lw}}{||l_{ls-le} \otimes l_{le-lw}|}\right] \\
[\mathrm{x}_c, \mathrm{y}_c, \mathrm{z}_c] = \left[\dfrac{l_{le-lw}}{||l_{le-lw}|}, \dfrac{\mathrm{z}_c \otimes \mathrm{x}_c}{|\mathrm{z}_c \otimes \mathrm{x}_c|}, \mathrm{z}_p\right]
\end{cases}, \quad (10)
$$

(2) left shoulder:

$$
\begin{cases}
[\mathrm{x}_c, \mathrm{y}_c, \mathrm{z}_c] = \left[\dfrac{l_{ls-le}}{||l_{ls-le}|}, \dfrac{\mathrm{z}_c \otimes \mathrm{x}_c}{|\mathrm{z}_c \otimes \mathrm{x}_c|}, \dfrac{-l_{ls-le} \otimes l_{le-lw}}{||l_{ls-le} \otimes l_{le-lw}|}\right] \\
[\mathrm{x}_p, \mathrm{y}_p, \mathrm{z}_p] = \left[\dfrac{l_{lcl-ls}}{||l_{lcl-ls}|}, \dfrac{\mathrm{z}_p \otimes \mathrm{x}_p}{|\mathrm{z}_p \otimes \mathrm{x}_p|}, \dfrac{l_{lcl-mspine} \otimes \mathrm{x}_p}{|l_{lcl-mspine} \otimes \mathrm{x}_p \mathrm{x}_p|}\right]
\end{cases},
$$
$$(11)$$

(3) left clavicle:

$$
\begin{cases}
[\mathrm{x}_c, \mathrm{y}_c, \mathrm{z}_c] = \left[\dfrac{l_{lcl-ls}}{||l_{lcl-ls}|}, \dfrac{\mathrm{z}_c \otimes \mathrm{x}_c}{|\mathrm{z}_c \otimes \mathrm{x}_c|}, \dfrac{l_{lcl-mspine} \otimes \mathrm{x}_c}{|l_{lcl-mspine} \otimes \mathrm{x}_c \mathrm{x}_c|}\right] \\
[\mathrm{x}_p, \mathrm{y}_p, \mathrm{z}_p] = \left[\dfrac{l_{rcl-lcl}}{||l_{rcl-lcl}|}, \dfrac{\mathrm{z}_p \otimes \mathrm{x}_p}{|\mathrm{z}_p \otimes \mathrm{x}_p|}, \dfrac{l_{lownec-lcl} \otimes l_{lownec-rcl}}{||l_{lownec-lcl} \otimes l_{lownec-rcl}|}\right]
\end{cases},
$$
$$(12)$$

(4) right elbow:

$$
\begin{cases}
[\mathrm{x}_c, \mathrm{y}_c, \mathrm{z}_c] = \left[\dfrac{l_{rw-re}}{||l_{rw-re}|}, \dfrac{\mathrm{z}_c \otimes \mathrm{x}_c}{|\mathrm{z}_c \otimes \mathrm{x}_c|}, \dfrac{l_{re-rw} \otimes l_{re-rs}}{||l_{re-rw} \otimes l_{re-rs}|}\right] \\
[\mathrm{x}_p, \mathrm{y}_p, \mathrm{z}_p] = \left[\dfrac{l_{re-rs}}{||l_{re-rs}|}, \dfrac{\mathrm{z}_p \otimes \mathrm{x}_p}{|\mathrm{z}_p \otimes \mathrm{x}_p|}, \mathrm{z}_c\right]
\end{cases}, \quad (13)
$$

(5) right shoulder:

$$
\begin{cases}
[\mathrm{x}_c, \mathrm{y}_c, \mathrm{z}_c] = \left[\dfrac{l_{re-rs}}{||l_{re-rs}|}, \dfrac{\mathrm{z}_c \otimes \mathrm{x}_c}{|\mathrm{z}_c \otimes \mathrm{x}_c|}, \dfrac{l_{re-rw} \otimes l_{re-rs}}{||l_{re-rw} \otimes l_{re-rs}|}\right] \\
[\mathrm{x}_p, \mathrm{y}_p, \mathrm{z}_p] = \left[\dfrac{l_{rs-rcl}}{||l_{rs-rcl}|}, \dfrac{\mathrm{z}_p \otimes \mathrm{x}_p}{|\mathrm{z}_p \otimes \mathrm{x}_p|}, \dfrac{l_{rcl-mspine} \otimes \mathrm{x}_p}{|l_{rcl-mspine} \otimes \mathrm{x}_p|}\right]
\end{cases}, \quad (14)
$$

(6) right clavicle:

$$
\begin{cases}
[\mathrm{x}_c, \mathrm{y}_c, \mathrm{z}_c] = \left[\dfrac{l_{rs-rcl}}{||l_{rs-rcl}|}, \dfrac{\mathrm{z}_c \otimes \mathrm{x}_c}{|\mathrm{z}_c \otimes \mathrm{x}_c|}, \dfrac{l_{rcl-mspine} \otimes \mathrm{x}_c}{|l_{rcl-mspine} \otimes \mathrm{x}_c|}\right] \\
[\mathrm{x}_p, \mathrm{y}_p, \mathrm{z}_p] = \left[\dfrac{l_{rcl-lcl}}{||l_{rcl-lcl}|}, \dfrac{\mathrm{z}_p \otimes \mathrm{x}_p}{|\mathrm{z}_p \otimes \mathrm{x}_p|}, \dfrac{l_{lownec-lcl} \otimes l_{lownec-rcl}}{||l_{lownec-lcl} \otimes l_{lownec-rcl}|}\right]
\end{cases},
$$
$$(15)$$

(7) lower neck

$$
\begin{cases}
[\mathrm{x}_c, \mathrm{y}_c, \mathrm{z}_c] = \left[\dfrac{l_{rcl-lcl}}{||l_{rcl-lcl}|}, \dfrac{\mathrm{z}_c \otimes \mathrm{x}_c}{|\mathrm{z}_c \otimes \mathrm{x}_c|}, \dfrac{l_{lownec-lcl} \otimes l_{lownec-rcl}}{||l_{lownec-lcl} \otimes l_{lownec-rcl}|}\right] \\
[\mathrm{x}_p, \mathrm{y}_p, \mathrm{z}_p] = \left[\dfrac{\mathrm{y}_p \otimes l_{th-mspine}}{|\mathrm{y}_p \otimes l_{th-mspine}|}, \dfrac{l_{th-lownec}}{||l_{th-lownec}|}, \dfrac{\mathrm{x}_p \otimes \mathrm{y}_p}{|\mathrm{x}_p \otimes \mathrm{y}_p|}\right]
\end{cases},
$$
$$(16)$$

(8) thorax

$$
\begin{cases}
[\mathrm{x}_c, \mathrm{y}_c, \mathrm{z}_c] = \left[\dfrac{\mathrm{y}_c \otimes l_{th-mspine}}{|\mathrm{y}_c \otimes l_{th-mspine}|}, \dfrac{l_{th-lownec}}{||l_{th-lownec}|}, \dfrac{\mathrm{x}_c \otimes \mathrm{y}_c}{|\mathrm{x}_c \otimes \mathrm{y}_c|}\right] \\
[\mathrm{x}_p, \mathrm{y}_p, \mathrm{z}_p] = \left[\dfrac{\mathrm{y}_p \otimes l_{mspine-root}}{|\mathrm{y}_p \otimes l_{mspine-root}|}, \dfrac{l_{mspine-th}}{||l_{mspine-th}|}, \dfrac{\mathrm{x}_p \otimes \mathrm{y}_p}{|\mathrm{x}_p \otimes \mathrm{y}_p|}\right]
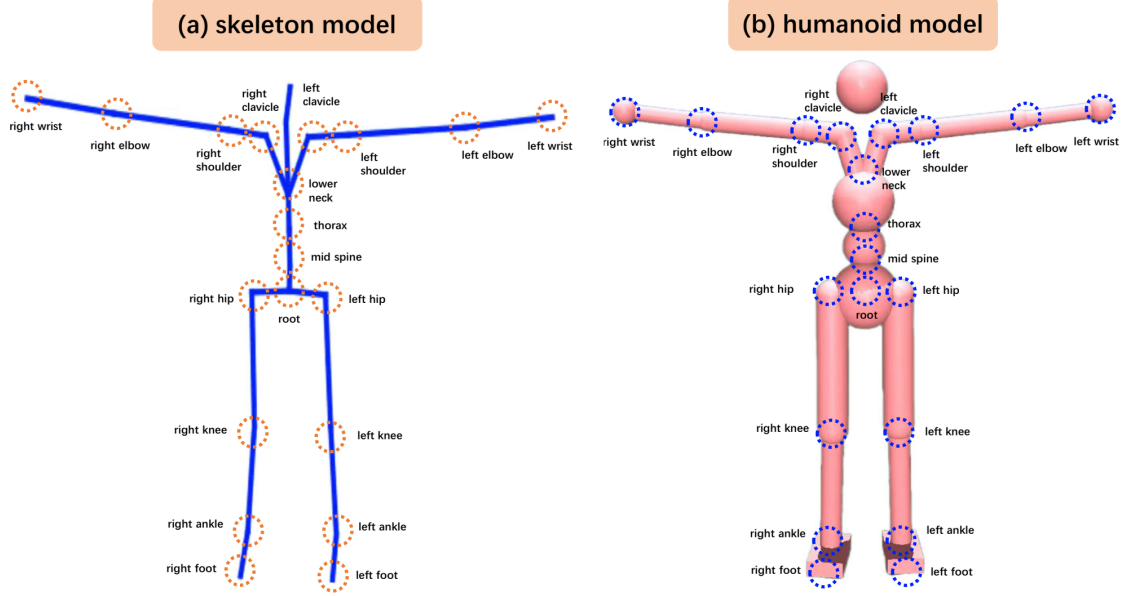\end{cases},
$$
$$(17)$$

Figure 8: Visualization of the skeleton model (a) and the humanoid model (b). Each model contains 20 joints.

(9) mid spine

$$
\begin{cases}
[x_c, y_c, z_c] = \left[\dfrac{y_c \otimes l_{mspine-root}}{|y_c \otimes l_{mspine-root}|}, \dfrac{l_{mspine-th}}{|l_{mspine-th}|}, \dfrac{x_c \otimes y_c}{|x_c \otimes y_c|} \right] \\[3mm]
[x_p, y_p, z_p] = \left[\dfrac{l_{rhip-lhip}}{|l_{rhip-lhip}|}, \dfrac{l_{root-rhip} \otimes l_{root-lhip}}{|l_{root-rhip} \otimes l_{root-lhip}|}, \dfrac{x_p \otimes y_p}{|x_p \otimes y_p|} \right]
\end{cases},
\tag{18}
$$

(10) left hip

$$
\begin{cases}
[x_c, y_c, z_c] = \left[\dfrac{y_c \otimes z_c}{|y_c \otimes z_c|}, \dfrac{l_{lk-lhip}}{|l_{lk-lhip}|}, \dfrac{l_{lhip-rhip} \otimes -y_c}{|l_{lhip-rhip} \otimes -y_c|} \right] \\[3mm]
[x_p, y_p, z_p] = \left[\dfrac{l_{rhip-lhip}}{|l_{rhip-lhip}|}, \dfrac{l_{root-rhip} \otimes l_{root-lhip}}{|l_{root-rhip} \otimes l_{root-lhip}|}, \dfrac{x_p \otimes y_p}{|x_p \otimes y_p|} \right]
\end{cases},
\tag{19}
$$

(11) left knee

$$
\begin{cases}
[x_c, y_c, z_c] = \left[\dfrac{-y_c \otimes l_{lk-lhip}}{|-y_c \otimes l_{lk-lhip}|}, \dfrac{l_{lan-lk}}{|l_{lan-lk}|}, \dfrac{x_c \otimes y_c}{|x_c \otimes y_c|} \right] \\[3mm]
[x_p, y_p, z_p] = \left[\dfrac{y_c \otimes z_c}{|y_c \otimes z_c|}, \dfrac{l_{lk-lhip}}{|l_{lk-lhip}|}, \dfrac{l_{lhip-rhip} \otimes -y_c}{|l_{lhip-rhip} \otimes -y_c|} \right]
\end{cases},
\tag{20}
$$

(12) left ankle:

$$
\begin{cases}
[x_c, y_c, z_c] = \left[\dfrac{y_c \otimes z_c}{|y_c \otimes z_c|}, \dfrac{l_{lan-lk}}{|l_{lan-lk}|}, \dfrac{l_{lan-lf}}{|l_{lan-lf}|} \right] \\[3mm]
[x_p, y_p, z_p] = \left[\dfrac{-y_p \otimes l_{lk-lhip}}{|-y_p \otimes l_{lk-lhip}|}, \dfrac{l_{lan-lk}}{|l_{lan-lk}|}, \dfrac{x_p \otimes y_p}{|x_p \otimes y_p|} \right]
\end{cases},
\tag{21}
$$

(13) right hip

$$
\begin{cases}
[x_c, y_c, z_c] = \left[\dfrac{y_c \otimes z_c}{|y_c \otimes z_c|}, \dfrac{l_{rk-rhip}}{|l_{rk-rhip}|}, \dfrac{-y_c \otimes l_{rhip-lhip}}{|-y_c \otimes l_{rhip-lhip}|} \right] \\[3mm]
[x_p, y_p, z_p] = \left[\dfrac{l_{rhip-lhip}}{|l_{rhip-lhip}|}, \dfrac{l_{root-rhip} \otimes l_{root-lhip}}{|l_{root-rhip} \otimes l_{root-lhip}|}, \dfrac{x_p \otimes y_p}{|x_p \otimes y_p|} \right]
\end{cases},
\tag{22}
$$

(14) right knee

$$
\begin{cases}
[x_c, y_c, z_c] = \left[\dfrac{-y_c \otimes l_{rk-rhip}}{|-y_c \otimes l_{rk-rhip}|}, \dfrac{l_{ran-rk}}{|l_{ran-rk}|}, \dfrac{x_c \otimes y_c}{|x_c \otimes y_c|} \right] \\[3mm]
[x_p, y_p, z_p] = \left[\dfrac{y_p \otimes z_p}{|y_p \otimes z_p|}, \dfrac{l_{rk-rhip}}{|l_{rk-rhip}|}, \dfrac{-y_p \otimes l_{rhip-lhip}}{|-y_p \otimes l_{rhip-lhip}|} \right]
\end{cases},
\tag{23}
$$

(15) right ankle

$$
\begin{cases}
[x_c, y_c, z_c] = \left[\dfrac{y_c \otimes z_c}{|y_c \otimes z_c|}, \dfrac{l_{ran-rk}}{|l_{ran-rk}|}, \dfrac{l_{ran-rf}}{|l_{ran-rf}|} \right] \\[3mm]
[x_p, y_p, z_p] = \left[\dfrac{-y_p \otimes l_{rk-rhip}}{|-y_p \otimes l_{rk-rhip}|}, \dfrac{l_{ran-rk}}{|l_{ran-rk}|}, \dfrac{x_p \otimes y_p}{|x_p \otimes y_p|} \right]
\end{cases},
\tag{24}
$$

where ls, le, lw, lcl, $mspine$, $lownec$, th, $root$, $lhip$, lk, lan and lf represent left shoulder, left elbow, left wrist, left clavicle, mid spine, lower neck, thorax, root, left hip, left knee, left ankle and left foot.

B IMPLEMENTATION DETAILS

B.1 Network Architecture

Our motion in-betweening network is the same as Harvey et al. [1], which is a popular motion in-betweening model based on recurrent neural network. Our curriculum residual force control policy π_{CRP}
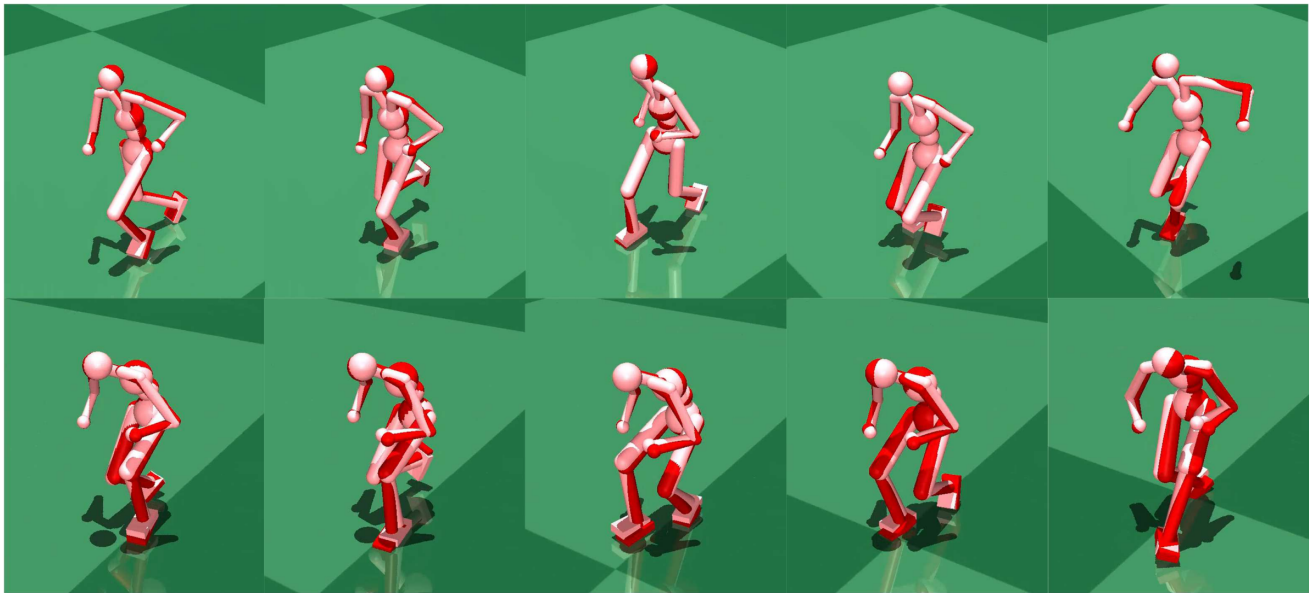
Figure 9: Qualitative results of the reference motions (red humanoids) and imitated motions (pink humanoids) by our curriculum residual force control policy π_{CRP}.

is a multi-layer preceptron (MLP) network with 3 hidden layers whose size is (1024,512,256).

B.2 Experiments Details

We implement our whole system on PyTorch [67]. The three stages in our Skeleton2Humanoid system are implemented in a sequential manner. The motion in-betweening network is trained following [1] on the train set of the LaFAN1/Noobjects LaFAN1 dataset with various transition lengths (from 5 frames to 30 frames) between past keyframes and a future keyframe, then the test time adaptation stage individually optimizes the pretrained motion in-betweening network on different test sets whose transition lengths of skeleton motions are 5 frames, 15 frames and 30 frames. The skeleton to humanoid matching stage converts human skeleton motions synthesized from the test sets to humanoid motions. Our curriculum residual force control policy π_{CRP} is trained on the converted humanoid motions from the test sets. We severally train a policy for the converted humanoid motions with different transition lengths (5 frames, 15 frames and 30 frames).

We utilize Mujoco as our physics simulator. During RL training, we apply the standard reinforcement learning algorithm PPO. The policy network is a typical 3-layer MLP network with size (1024, 512, 256). We train the curriculum residual force control policy for 1500 iterations. For each iteration, we keep collecting data by sampling RL episodes until the total number of time steps reaches 50000. For each RL episode, the RL agent explores to act and imitate a randomly sampled matched humanoid motion sequence and creates corresponding rewards. The RL episode is terminated when the end frame is reached or the humanoid's root height is 0.5 below the root height of the reference pose.

	L2P			L2Q		
length	5	15	30	5	15	30
baseline[1]	0.183	0.548	1.167	0.138	0.373	0.667
iteration 1	0.171	0.538	1.153	0.135	0.370	0.663
iteration 2	0.167	0.532	1.145	0.134	0.368	0.661
iteration 3	0.164	0.529	1.141	0.134	0.367	0.661
iteration 4	0.163	0.527	1.140	0.133	0.367	0.661
iteration 5	0.162	0.526	1.142	0.133	0.367	0.663
iteration 6	0.162	0.526	1.146	0.133	0.367	0.666
iteration 7	0.162	0.527	1.151	0.133	0.367	0.669

Table 5: performance gains from test time adaptation and the number of iterations.

C QUALITATIVE EVALUATION

C.1 Qualitative Results of Skeleton2Humanoid System

We provide more visualization results of our Skeleton2humanoid System including ground truth skeleton motions, predicted skeleton motions, converted humanoid motions and imitated humanoid motions in attached videos. As shown in our videos, our approach can synthesize various physically-plausible humanoid motions.

C.2 Qualitative Results of Our Curriculum Residual Force Control Policy

In this section, we provide more qualitative results of our curriculum residual force policy π_{CRP} (CRP) which are shown in Fig. 9. we can see that our curriculum residual force control policy can imitate many complex motions accurately.

	wall time		
length	5	15	30
baseline[1]	8.30ms	8.85ms	9.88ms
iteration 1	0.91ms	1.75ms	3.08ms
iteration 2	1.83ms	3.56ms	6.18ms
iteration 3	2.75ms	5.39ms	9.30ms
iteration 4	3.68ms	7.18ms	12.43ms
iteration 5	4.61ms	8.97ms	15.54ms
iteration 6	5.54ms	10.78ms	18.70ms
iteration 7	6.46ms	12.57ms	21.83ms

Table 6: The relation between the inference/wall time (per sequence) and the number of iterations.

C.3 inference time penalty of the TTA stage

We provide the trade-off results between the extra performance gains from TTA and the number of iterations as suggested by the reviewer in table. 5 and table. 6. The result is obtained on the LaFAN1 dataset (about 2245 sequences) with various transition lengths (5 frames, 15 frames, 30 frames) using a single 3090 GPU.