SRBTrack: Terrain-Adaptive Tracking of a Single-Rigid-Body Character Using Momentum-Mapped Space-Time Optimization

HANYANG CAO[*], Hanyang University, Republic of Korea
HEYUAN YAO[*], Peking University, China
LIBIN LIU, Peking University, China
TAESOO KWON[†], Hanyang University, Republic of Korea

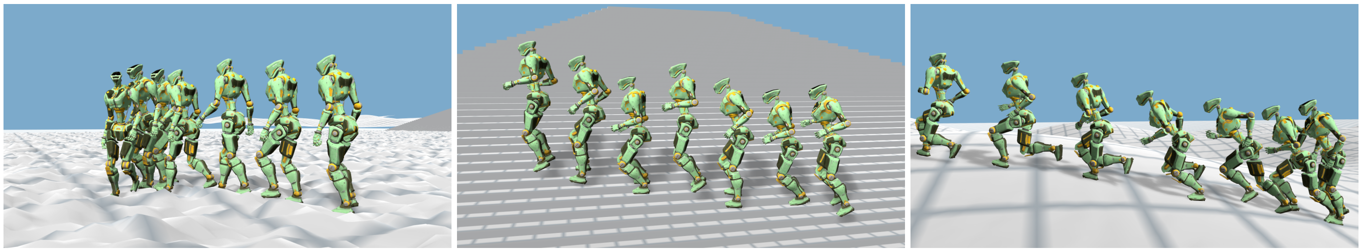


Fig. 1. We present SRBTrack, a general controller that enables a Single-Rigid-Body character to generate diverse full-body motion adaptive to versatile environments. Our framework can be used for a wide range of applications, including plug-and-play motion tracking, robust adaptation to unseen terrains and perturbations, and generalizable full-body motion synthesis.

Generating realistic and robust motion for virtual characters under complex physical conditions, such as irregular terrain, real-time control scenarios, and external disturbances, remains a key challenge in computer graphics. While deep reinforcement learning has enabled high-fidelity physics-based character animation, such methods often suffer from limited generalizability, as learned controllers tend to overfit to the environments they were trained in. In contrast, simplified models, such as single rigid bodies, offer better adaptability, but traditionally require hand-crafted heuristics and can only handle short motion segments. In this paper, we present a general learning framework that trains a single-rigid-body (SRB) character controller from long and unstructured datasets, without the reliance on human-crafted rules. Our method enables zero-shot adaptation to diverse environments and unseen motion styles. The resulting controller generates expressive and physically plausible motions in real time and seamlessly integrates with high-level kinematic motion planners without retraining, enabling a wide range of downstream tasks.

CCS Concepts: • **Computing methodologies** → **Animation**; **Physical simulation**; *Reinforcement learning*.

Additional Key Words and Phrases: physics-based character animation, motion control, deep reinforcement learning, simplified model

[*] Equal contribution.
[†] Corresponding author.

Authors' Contact Information: Hanyang Cao, Department of Computer Science, Hanyang University, Seoul, Republic of Korea, caohanyang@hanyang.ac.kr; Heyuan Yao, School of Computer Science, Peking University, Beijing, China, heyuanyao@pku.edu.cn; Libin Liu, State Key Laboratory of General Artificial Intelligence, Peking University, Beijing, China, libin.liu@pku.edu.cn; Taesoo Kwon, Department of Computer Science, Hanyang University, Seoul, Republic of Korea, taesoo@hanyang.ac.kr.

SA Conference Papers '25, Hong Kong, Hong Kong
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2137-3/25/12
https://doi.org/10.1145/3757377.3763823

ACM Reference Format:
Hanyang Cao, Heyuan Yao, Libin Liu, and Taesoo Kwon. 2025. SRBTrack: Terrain-Adaptive Tracking of a Single-Rigid-Body Character Using Momentum-Mapped Space-Time Optimization. In *SIGGRAPH Asia 2025 Conference Papers (SA Conference Papers '25), December 15–18, 2025, Hong Kong, Hong Kong*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3757377.3763823

1 Introduction

Generating physically realistic motion for virtual characters under challenging conditions remains a core problem in computer graphics—particularly for interactive character animation in games and AR/VR applications. These conditions include irregular terrain, real-time interactions between users, characters, and the environment, as well as unpredictable external disturbances. In such scenarios, characters must maintain both physical plausibility and motion fidelity across diverse and dynamic settings. The introduction of deep reinforcement learning (DRL) to control simulated characters has enabled a wide range of adaptive and physically accurate motions. Physics-based methods excel at producing motions that are consistent with physical laws. They enable characters to interact with varied scenes, manipulate objects, and withstand external disturbances—achieving realistic contacts and fluid interactions beyond the reach of kinematic approaches. However, this realism comes at a cost: character controllers trained with DRL tend to be tightly coupled to the specific environments in which they were developed. For instance, a full-body controller trained only on flat terrain often fails to generalize to uneven surfaces or unexpected perturbations without further environment-specific fine-tuning [Rempe et al. 2023; Tessler et al. 2024; Won et al. 2022a]. This limits the practical deployment of such methods in games and other real-world scenarios.

An alternative and orthogonal approach to full-body simulation is the use of simplified models. By abstracting the full body dynamics into a reduced representation such as a single rigid body model [Kwon et al. 2023; Winkler et al. 2018], it becomes possible to capture key physical characteristics of character movements, including contact forces and the resulting momentum. This abstraction also enables the development of learned policies that can adapt to various unobserved environmental changes and controller transitions without requiring additional training.

Despite the many advantages of simplified model methods—such as robustness and sample-efficient learning—previous approaches have often depended heavily on human expertise. For instance, they typically require manually tuned stride parameters or rely on foot contact information extracted from motion capture data. Furthermore, these methods are generally limited to learning relatively short motion clips, which significantly constrains their expressive capabilities. To better navigate these trade-offs, we complement physics-based full-body methods by developing a controller for a simplified model in an auto-regressive manner, yielding significant improvements in robustness, sample efficiency, generalizability, and training speed.

In this paper, we present a general reinforcement learning (RL) framework for acquiring robust and user-friendly locomotion strategies for a single-rigid-body (SRB) character model, trained directly from long and unstructured datasets. Unlike prior methods, our approach employs a fully data-driven learning process for footstep planning, enabling the model to predict future states based solely on its current state. This eliminates the need for human intervention and removes dependence on full-body motion capture data during inference.

Our framework is capable of learning long-term, unified motion skills rather than being limited to short-term, isolated motion clips. This flexibility allows for seamless integration with various motion generation techniques for downstream tasks. To reconstruct high-quality, dynamically realistic full-body motions, we also introduce a novel space-time optimization method that leverages dynamic properties of the SRB model, such as momentum and center of mass (COM) position.

Experimental results demonstrate that our trained controller consistently generates high-quality motions across diverse terrain and interaction scenarios without requiring additional training. Moreover, it can accurately and smoothly track the output of a kinematic motion generator, producing motions that are both physically plausible and highly dynamic. All capabilities are achieved using a single policy trained without any prior task-specific information. These results underscore the potential of our approach for applications in gaming and virtual reality.

In summary, our main contributions are as follows:

- We propose an SRB controller that learns locomotion skills from unstructured data without relying on heuristic rules or fixed foot planning, enabling zero-shot adaptation to diverse environments and previously unseen motions.
- We introduce a real-time full-body motion generator that enables physically realistic responses, while effectively eliminating foot skating and noise.
- Our approach generalizes well across multiple downstream tasks, and produces high-quality motions.

2 Related Work

We review prior work in character animation and control, focusing on two key categories relevant to our approach: physics-based motion controllers and physics-informed motion generators. These research directions have significantly advanced the development of physically plausible, generalizable, and natural character animation.

2.1 Physics-Based Motion Control

To ensure the physical realism of generated motions, a straightforward approach is to construct a full physical model of the character and control its behavior through physical simulation. Early efforts relied on rule-based controllers to achieve stable locomotion [Coros et al. 2008; Lee et al. 2010; Yin et al. 2007]. Over time, techniques such as spacetime optimization [Mordatch et al. 2012; Witkin and Kass 1988], model predictive control [Hämäläinen et al. 2015; Liu et al. 2010; Macchietto et al. 2009; Mordatch et al. 2010; Tassa et al. 2012], and reinforcement learning [Levine and Koltun 2013; Yin et al. 2021] were introduced to support increasingly complex movements. However, these methods often depend on carefully crafted objective functions and extensive parameter tuning, which can limit their practicality for complex scenarios and general-purpose motion generation.

With the advent of deep reinforcement learning (DRL), researchers have demonstrated that simulated characters can effectively imitate motion capture data [Peng et al. 2018a, 2017], often with only simple reward function designs. By employing pose reconstruction rewards [Fussell et al. 2021; Peng et al. 2018a; Won et al. 2020], or leveraging techniques such as behavior cloning [Luo et al. 2023; Truong et al. 2024; Won et al. 2022b; Wu et al. 2025] and Generative Adversarial Imitation Learning (GAIL) [Ho and Ermon 2016; Xu and Karamouzas 2021; Xu et al. 2023], characters can exhibit fluid and stylized locomotion [Bergamin et al. 2019; Peng et al. 2022, 2018b; Tessler et al. 2024, 2023; Yao et al. 2022, 2024], as well as complex human-object interactions [Hassan et al. 2023; Kim et al. 2025; Lee et al. 2022; Wu et al. 2024], and human-scene interactions [Pan et al. 2025; Rempe et al. 2023; Tevet et al. 2024].

While the motions generated by these methods adhere to physical principles, the high degrees of freedom inherent in full-body movements often require several days of training. Moreover, policies trained in fixed environments generally struggle to transfer directly to novel scenarios or terrains. To address this limitation, some studies have introduced adaptation mechanisms, such as auxiliary helper branches, to fine-tune the policy for new environments [Won et al. 2022b; Xu et al. 2023]. However, these methods still require additional environment-specific training.

Although recent universal controllers [He et al. 2024; Tessler et al. 2024, 2025] achieve locomotion over varied terrains, complex human–object interactions, and multiple control modes, they typically require large motion datasets, detailed terrain models, and additional training. In contrast, our method is sample-efficient, trained only on flat terrain, and fast to train. Additionally, our approach retains the inherent flexibility of kinematic-based methods to synthesize stylized motions that may deviate from rigid-multi-body constraints while preserving visual plausibility.

2.2 Kinematic and Physics-Informed Motion Generation

An alternative to physics-based character animation is the use of data-driven methods to generate kinematic motion. These approaches span a broad range—from early techniques such as finite state machines [Dosé and Cachelin 2016; Lee et al. 2002; Levine et al. 2011; Safonova and Hodgins 2007; van de Panne 2014] and statistical models [Chen et al. 2009; Levine et al. 2012; Min et al. 2009], to more recent neural architectures including Recurrent Neural Networks (RNNs)[Zhang and van de Panne 2018; Zhao et al. 2018], Transformers[Lu et al. 2024; Petrovich et al. 2021; Zhang et al. 2023b; Zhou et al. 2024], and generative models such as VAEs [Ling et al. 2020a; Qin et al. 2022], diffusion models [Alexanderson et al. 2023; Chen et al. 2024; Tevet et al. 2023; Xin et al. 2023; Zhang et al. 2024, 2023a; Zhou et al. 2024], and GANs [Li et al. 2022; Raab et al. 2023]. These methods learn plausible motion distributions from motion capture data and are widely used in various downstream tasks. However, due to the absence of integrated physics simulation, these approaches often require post-processing steps to mitigate artifacts and enable physical interactions. For example, Holden et al. [2020, 2017] use an inverse kinematics solver to address issues such as foot sliding and floating. However, these techniques typically constrain only foot movement. Other researchers incorporates physics-inspired loss functions to impose constraints on the network [Yi et al. 2022; Yuan et al. 2023]. However, they are often designed such that corrective torques can act on the root joint, limiting the model's ability to capture essential physical principles—such as center of mass (COM) dynamics and swing foot behavior—which are critical for producing adaptive motion.

Our method also falls under the category of kinematic motion generators. However, unlike the aforementioned approaches, it integrates accurate physics simulation based on single-rigid-body (SRB) dynamics. Among previous works, the methods proposed by Kwon et al. [2023, 2020] also utilize the SRB character model, making them the most closely related to our approach. This model generates physically plausible full-body motions, where environmental variations—such as external pushes or uneven terrain—naturally influence the character's behavior through SRB simulation, leading to more realistic and adaptive motion. In particular, [Kwon et al. 2023], like our approach, combines online quadratic programming with offline reinforcement learning to achieve real-time performance. However, our approach differs in that it is trained on long, unlabeled motion datasets and employs a single policy capable of tracking a wide range of motions.

3 Overview

As illustrated in Figure 2, our SRBtrack framework consists of two main components: the SRB simulation module (bottom of the figure) and the full-body motion generation module (top of the figure). These two parts are trained independently.

In the SRB simulation module, a motion tracking policy is trained to control a simulated SRB character to follow a reference SRB motion extracted from a large and unstructured full-body motion dataset. The reference SRB motion is represented using a simplified set of features: the center of mass (COM) position, pelvis orientation, and the position and orientation of the midpoint of each feet. At each time step, the SRB character observes its current state $\mathbf{s}_t$ as well as a short window of future reference SRB states, $\mathbf{s}_{t+1:t+T}$. These future reference states serve as control signals to guide the character toward the desired motion. During training, these future states are sampled directly from the reference SRB motion. At inference time, they can either be extracted from arbitrary full-body motion clips or produced by a kinematics-based motion generator. Based on the current internal state and future conditions of the SRB character, the policy is implemented as a neural network that outputs an action through different action heads. The action includes a target transformation, a generalized velocity, desired footsteps, and target contact states, which will be detailed in Section 4. A quadratic programming (QP) solver is then used to compute the contact forces required to realize the desired acceleration in a physically plausible way. The desired footsteps are first projected onto the terrain surface to determine the contact points where these forces will be applied. The resulting contact forces, together with any external disturbances, are then fed into the simulator to generate the character's motion (Section A.3 of the Supplementary Material).

In the full-body motion generation module, a conditional variational autoencoder (CVAE) is used as the full-body motion predictor. The predictor is trained on pairs of reference full-body motions and corresponding ground-truth SRB trajectories. Since the reference SRB trajectories used during the training of the SRB character controller are not physically accurate, we first generate ground-truth SRB trajectories by tracking the reference SRB motions through simulation. These ground-truth SRB and full-body trajectories are then used to train the full-body motion predictor (Section 5.1).

At each time step during runtime, the predictor observes the current SRB state $\mathbf{s}_t$ and a history of past states $\mathbf{s}_{t-\tau:t-1}$ to predict the future full-body motion $\mathbf{s}^{full}_{t:t+W}$. The predicted motion is refined by the momentum-mapped space-time optimizer $\mathcal{M}$, resulting in a smooth and physically plausible trajectory $\mathcal{M}(\mathbf{s}^{full}_{t:t+W})$. The first frame of the optimized motion, $\mathcal{M}(\mathbf{s}^{full}_{t})$, is rendered at the current time step t (Section 5.2).

4 SRB Tracking Controller

4.1 SRB Character and Frames

Our SRB character follows a design similar to that proposed by Kwon et al. [Kwon et al. 2023]. The SRB model approximates the physical properties of a full-body articulated character using a box-shaped rigid body that inherits the same mass and composite inertia as the original character in its default posture. To model foot-ground interaction, each foot is equipped with five contact points: two at the big toe and little toe, two at the heel, and one at the center of the foot.

We first define three reference frames to represent the state of the SRB character. The **SRB frame** is attached to the center of mass of the SRB character. The **forward-facing SRB frame** shares the same origin as the SRB frame, but its x-axis is aligned with the horizontal projection of the SRB's forward direction, and its z-axis is aligned with the global vertical axis. The **projected SRB frame** is defined by vertically projecting the origin of the forward-facing SRB frame onto the ground or terrain surface.

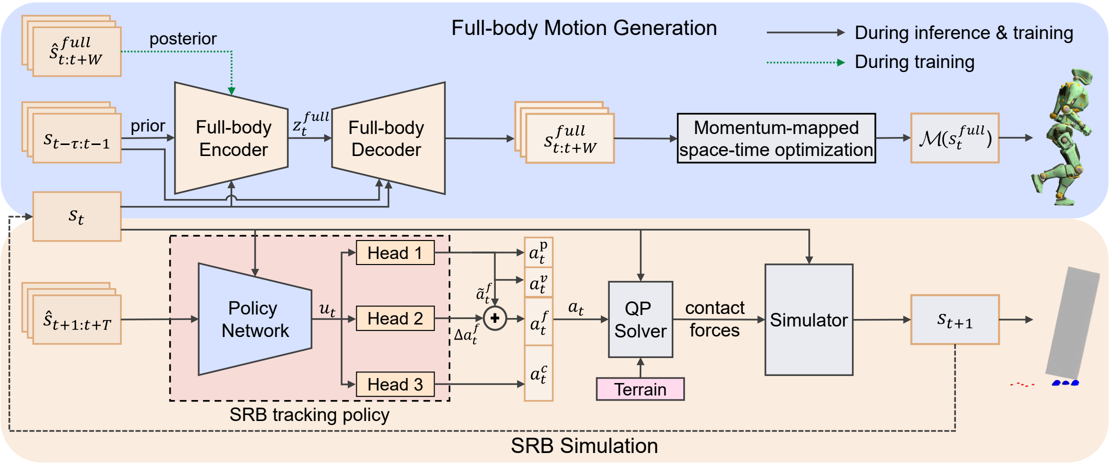


Fig. 2. Overview of our SRBTrack framework. The SRB tracking policy, trained using a combination of reinforcement and supervised learning on flat terrain, generalizes to uneven terrain at inference. A QP solver computes contact forces from predicted actions, while a full-body motion predictor outputs future states. The states are refined via momentum-mapped space–time optimization for rendering.

4.2 Reinforcement Learning Formulation

State. The state vector $\mathbf{s} \in \mathbb{R}^{25}$ of each frame is composed of the following components.

- The **center of mass state** $\mathbf{s}_o \in \mathbb{R}^{13}$ consists of the height, orientation, and generalized velocity of the SRB character's center of mass, all expressed in the projected SRB frame. The orientation is represented using two axes, commonly referred to as the 6D representation [Zhou et al. 2019].
- The **foot state** $\mathbf{s}_f^j \in \mathbb{R}^4$ $(j \in \{\text{left}, \text{right}\})$ consists the 3D position of each foot expressed in the forward-facing SRB frame, and the 1D orientation of each foot around the local vertical z-axis of forward-facing frame.
- The **foot contact state** $\mathbf{s}_c \in \mathbb{R}^4$ is a one-hot 4D vector that encodes the four possible combinations of binary contact states for the left and right feet with the ground.

Action. The action vector $\mathbf{a} \in \mathbb{R}^{22}$ consists of the following components.

- The **desired delta transformation of center of mass** $\mathbf{a}_p \in \mathbb{R}^6$ represents the local change in position and orientation, expressed in the SRB frame. It is applied to the current pose to compute the desired pose for the next frame.
- The **desired velocity of center of mass** $\mathbf{a}_v \in \mathbb{R}^6$ refers to the desired generalized velocity of the center of mass of the SRB character expressed in the SRB frame.
- The **desired foot state** $\mathbf{a}_f^j \in \mathbb{R}^3 (j \in \{\text{left}, \text{right}\})$ specifies the desired 2D planar landing position of each foot, along with the desired 1D rotation around the vertical axis (z-axis), all expressed in the projected SRB frame.
- The **desired feet contact state** $\mathbf{a}_c \in \mathbb{R}^4$ is a one-hot 4D vector which predicts the desired feet contact state fot the next time step.

Reward. The reward function r_t at each step t is configured as below:

$$r_t = w^s r_t^s - w^p r_t^p - w^e r_t^e - w^c r_t^c, \tag{1}$$

where r_t^s is the alive reward given when the episode continues, encouraging the character to maintain a stable and upright posture. An early termination (ET) condition is applied to end episodes when the character becomes unstable. The posture penalty $\mathbf{r}_t^p$ encourages the simulated SRB character to follow the COM trajectory of the reference SRB motion. The end-effector penalty $\mathbf{r}_t^e$ guides the policy to generate accurate positions and orientations of the feet. The contact penalty $\mathbf{r}_t^c$ ensures that the generated foot contact states match those of the reference SRB motion. A more detailed description of the reward functions is provided in Section A.1 of the Supplementary Material.

4.3 Policy

Unlike the approach in [Kwon et al. 2023], the feet state and contact states in our method are generated autoregressively by the policy. A key distinction of our approach lies in the policy architecture, which uniquely combines the advantages of kinematics-based motion supervision with RL-based control. This design also enhances training stability and sample efficiency compared to conventional RL methods.

The SRB tracking policy consists of an encoder, a decoder, and three action heads (Figure 2). The encoder takes as input the current state $\mathbf{s}_t$ and a short sequence of future reference states $\mathbf{s}_{t+1:t+T}$, where T denotes the window size. The decoder outputs an intermediate feature $\mathbf{u}_t$, which is then passed to the action heads. Supervising the outputs of the action heads allows this data-driven approach to avoid handcrafted foot trajectories and contact logic.

Specifically, the first action head predicts the desired delta transformation of the center of mass $\mathbf{a}_t^d$, the desired velocity of the center of mass $\mathbf{a}_t^v$, and the desired base feet state $\tilde{\mathbf{a}}_t^f$. The second head predicts the residual feet state $\Delta\mathbf{a}_t^f$, which is added to the base state to compute the final desired feet state: $\mathbf{a}_t^f = \tilde{\mathbf{a}}_t^f + \Delta\mathbf{a}_t^f$. The third head predicts the discrete desired foot contact state $\mathbf{a}_t^c$. All foot-related outputs are supervised using a dedicated loss function.

The feet of the SRB character are modeled as a set of massless contact points. Due to inherent differences in dynamics between the SRB character and a full humanoid, precisely replicating full-body footstep trajectories is not feasible. Instead of enforcing exact tracking of reference foot positions, we use the following loss function to guide the SRB character to adjust its foot targets in a way that better aligns with its simplified dynamics, while still reasonably following the reference motion:

$$\mathcal{L} = w_f \mathcal{L}_f + w_r \mathcal{L}_r + w_c \mathcal{L}_c. \tag{2}$$

Here, the feet state loss $\mathcal{L}_f$ penalizes the difference between the desired base feet state $\tilde{\mathbf{a}}_t^f$ and the target feet state $\hat{\mathbf{s}}_{t+1}^f$. The regularization loss $\mathcal{L}_r$ constrains the magnitude of the residual feet state vector accounting for model discrepancies. The contact loss $\mathcal{L}_c$ is defined as the cross-entropy (CE) loss between the predicted contact logit and the ground-truth discrete contact label. During training, this loss $\mathcal{L}$ is used as an auxiliary objective alongside the RL reward, forming a joint optimization objective that balances motion tracking performance with accurate footstep planning. A more detailed description is provided in Section A.2 of the Supplementary Material.

4.4 QP Problem

We compute physically consistent contact forces by formulating a QP problem. Desired COM acceleration is derived via a PD controller using pose and velocity errors. The policy predicts the desired pose, velocity, and foot landing states at each timestep. The predicted foot positions are first projected onto the terrain surface. To prevent foot penetration, they are further adjusted to the nearest safe region if they fall near sharp features such as steps or edges. A quadratic programming (QP) solver then optimizes the COM acceleration and the contact forces at these locations. The optimization minimizes both the acceleration tracking error and the magnitude of the contact forces, subject to the single-rigid-body dynamics constraint. Contact forces are parameterized using friction cone basis vectors, with a non-negativity constraint. The optimized contact forces are then passed to the simulator to advance the SRB state. Further details are provided in Section A.3 of the Supplementary Material.

5 Full-body Motion Generation

5.1 Full-body Motion Predictor

After training the SRB policy, we collect ground-truth SRB trajectories by tracking the reference SRB motions used during training. These ground-truth trajectories closely follow the reference but exhibit slight tracking errors that accumulate over time. We apply a simple refinement process to correct the root trajectory and remove foot sliding [Kwon et al. 2023]. The resulting refined full-body trajectories, paired with their corresponding ground-truth SRB trajectories, serve as supervision to train a conditional variational autoencoder (CVAE) capable of reconstructing full-body motion from arbitrary SRB motion inputs. The model takes as input a history sequence of SRB motion and predicts the current and future full-body states over a short horizon. The input of the full-body motion predictor $\mathbf{s}_{t-\tau:t} \in \mathbb{R}^{(\tau+1)\times 25}$ is the history sequence of SRB states, where $\tau = 10$. Each SRB state within the interval $[t-\tau, t]$ is concatenated along the temporal dimension. The predictor outputs the current and near-future full-body states over a short horizon.

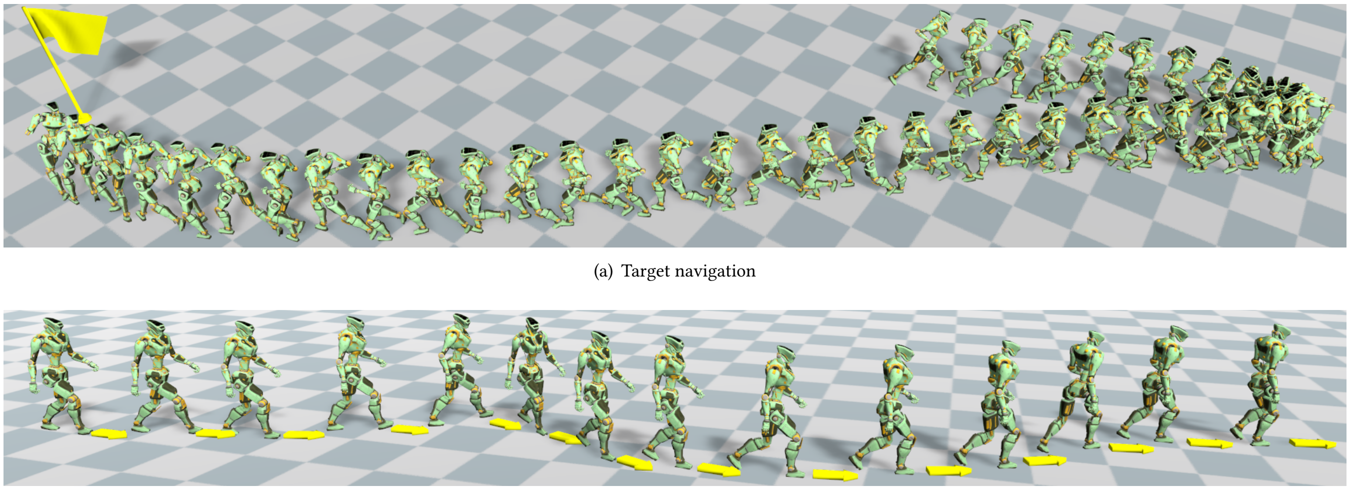
Full-body state. The predicted full-body state $\mathbf{s}_t^{full} \in \mathbb{R}^{214}$ consists of the character's pose $\mathbf{x} \in \mathbb{R}^{59}$ (represented using Euler angles), its first and second time derivatives $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$, the center of mass (COM) position $\mathbf{x}_{COM} \in \mathbb{R}^3$, the generalized momentum $\mathbf{h} \in \mathbb{R}^6$, and the positions, velocities and contact state of the four contact points (toe and heel for each foot) resulting in a total of 28 dimensions. To ensure coordinate-frame independence, all pose-related vectors are expressed relative to the simulated SRB's pose and generalized velocity. The positions and velocities of the full-body feet are encoded relative to the SRB's feet. Instead of directly predicting the full-body momentum $\mathbf{h}$, we train the network to learn the error between the full-body momentum and the SRB momentum. Further architectural and training details of the full-body predictor are provided in Section B of the the Supplementary Material.

5.2 Momentum-mapped Space Time Optimization

The full-body motion predictor provides an effective and efficient mapping from SRB states to full-body poses. However, due to the highly abstracted state representation of the SRB character, the resulting full-body pose from the predictor often exhibits noise and lacks physical realism. To address this, we propose momentum-mapped space-time optimization (MMSTO), a novel reconstruction method that improves motion quality and physical plausibility of the predictions $\mathbf{s}_{t:t+W}^{full}$ from the full-body motion predictor through a space-time optimizer $\mathcal{M}$. The first frame $\mathbf{x}_0 = \mathcal{M}(\mathbf{s}_t^{full})$ of the optimized full-body motion is used for rendering the current step.

The space-time optimization minimizes the cost function E_{mmsto} using an L-BFGS solver:

$$\begin{aligned}
E_{mmsto}(\mathbf{x}) = {} & w_1 \sum_{i=-1}^{3} \left\|\ddot{\mathbf{x}}_i - \bar{\ddot{\mathbf{x}}}_i\right\|^2 + w_2 \sum_{i=-1}^{3} \left\|\dot{\mathbf{x}}_i - \bar{\dot{\mathbf{x}}}_i\right\|^2 \\
& + w_3 \left\|\sum_{i=0}^{4} \mathbf{x}_i - \sum_{i=0}^{4} \bar{\mathbf{x}}_i\right\|^2 \\
& + w_4 \sum_{i=-1}^{3} \sum_{j\in\{L,R\}} \sum_{k=1}^{2} \left\|\dot{\mathbf{C}}^{jk}(\mathbf{x}_i) - \bar{\dot{\mathbf{C}}}^{ijk}\right\|^2 \\
& + w_5 \sum_{j\in\{L,R\}} \sum_{k=1}^{2} \left\|\sum_{i=0}^{4} \mathbf{C}^{jk}(\mathbf{x}_i) - \sum_{i=0}^{4} \bar{\mathbf{C}}^{ijk}\right\|^2 \\
& + w_6 \sum_{i=-1}^{3} \left\|\mathbf{J}_{m,i}(\dot{\mathbf{x}}_i) - \bar{\mathbf{h}}_i)\right\|^2 \\
& + w_7 \left\|\sum_{i=0}^{4} COM(\mathbf{x}_i) - \sum_{i=0}^{4} \bar{\mathbf{x}}_i^{COM})\right\|^2,
\end{aligned} \tag{3}$$



(a) Target navigation

(b) Joystick control

Fig. 3. Interactive character control

where $\mathbf{x} = (\mathbf{x}_{-2}, \mathbf{x}_{-1}, \cdots, \mathbf{x}_4)$ is the full-body poses over the planning horizon, represented in the world frame. The first two poses, $\mathbf{x}_{-2}$ and $\mathbf{x}_{-1}$, correspond to past states and are fixed, while the remaining poses are treated as optimization variables.

The first two terms are regularization terms that encourage the solution to stay close to the desired full-body velocity $\bar{\dot{\mathbf{x}}}$ and desired full-body acceleration $\bar{\ddot{\mathbf{x}}}$, respectively. The time-derivative $\dot{\mathbf{x}}$ is calculated via forward differentiation: $(\mathbf{x}_{i+1} - \mathbf{x}_i)\,/dt$. Acceleration $\ddot{\mathbf{x}}_i$ is computed as $\ddot{\mathbf{x}}_i = (-\mathbf{x}_{i-1} + 2\mathbf{x}_i - \mathbf{x}_{i+1})\,/dt^2$. All the desired values are obtained by transforming the full-body predictor's outputs into world coordinates using the SRB coordinate system. Instead of deriving the desired velocity and acceleration by differentiating the desired pose, these values are predicted directly to enhance robustness to prediction noise. The third term encourages the full-body poses to be, on average, similar to the desired pose. Using individual pose matching can transfer noise present in the desired poses to the generated poses, so the average is used instead.

The fourth term measures the difference between the velocity of the k^{th} contact point (toe and heel) of the j^{th} foot $\left(\dot{\mathbf{C}}^{jk}(\mathbf{x_i})\right)$, and its desired velocity $\bar{\dot{\mathbf{C}}}^{ijk}$. The next term measures the error between the average position of each contact point and the average of its corresponding desired positions.

The sixth term minimizes the error between the desired momentum $\bar{\mathbf{h}}_i$, computed from the SRB's momentum, and the full-body momentum $\mathbf{J}_{m,i}(\dot{\mathbf{x}}_i)$. The momentum Jacobian $\mathbf{J}_{m,i}$ relates the full-body's generalized velocity $\dot{\mathbf{x}}_i$ to the generalized centroidal momentum. The next term minimizes the average error between the desired COM positions, derived from the SRB's COM trajectory, and the full-body COM positions. Due to inherent dynamic differences between the SRB and the full body, some discrepancy between their momenta and COM positions are inevitable [Kwon et al. 2023]. These errors are predicted by the full-body predictor and used to compute the desired full-body momentum $\bar{\mathbf{h}}_i$ and the desired COM position $\bar{\mathbf{x}}_i^{COM}$ by correcting the SRB's momentum and COM position.

Since soft constraints are used to guide foot placement and COM position during optimization, the resulting motion may exhibit slight foot sliding or COM errors. These artifacts are corrected through a retargeting process using an analytic IK solver [Kovar et al. 2002b].

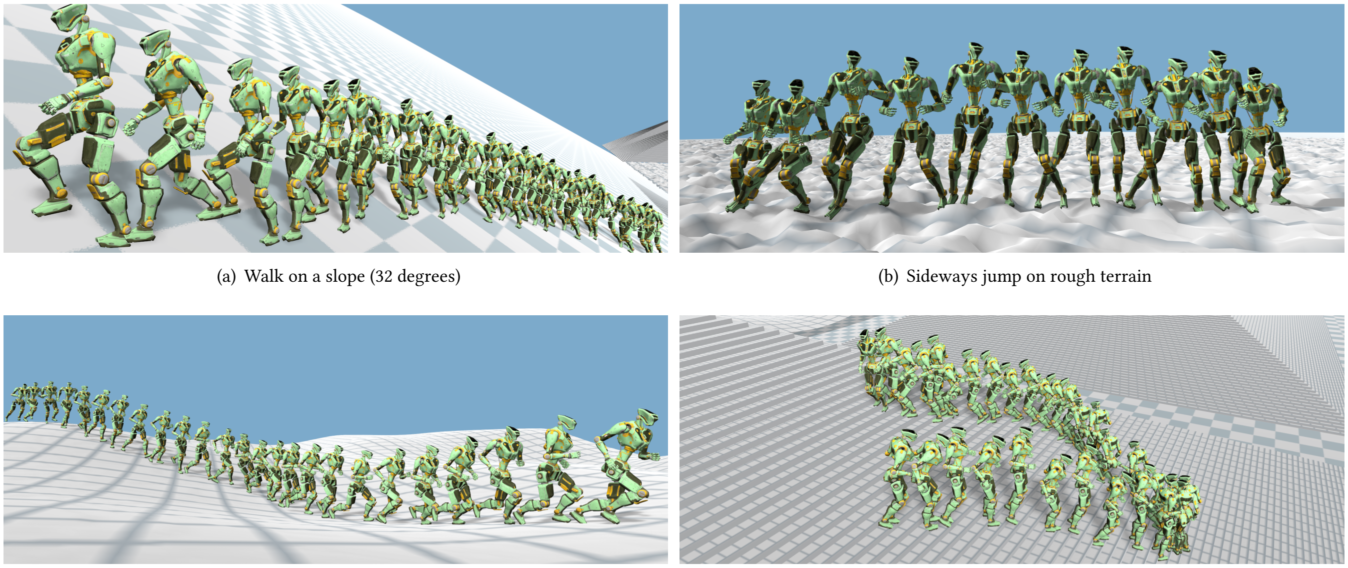
6 Experiments

We use Clarabel to solve the QP problem [Goulart and Chen 2024] and MuJoCo for forward dynamics [Todorov et al. 2012], both running at 120 Hz. All experiments are conducted using an Intel(R) Core(TM) i9-13980HX CPU and an NVIDIA GeForce RTX 4080 Laptop GPU. Our controller can generate full-body motion in real-time. We refer to Table 1 for related runtime statistics.

The training dataset consists of three unstructured locomotion sequences (walking, running, jumping) from LaFAN [Harvey et al. 2020], augmented with mirrored versions and a synthesized standing motion, with a total duration of approximately 25 minutes. The policy is trained using Proximal Policy Optimization (PPO) [Schulman et al. 2017] implemented in PyTorch [Paszke 2019], with 32 parallel environments. The learning rate is exponentially annealed from $1e-4$ to $1e-5$ over the first 15 million steps of the total 30 million training steps (approximately 7 hours). The full-body predictor is trained using the RAdam [Liu et al. 2019] with $\beta_1 = 0.9, \beta_2 = 0.999$ and a fixed learning rate $2e-3$. Further details on the training setup are provided in Section C of the Supplementary Material.

6.1 Tasks

This section showcases SRBTrack's generalization capability by integrating it with kinematics-based motion planners. Unlike task-specific controllers or end-to-end methods, SRBTrack enables plug-and-play use of existing planners without adaptation, tracking their



(a) Walk on a slope (32 degrees)

(b) Sideways jump on rough terrain

(c) Run on hill terrain

(d) Jump on pyramid terrain

Fig. 4. Adaptation for uneven terrain.

output motions online and requiring no retraining. To validate this, we choose two representative planners: MotionVAE [Ling et al. 2020b], a data-driven motion controller based on a CVAE model, and Motion Graph [Kovar et al. 2002a], a classical graph-based controller that enables kinematic motion synthesis via clip transitions. Leveraging existing implementations, we demonstrate two downstream tasks: target navigation and joystick control. For visual demonstrations of these tasks, as well as additional results on uneven terrain adaptation, external perturbations, and unseen motion tracking, please refer to the supplementary video.

Target navigation. We use the original MotionVAE implementation to generate future motion trajectories that guide the character to sprint toward randomly placed targets. Notably, MotionVAE produces motions at approximately $5.5m/s$, whereas the fastest motion in our SRB training set—running— reaches only around $3.5m/s$. Despite this speed mismatch, our method successfully generalizes to follow the trajectory and reach the target, as shown in Figure 3(a).

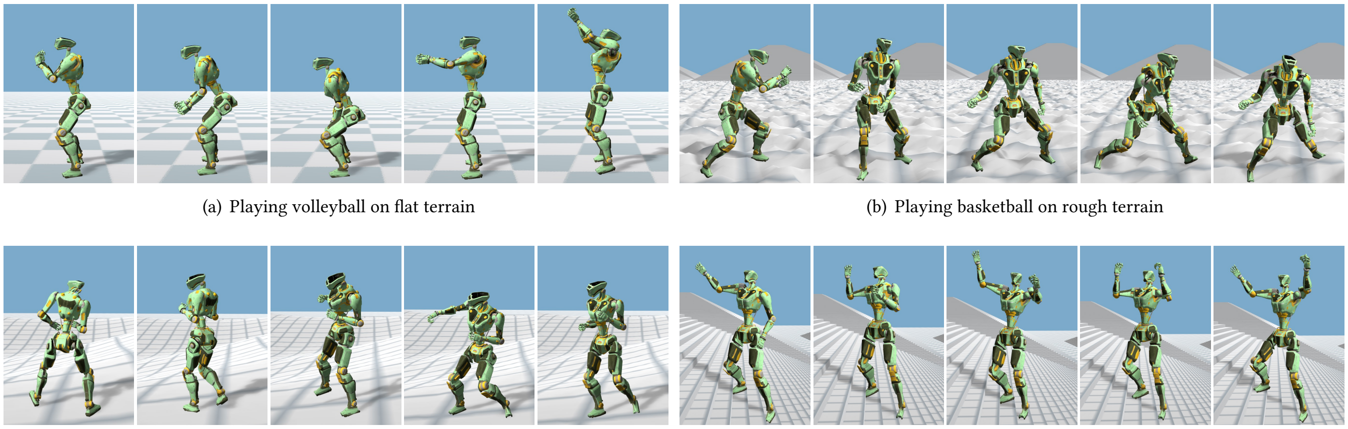
Joystick control. We implement joystick control using Motion Graph, where the character's COM velocity is guided to match a target 2D velocity on the ground plane, as shown in Figure 3(b). This Motion Graph can also be extended to other tasks such as speed control, motion switching, and locomotion over uneven terrain.

Uneven terrain. We construct three challenging uneven terrains–hill, rough, and pyramid—to demonstrate the robustness of our controller, as shown in Figure 4. Despite being trained only on flat terrain and without using scene observations, it successfully tracks reference motions on these unseen surfaces. We also test sloped terrains with constant inclinations, where the controller handles walking and jumping on slopes over 30° and runs on slopes around 20°, showing strong adaptability to real-world conditions. For additional scenarios, such as walking on stepping stones and crouching to avoid an obstacle, please refer to the supplementary video. On terrains with sharp or discontinuous edges, such as pyramid-shaped surfaces, we allow minor swing-foot penetration as a practical trade-off: while it slightly compromises physical accuracy, it effectively suppresses jitter and ensures smoother motion when the foot passes close to these edges.

Table 1. Statistics for generating locomotion on a flat ground. Average computation time for generating one second of final motion is measured across the entire training dataset.

	System Component	Time Cost
Ours	Policy Output	0.036s
	Environment Step(QP+simulation)	0.159s
	Full-body Predictor Output	0.012s
	MMSTO(with Δ)	0.806s
	Total	1.013s

External perturbations. We further evaluate the controller's robustness under strong external perturbations. It maintains balance while standing under $650N$ pushes, and remains stable during walking, running, and jumping under forces over $1000N$. We also simulate interactive scenarios with thrown balls hitting the character, where the controller remains stable without falling even on challenging terrains. These results demonstrate its ability to handle strong pushes and collisions in realistic conditions. When the policy attempts to recover from external disturbances, artifacts such as leg intersections may occur. This happens because both the swing


(a) Playing volleyball on flat terrain

(b) Playing basketball on rough terrain

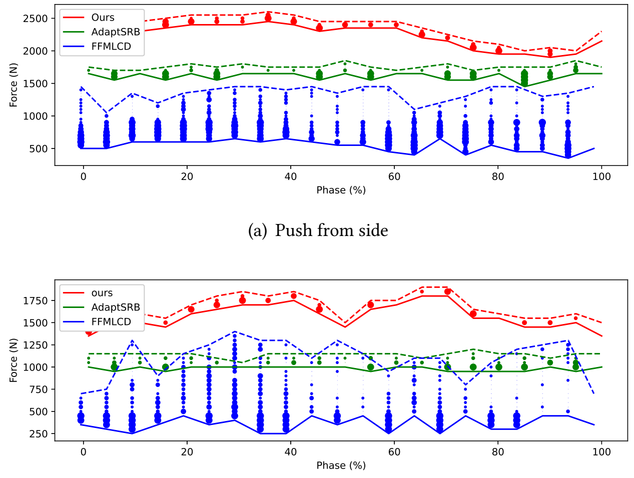
(c) Fighting on hill terrain

(d) Dancing on pyramid terrain

Fig. 5. Tracking unseen motions across diverse terrain.

foot positions planned by the policy and the character's foot positions computed in a data-driven manner lack explicit self-collision constraints. In extreme cases where disturbances exceed the controller's stabilization capacity, our system transitions seamlessly to a ragdoll fall mode, with each joint controlled by low-gain PD torques to produce natural falling motions. This design also allows future integration of full-body simulation–based recovery strategies to return to the SRB mode. Further details are provided in Section C.4 of the Supplementary Material.

Unseen motion tracking. We further assess the generalizability of the SRBTrack framework by tracking a variety of full-body motions not seen during policy training. As shown in Figure 5, we select representative samples from the LaFAN dataset [Harvey et al. 2020], including volleyball, basketball, fighting, and dancing. These motions involve complex full-body coordination that goes well beyond the basic locomotion patterns used during training. Despite this gap, our controller achieves high-quality tracking of these unseen motions and maintains robustness under challenging conditions such as uneven terrain and external disturbances. Our framework is also applied to the kinematic Text2Motion generator MoMask [Guo et al. 2023], where it effectively removes artifacts such as foot sliding and enables terrain adaptation. The supplementary video presents visual results.



(a) Push from side

(b) Push from back

Fig. 6. Robustness evaluation under external pushes. Solid line: Maximum force where the character consistently maintains balance. Dotted line: Minimum force where the character consistently falls. Circles: Success rate (out of 10 trials) for intermediate forces; size indicates the proportion of successful trials.

6.2 Comparison

To quantitatively evaluate robustness, we compare the balance maintenance rates of characters controlled by our method, AdaptSRB [Kwon et al. 2023], and FFMLCD [Kwon et al. 2020] in push experiments conducted across 20 evenly sampled motion phases of a Sprint reference motion. External forces ranging from 200 N to 3000 N are applied to the character's root from the side or back, parallel to the ground, for 0.2 seconds. Each force magnitude is tested 10 times per phase.

As shown in Figure 6, our method achieves significantly higher tolerence in both directions. Unlike AdaptSRB which relies on reference contact timing and manual motion phase adjustments, our policy learns to adapt foot placement and contact timing automatically through reinforcement learning and data-driven footstep modeling. FFMLCD, on the other hand, depends on nonlinear optimization per half gait cycle, limiting its responsiveness under strong perturbations.

6.3 Ablations

To demonstrate the effectiveness of our full-body reconstruction method, MMSTO, we perform an ablation study comparing two approaches: the fullbody motion predictor with conventional IK and the predictor combined with MMSTO. As shown in the supplementary video, conventional IK produces noisy results, particularly during foot landings, where abrupt pose changes occur due to sensitivity to contact variations. Moreover, since the predictor is trained without using any recovery motions, the IK-based approach fails to generate natural responses under external disturbances. In contrast, MMSTO produces smooth, physically plausible, and natural motions, even under challenging conditions. The supplementary video provides visual comparisons, along with additional ablations comparing the VAE and MLP architectures for the full-body motion predictor, and evaluations of SRBTrack outputs versus motion planner outputs.

7 Conclusion

In this paper, we introduced a robust, terrain-adaptive tracking framework based on a single-rigid-body (SRB) character. By leveraging a large motion dataset and a structured policy architecture, our method achieves strong zero-shot generalization to unseen terrains and external disturbances—without requiring environment observations or explicit modeling of external forces.

Despite being trained solely on locomotion behaviors and using a simplified SRB model with no articulated joints, our framework demonstrates strong generalization. SRBTrack accurately reconstructs diverse full-body motions, including complex upper-body styles. This is made possible by our momentum-mapped space–time optimization (MMSTO) module, which reconstructs full-body motion from SRB trajectories. While physics-based methods often exhibit superior performance in complex environment, they require terrain modeling, environment observations, extensive demonstrations, and costly training. In this sense, our method offers a pragmatic alternative by augmenting kinematic motion synthesis with physics-based simulation— providing a workable solution until full-body simulation approaches achieve the robustness required for deployment in real-world applications such as games. It retains the visual fidelity of kinematic methods while adding the dynamic interaction capabilities of physics-based models.

While SRBTrack performs well under a wide range of conditions, there is still room for improvement. Currently, gait responses to perturbations are entirely policy-driven, and in highly out-of-distribution states, recovery motions can appear noisy or unnatural. This could be addressed by incorporating recovery examples for disturbances into the training dataset or introducing phase-conditioned constraints [Holden et al. 2017; Starke et al. 2022, 2020]. Additionally, our current CPU-based QP solver poses a training bottleneck. Future research directions include using differentiable rigid-body dynamics and a GPU-accelerated QP solver to enhance training speed and scalability. We also aim to explore full-body dynamics controllers that use the SRB policy as a short-term motion planner and to exploit redundant degrees of freedom for secondary tasks.

Acknowledgments

This work was supported by grants from the National Research Foundation of Korea (NRF), funded by the Korean government (MSIT) (Nos. RS-2024-00354549 and RS-2023-00222776).

References

Simon Alexanderson, Rajmund Nagy, Jonas Beskow, and Gustav Eje Henter. 2023. Listen, Denoise, Action! Audio-Driven Motion Synthesis with Diffusion Models. *ACM Trans. Graph.* 42, 4 (2023), 44:1–44:20. doi:10.1145/3592458

Kevin Bergamin, Simon Clavet, Daniel Holden, and James Richard Forbes. 2019. DReCon: data-driven responsive control of physics-based characters. *ACM Transactions on Graphics (TOG)* 38, 6 (Nov. 2019), 206:1–206:11.

Rui Chen, Mingyi Shi, Shaoli Huang, Ping Tan, Taku Komura, and Xuelin Chen. 2024. Taming Diffusion Probabilistic Models for Character Control. In *ACM SIGGRAPH 2024 Conference Papers* (Denver, CO, USA) *(SIGGRAPH '24)*. Association for Computing Machinery, New York, NY, USA, Article 67, 10 pages. doi:10.1145/3641519.3657440

Yen-Lin Chen, Jianyuan Min, and Jinxiang Chai. 2009. Flexible registration of human motion data with parameterized motion models. In *Proceedings of the 2009 Symposium on Interactive 3D Graphics, SI3D 2009, February 27 - March 1, 2009, Boston, Massachusetts, USA*, Eric Haines, Morgan McGuire, Daniel G. Aliaga, Manuel M. Oliveira, and Stephen N. Spencer (Eds.). ACM, 183–190. doi:10.1145/1507149.1507180

Stelian Coros, Philippe Beaudoin, Kang Kang Yin, and Michiel van de Panne. 2008. Synthesis of constrained walking skills. In *ACM SIGGRAPH Asia 2008 Papers* (Singapore) *(SIGGRAPH Asia '08)*. Association for Computing Machinery, New York, NY, USA, Article 113, 9 pages. doi:10.1145/1457515.1409066

Frances Dosé and Arnie Cachelin. 2016. MotionGraphix: a 2.5D stereoscopic animation system on your iPad. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference, SIGGRAPH '16, Anaheim, CA, USA, July 24-28, 2016, Appy Hour*. ACM, 4:1–4:2. doi:10.1145/2936744.2936748

Levi Fussell, Kevin Bergamin, and Daniel Holden. 2021. SuperTrack: motion tracking for physically simulated characters using supervised learning. *ACM Transactions on Graphics (TOG)* 40, 6 (2021), 1–13.

Paul J. Goulart and Yuwen Chen. 2024. Clarabel: An interior-point solver for conic programs with quadratic objectives. arXiv:2405.12762 [math.OC]

Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. 2023. MoMask: Generative Masked Modeling of 3D Human Motions. (2023). arXiv:2312.00063 [cs.CV]

Perttu Hämäläinen, Joose Rajamäki, and C Karen Liu. 2015. Online control of simulated humanoids using particle belief propagation. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–13.

Félix G Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. 2020. Robust motion in-betweening. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 60–1.

Mohamed Hassan, Yunrong Guo, Tingwu Wang, Michael Black, Sanja Fidler, and Xue Bin Peng. 2023. Synthesizing Physical Character-Scene Interactions. In *ACM SIGGRAPH 2023 Conference Proceedings* (Los Angeles, CA, USA) *(SIGGRAPH '23)*. Association for Computing Machinery, New York, NY, USA, Article 63, 9 pages. doi:10.1145/3588432.3591525

Tairan He, Wenli Xiao, Toru Lin, Zhengyi Luo, Zhenjia Xu, Zhenyu Jiang, Changliu Liu, Guanya Shi, Xiaolong Wang, Linxi Fan, and Yuke Zhu. 2024. HOVER: Versatile Neural Whole-Body Controller for Humanoid Robots. *arXiv preprint arXiv:2410.21229* (2024).

Jonathan Ho and Stefano Ermon. 2016. Generative Adversarial Imitation Learning. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett (Eds.). 4565–4573. https://proceedings.neurips.cc/paper/2016/hash/cc7e2b878868cbae992d1fb743995d8f-Abstract.html

Daniel Holden, Oussama Kanoun, Maksym Perepichka, and Tiberiu Popa. 2020. Learned motion matching. *ACM Transactions on Graphics (ToG)* 39, 4 (2020), 53–1.

Daniel Holden, Taku Komura, and Jun Saito. 2017. Phase-Functioned Neural Networks for Character Control. *ACM Transactions on Graphics* 36, 4, Article 42 (2017), 13 pages.

Minsu Kim, Eunho Jung, and Yoonsang Lee. 2025. PhysicsFC: Learning User-Controlled Skills for a Physics-Based Football Player Controller. *ACM Trans. Graph.* 44, 4, Article 130 (July 2025), 21 pages. doi:10.1145/3731425

Lucas Kovar, Michael Gleicher, and Frédéric Pighin. 2002a. Motion graphs. *ACM Trans. Graph.* 21, 3 (July 2002), 473–482. doi:10.1145/566654.566605

Lucas Kovar, John Schreiner, and Michael Gleicher. 2002b. Footskate cleanup for motion capture editing. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (San Antonio, Texas) *(SCA '02)*. Association for Computing Machinery, New York, NY, USA, 97–104. doi:10.1145/545261.545277

Taesoo Kwon, Taehong Gu, Jaewon Ahn, and Yoonsang Lee. 2023. Adaptive Tracking of a Single-Rigid-Body Character in Various Environments. In *SIGGRAPH Asia 2023*

Conference Papers. 1–11.

Taesoo Kwon, Yoonsang Lee, and Michiel Van De Panne. 2020. Fast and flexible multilegged locomotion using learned centroidal dynamics. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 46–1.

Jehee Lee, Jinxiang Chai, Paul S. A. Reitsma, Jessica K. Hodgins, and Nancy S. Pollard. 2002. Interactive control of avatars animated with human motion data. *ACM Trans. Graph.* 21, 3 (2002), 491–500. doi:10.1145/566654.566607

Seunghwan Lee, Phil Sik Chang, and Jehee Lee. 2022. Deep Compliant Control. In *ACM SIGGRAPH 2022 Conference Proceedings* (Vancouver, BC, Canada) *(SIGGRAPH '22)*. Association for Computing Machinery, New York, NY, USA, Article 23, 9 pages. doi:10.1145/3528233.3530719

Yoonsang Lee, Sungeun Kim, and Jehee Lee. 2010. Data-driven biped control. *ACM Trans. Graph.* 29, 4 (2010), 1–8.

Sergey Levine and Vladlen Koltun. 2013. Guided policy search. In *International conference on machine learning*. PMLR, 1–9.

Sergey Levine, Yongjoon Lee, Vladlen Koltun, and Zoran Popovic. 2011. Space-time planning with parameterized locomotion controllers. *ACM Trans. Graph.* 30, 3 (2011), 23:1–23:11. doi:10.1145/1966394.1966402

Sergey Levine, Jack M. Wang, Alexis Haraux, Zoran Popovic, and Vladlen Koltun. 2012. Continuous character control with low-dimensional embeddings. *ACM Trans. Graph.* 31, 4 (2012), 28:1–28:10. doi:10.1145/2185520.2185524

Peizhuo Li, Kfir Aberman, Zihan Zhang, Rana Hanocka, and Olga Sorkine-Hornung. 2022. GANimator: neural motion synthesis from a single sequence. *ACM Trans. Graph.* 41, 4 (2022), 138:1–138:12. doi:10.1145/3528223.3530157

Hung Yu Ling, Fabio Zinno, George Cheng, and Michiel van de Panne. 2020a. Character controllers using motion VAEs. *ACM Trans. Graph.* 39, 4 (2020), 40. doi:10.1145/3386569.3392422

Hung Yu Ling, Fabio Zinno, George Cheng, and Michiel Van De Panne. 2020b. Character controllers using motion VAEs. *ACM Transactions on Graphics* 39, 4 (2020).

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2019. On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265* (2019).

Libin Liu, KangKang Yin, Michiel van de Panne, Tianjia Shao, and Weiwei Xu. 2010. Sampling-based contact-rich motion control. *ACM Trans. Graph.* 29, 4, Article 128 (July 2010), 10 pages. doi:10.1145/1778765.1778865

Shunlin Lu, Ling-Hao Chen, Ailing Zeng, Jing Lin, Ruimao Zhang, Lei Zhang, and Heung-Yeung Shum. 2024. HumanTOMATO: Text-aligned Whole-body Motion Generation. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024.* OpenReview.net. https://openreview.net/forum?id=maVIKlGqr7

Zhengyi Luo, Jinkun Cao, Alexander Winkler, Kris Kitani, and Weipeng Xu. 2023. Perpetual Humanoid Control for Real-time Simulated Avatars. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023.* IEEE, 10861–10870. doi:10.1109/ICCV51070.2023.01000

Adriano Macchietto, Victor Zordan, and Christian R. Shelton. 2009. Momentum control for balance. *ACM Transactions on Graphics* 28, 3 (July 2009), 80:1–80:8.

Jianyuan Min, Yen-Lin Chen, and Jinxiang Chai. 2009. Interactive generation of human animation with deformable motion models. *ACM Trans. Graph.* 29, 1 (2009), 9:1–9:12. doi:10.1145/1640443.1640452

Igor Mordatch, Martin de Lasa, and Aaron Hertzmann. 2010. Robust physics-based locomotion using low-dimensional planning. *ACM Transactions on Graphics* 29, 4 (2010), 71.

Igor Mordatch, Emanuel Todorov, and Zoran Popović. 2012. Discovery of complex behaviors through contact-invariant optimization. *ACM Trans. Graph.* 31, 4, Article 43 (July 2012), 8 pages. doi:10.1145/2185520.2185539

Liang Pan, Zeshi Yang, Zhiyang Dou, Wenjia Wang, Buzhen Huang, Bo Dai, Taku Komura, and Jingbo Wang. 2025. TokenHSI: Unified Synthesis of Physical Human-Scene Interactions through Task Tokenization. *CoRR* abs/2503.19901 (2025). doi:10.48550/ARXIV.2503.19901 arXiv:2503.19901

A Paszke. 2019. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703* (2019).

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

Xue Bin Peng, Glen Berseth, KangKang Yin, and Michiel van de Panne. 2017. DeepLoco: Dynamic Locomotion Skills Using Hierarchical Deep Reinforcement Learning. *ACM Transactions on Graphics (Proc. SIGGRAPH 2017)* 36, 4 (2017).

Xue Bin Peng, Yunrong Guo, Lina Halper, Sergey Levine, and Sanja Fidler. 2022. ASE: large-scale reusable adversarial skill embeddings for physically simulated characters. *ACM Transactions on Graphics* 41, 4 (July 2022), 94:1–94:17.

Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. 2018b. SFV: Reinforcement Learning of Physical Skills from Videos. *ACM Trans. Graph.* 37, 6, Article 178 (dec 2018), 14 pages. doi:10.1145/3272127.3275014

Mathis Petrovich, Michael J. Black, and Gül Varol. 2021. Action-Conditioned 3D Human Motion Synthesis with Transformer VAE. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021.* IEEE, 10965–10975. doi:10.1109/ICCV48922.2021.01080

Jia Qin, Youyi Zheng, and Kun Zhou. 2022. Motion In-Betweening via Two-Stage Transformers. *ACM Trans. Graph.* 41, 6, Article 184 (Nov. 2022), 16 pages. doi:10.1145/3550454.3555454

Sigal Raab, Inbal Leibovitch, Peizhuo Li, Kfir Aberman, Olga Sorkine-Hornung, and Daniel Cohen-Or. 2023. MoDi: Unconditional Motion Synthesis from Diverse Data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023.* IEEE, 13873–13883. doi:10.1109/CVPR52729.2023.01333

Davis Rempe, Zhengyi Luo, Xue Bin Peng, Ye Yuan, Kris Kitani, Karsten Kreis, Sanja Fidler, and Or Litany. 2023. Trace and Pace: Controllable Pedestrian Animation via Guided Trajectory Diffusion. In *Conference on Computer Vision and Pattern Recognition (CVPR).*

Alla Safonova and Jessica K. Hodgins. 2007. Construction and optimal search of interpolated motion graphs. *ACM Trans. Graph.* 26, 3 (2007), 106. doi:10.1145/1276377.1276510

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

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

Sebastian Starke, Yiwei Zhao, Taku Komura, and Kazi A Zaman. 2020. Local motion phases for learning multi-contact character movements. *ACM Trans. Graph.* 39, 4 (2020), 54.

Yuval Tassa, Tom Erez, and Emanuel Todorov. 2012. Synthesis and stabilization of complex behaviors through online trajectory optimization. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems.* IEEE, 4906–4913.

Chen Tessler, Yunrong Guo, Ofir Nabati, Gal Chechik, and Xue Bin Peng. 2024. MaskedMimic: Unified Physics-Based Character Control Through Masked Motion Inpainting. *ACM Transactions on Graphics (TOG)* (2024).

Chen Tessler, Yifeng Jiang, Erwin Coumans, Zhengyi Luo, Gal Chechik, and Xue Bin Peng. 2025. MaskedManipulator: Versatile Whole-Body Control for Loco-Manipulation. arXiv:2505.19086 [cs.RO] https://arxiv.org/abs/2505.19086

Chen Tessler, Yoni Kasten, Yunrong Guo, Shie Mannor, Gal Chechik, and Xue Bin Peng. 2023. CALM: Conditional Adversarial Latent Models&nbsp; for Directable Virtual Characters. In *ACM SIGGRAPH 2023 Conference Proceedings* (Los Angeles, CA, USA) *(SIGGRAPH '23)*. Association for Computing Machinery, New York, NY, USA, Article 37, 9 pages. doi:10.1145/3588432.3591541

Guy Tevet, Sigal Raab, Setareh Cohan, Daniele Reda, Zhengyi Luo, Xue Bin Peng, Amit H Bermano, and Michiel van de Panne. 2024. CLoSD: Closing the Loop between Simulation and Diffusion for multi-task character control. *arXiv preprint arXiv:2410.03441* (2024).

Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit Haim Bermano. 2023. Human Motion Diffusion Model. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023.* OpenReview.net. https://openreview.net/forum?id=SJ1kSyO2jwu

Emanuel Todorov, Tom Erez, and Yuval Tassa. 2012. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems.* IEEE, 5026–5033.

Takara Everest Truong, Michael Piseno, Zhaoming Xie, and Karen Liu. 2024. PDP: Physics-Based Character Animation via Diffusion Policy. In *SIGGRAPH Asia 2024 Conference Papers* (Tokyo, Japan) *(SA '24)*. Association for Computing Machinery, New York, NY, USA, Article 86, 10 pages. doi:10.1145/3680528.3687683

Michiel van de Panne. 2014. Motion fields for interactive character animation: technical perspective. *Commun. ACM* 57, 6 (2014), 100. doi:10.1145/2602759

Alexander W Winkler, C Dario Bellicoso, Marco Hutter, and Jonas Buchli. 2018. Gait and trajectory optimization for legged systems through phase-based end-effector parameterization. *IEEE Robotics and Automation Letters* 3, 3 (2018), 1560–1567.

Andrew Witkin and Michael Kass. 1988. Spacetime constraints. In *Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '88).* Association for Computing Machinery, New York, NY, USA, 159–168. doi:10.1145/54852.378507

Jungdam Won, Deepak Gopinath, and Jessica Hodgins. 2020. A scalable approach to control diverse behaviors for physically simulated characters. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 33–1.

Jungdam Won, Deepak Gopinath, and Jessica Hodgins. 2022a. Physics-based Character Controllers Using Conditional VAEs. *ACM Trans. Graph.* 41, 4, Article 96 (2022). https://doi.org/10.1145/3528223.3530067

Jungdam Won, Deepak Gopinath, and Jessica Hodgins. 2022b. Physics-based character controllers using conditional VAEs. *ACM Transactions on Graphics* 41, 4 (2022), 96:1–96:12.

Yan Wu, Korrawe Karunratanakul, Zhengyi Luo, and Siyu Tang. 2025. UniPhys: Unified Planner and Controller with Diffusion for Flexible Physics-Based Character Control. *arXiv preprint arXiv:2504.12540* (2025).

Zhen Wu, Jiaman Li, and C. Karen Liu. 2024. Human-Object Interaction from Human-Level Instructions. *CoRR* abs/2406.17840 (2024). doi:10.48550/ARXIV.2406.17840 arXiv:2406.17840

Chen Xin, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, Jingyi Yu, and Gang Yu. 2023. Executing your Commands via Motion Diffusion in Latent Space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Pei Xu and Ioannis Karamouzas. 2021. A GAN-Like Approach for Physics-Based Imitation Learning and Interactive Character Control. *Proc. ACM Comput. Graph. Interact. Tech.* 4, 3, Article 44 (Sept. 2021), 22 pages. doi:10.1145/3480148

Pei Xu, Kaixiang Xie, Sheldon Andrews, Paul G. Kry, Michael Neff, Morgan Mcguire, Ioannis Karamouzas, and Victor Zordan. 2023. AdaptNet: Policy Adaptation for Physics-Based Character Control. *ACM Trans. Graph.* 42, 6, Article 177 (Dec. 2023), 17 pages. doi:10.1145/3618375

Heyuan Yao, Zhenhua Song, Baoquan Chen, and Libin Liu. 2022. ControlVAE: Model-Based Learning of Generative Controllers for Physics-Based Characters. *ACM Transactions on Graphics* 41, 6 (2022), 183:1–183:16.

Heyuan Yao, Zhenhua Song, Yuyang Zhou, Tenglong Ao, Baoquan Chen, and Libin Liu. 2024. Moconvq: Unified physics-based motion control via scalable discrete representations. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–21.

Xinyu Yi, Yuxiao Zhou, Marc Habermann, Soshi Shimada, Vladislav Golyanik, Christian Theobalt, and Feng Xu. 2022. Physical Inertial Poser (PIP): Physics-Aware Real-Time Human Motion Tracking From Sparse Inertial Sensors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 13167–13178.

KangKang Yin, Kevin Loken, and Michiel Van de Panne. 2007. Simbicon: Simple biped locomotion control. *ACM Transactions on Graphics (TOG)* 26, 3 (2007), 105–es.

Zhiqi Yin, Zeshi Yang, Michiel van de Panne, and KangKang Yin. 2021. Discovering diverse athletic jumping strategies. *ACM Trans. Graph.* 40, 4 (2021), 91:1–91:17. doi:10.1145/3450626.3459817

Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. 2023. PhysDiff: Physics-Guided Human Motion Diffusion Model. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*. IEEE, 15964–15975. doi:10.1109/ICCV51070.2023.01467

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. 2023b. T2M-GPT: Generating Human Motion from Textual Descriptions with Discrete Representations. *CoRR* abs/2301.06052 (2023). doi:10.48550/ARXIV.2301.06052 arXiv:2301.06052

Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. 2024. MotionDiffuse: Text-Driven Human Motion Generation With Diffusion Model. *IEEE Trans. Pattern Anal. Mach. Intell.* 46, 6 (2024), 4115–4128. doi:10.1109/TPAMI.2024.3355414

Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. 2023a. ReMoDiffuse: Retrieval-Augmented Motion Diffusion Model. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*. IEEE, 364–373. doi:10.1109/ICCV51070.2023.00040

Xinyi Zhang and Michiel van de Panne. 2018. Data-driven autocompletion for keyframe animation. In *Proceedings of the 11th Annual International Conference on Motion, Interaction, and Games, MIG 2018, Limassol, Cyprus, November 08-10, 2018*, Panayiotis Charalambous, Yiorgos Chrysanthou, Ben Jones, and Jehee Lee (Eds.). ACM, 10:1–10:11. doi:10.1145/3274247.3274502

Xuan Zhao, Sakmongkon Chumkamon, Shuanda Duan, Juan Rojas, and Jia Pan. 2018. Collaborative Human-Robot Motion Generation Using LSTM-RNN. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. 1–9. doi:10.1109/HUMANOIDS.2018.8625068

Wenyang Zhou, Zhiyang Dou, Zeyu Cao, Zhouyingcheng Liao, Jingbo Wang, Wenjia Wang, Yuan Liu, Taku Komura, Wenping Wang, and Lingjie Liu. 2024. EMDM: Efficient Motion Diffusion Model for Fast and High-Quality Motion Generation. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 15060)*, Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (Eds.). Springer, 18–38. doi:10.1007/978-3-031-72627-9_2

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. 2019. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5745–5753.