Motion In-betweening for Physically Simulated Characters

Deepak Gopinath
Meta AI
USA
dgopinath@fb.com

Hanbyul Joo
Seoul National University
South Korea
hbjoo@snu.ac.kr

Jungdam Won
Meta AI
USA
jungdam@fb.com

ABSTRACT

We present a motion in-betweening framework to generate high quality, physically plausible character animation when we are given temporally sparse keyframes as soft animation constraints. More specifically, we learn imitation policies for physically simulated characters by using deep reinforcement learning where the policies can access limited information only. Once learned, the physically simulated characters are capable of adapting to external perturbations while following given sparse input keyframes. We demonstrate the performance of our framework on two different motion datasets and also compare our results with the the results generated by a baseline imitation policy.

CCS CONCEPTS

• **Computing methodologies → Physical simulation**; Motion capture.

KEYWORDS

Character Animation, Physics-based Simulation and Control, Reinforcement Learning, Deep Learning, Neural Network, Multi-agent

ACM Reference Format:
Deepak Gopinath, Hanbyul Joo, and Jungdam Won. 2022. Motion In-betweening for Physically Simulated Characters. In *SIGGRAPH Asia 2022 Posters (SA '22 Posters), December 06-09, 2022.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3550082.3564186

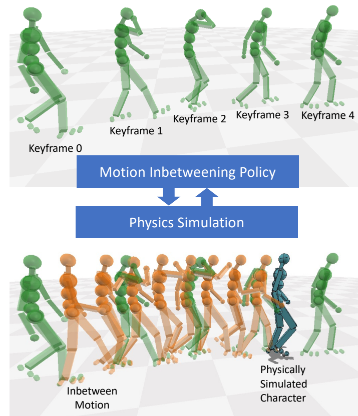


Figure 1: An overview of our system. The input to our motion in-betweening framework is a sequence of sparse keyframes temporally separated by a fixed time interval. We use a reinforcement learning based policy to generate in-between motion for a physically simulated character.

1 INTRODUCTION

Creating realistic human motions for 3D skeletal characters is one of the fundamental processes in graphics applications such as animation, games, or virtual/augmented reality. Motion *in-betweening* is a popular method to create skeletal animations, where users (artists) provide keyframe poses with less temporal granularity and the system automatically generates intermediate poses with finer granularity. When keyframe poses are temporally close enough, a simple linear or spline interpolation could generate smooth and plausible results while it becomes non-trivial as they get more sparse because the problem is highly under-constrained. Recently, motion in-betweening methods for sparsely keyframed poses (e.g. longer than 1 second) have been proposed [Harvey et al. 2020]. The main idea is to learn a deep neural network in a supervised fashion by using existing motion datasets. Once the model is learned, the generated intermediate poses will fall under the distribution of the training data, potentially resolving both under-constrained and naturalness issues if sufficiently large and high-quality data is used.

In this literature, we demonstrate a new approach to solve the motion *in-betweening* problem given sparsely keyframed poses. More specifically, our approach generates intermediate poses based on physically simulated characters of which control policies (a.k.a controllers) are trained by using deep reinforcement learning (RL). We develop new formulation that is suitable for the problem, where the state can only access the sparse input poses while the reward is computed from the ground-truth motions. Because our approach uses physically simulated characters, it has several unique advantages over existing kinematics-based approaches. For example, our method can generate physically plausible motions even if bad input poses are given. Additionally, motions adapting to external perturbation can emerge during the motion generation.

2 OUR APPROACH

Our system takes a sequence of keyframe poses $(\mathbf{P}_0^{\text{key}}, \mathbf{P}_k^{\text{key}}, \mathbf{P}_{2k}^{\text{key}}, \cdots)$ with a fixed coarse time interval as input ($k = 1$ in our model), then outputs a motion $(\mathbf{P}_0, \mathbf{P}_m, \cdots, \mathbf{P}_{k-m}, \mathbf{P}_k, \cdots, \mathbf{P}_{2k-m}, \mathbf{P}_{2k}, \cdots)$ at a desired dense time interval ($m = 1/30$ in our model). The goal is to generate motion that is smooth, physically plausible, natural-looking, and reasonably satisfying the input keyframe constraints (see Figure 1).

More specifically, our framework learns an imitation policy based on deep reinforcement learning (RL). The state $\mathbf{s}_t = (\mathbf{s}_t^{\text{key}}, \mathbf{s}_t^{\text{sim}})$ in

SA '22 Posters, December 06-09, 2022, Daegu, Republic of Korea

ACM ISBN 978-1-4503-9462-8/22/12.
https://doi.org/10.1145/3550082.3564186

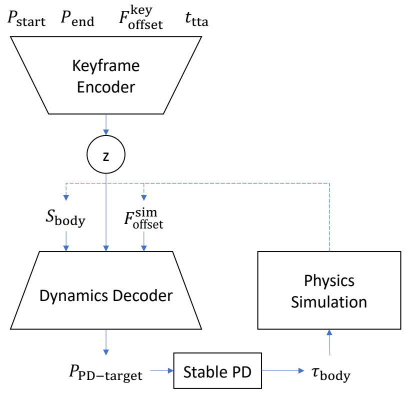


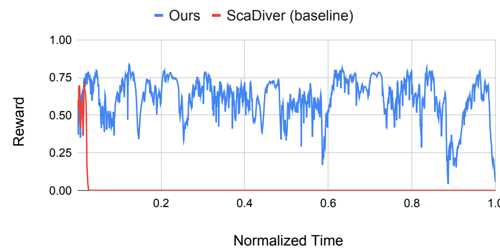
Figure 2: A deep RL control policy for motion in-betweening.

RL includes both the keyframe state $\mathbf{s}^{\text{key}}_t = (\mathbf{P}^{\text{key}}_{\text{t}}, \mathbf{P}^{\text{key}}_{\text{t+1}}, \mathbf{F}^{\text{key}}_{\text{offset}}, \mathbf{t}_{\text{tta}})$ and the simulation state $\mathbf{s}^{sim}_t = (\mathbf{S}^{\text{sim}}_{\text{body}}, \mathbf{F}^{\text{sim}}_{\text{offset}})$, where $(\mathbf{P}^{\text{key}}_{\text{t}}, \mathbf{P}^{\text{key}}_{\text{t+1}})$ are the current and next input keyframes, $\mathbf{F}^{\text{key}}_{\text{offset}}$ refers to the relative facing frame (i.e. homogeneous transformation of the root joint projected onto the ground plane) of the next keyframe with respect to the current keyframe, $\mathbf{t}_{\text{tta}}$ is the time-to-arrival feature that includes time remaining to reach the next keyframe (in seconds) and temporal embedding of the current time step as described in [Harvey et al. 2020], $\mathbf{S}^{\text{sim}}_{\text{body}}$ is the dynamic state of the simulated character as constructed in [Won et al. 2020], and $\mathbf{F}^{\text{key}}_{\text{offset}}$ is the relative facing frame of the simulated character with respect to the current keyframe. The action $\mathbf{a}_t$ in RL is a target pose for the stable PD controller, which computes joint torques τ_{body} to actuate the simulated character. Physics simulation then computes the next state $\mathbf{s}_{t+1}$. We use the same multiplicative reward function $\mathbf{r}_t$ used in [Won et al. 2020], which measures the similarity between simulated motion and the ground-truth motion through the five difference terms: joint angles, joint velocities, end-effector positions, center-of-mass position, and the root joint transformation.

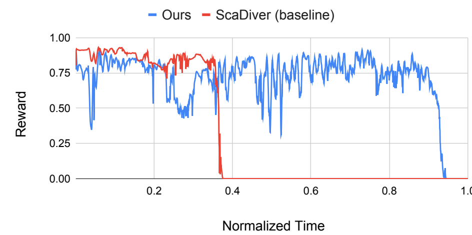
Figure 2 depicts an encoder-decoder structure that we adopt for our control policy. The idea here is to employ the keyframe encoder to produce a reduced vector representation $\mathbf{z}$ of the in-between pose for the current time step. The output of the encoder is concatenated with the simulation state and fed to the dynamics decoder to produce an action.

3 RESULTS

We adopted an open-sourced implementation of ScaDiver [Won et al. 2020] to deploy our method. PyBullet and RLlib were used for physics simulation and deep reinforcement learning, respectively, and *Proximal Policy Optimization* was used as our deep RL algorithm. We trained our model independently on two types of motion: 11 locomotion sequences (≈45.5 min) from the LaFAN1 dataset [Harvey et al. 2020], 10 walking sequences (≈10 min) randomly generated using a pretrained Phase-Functioned Neural Network (PFNN) controller [Holden et al. 2017]. The training takes approximately 3 days in total, where 400M simulated steps were approximately generated. Figure 1 shows a snapshot of intermediate postures generated by



(a) PFNN generated dataset



(b) LaFAN1 locomotion dataset

Figure 3: Imitation reward for each frame of in-betweening of keyframes from unseen motions.

our model trained with LaFAN1 dataset (see supplementary video for example sequences), where the learned control policy can successfully match input sparse keyframes while generating motions that are physically plausible and resemble ground-truth motions.

To compare the effectiveness of our approach, we trained a baseline imitation policy based on [Won et al. 2020] on the same data as above, where the policy consumes the densely specified keyframe postures (i.e., future reference motion). Since such information is not available during test time, we provided pseudo-reference motion by linearly interpolating the input sparse keyframe postures. Figure 3 shows the performance comparison over unseen motions (in-distribution), where the reward values are depicted along the normalized time. Our policy outperforms the baseline, that suffers from state mismatch between training/test time, by a large margin.

4 CONCLUSION

We demonstrated a new approach for motion in-betweening given temporally sparse keyframes by using physically simulated characters and deep RL. There are many directions for future work. We want to remove the constraint on the time interval between consecutive keyframes, which is currently fixed as 1s. Additionally, the control policy that we learned can only generate locomotion. We want to build a more general model that can generate more diverse behaviors, for example, the entire CMU or LaFAN1 dataset.

REFERENCES

Félix G. Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. 2020. Robust Motion In-Betweening. 39, 4 (2020).
Daniel Holden, Taku Komura, and Jun Saito. 2017. Phase-functioned Neural Networks for Character Control. *ACM Trans. Graph.* 36, 4, Article 42 (July 2017). http://doi.acm.org/10.1145/3072959.3073663
Jungdam Won, Deepak Gopinath, and Jessica Hodgins. 2020. A Scalable Approach to Control Diverse Behaviors for Physically Simulated Characters. *ACM Trans. Graph.* 39, 4, Article 33 (2020). https://doi.org/10.1145/3386569.3392381