

---

# SMPLlympics: Sports Environments for Physically Simulated Humanoids

---

Anonymous Author(s)

Affiliation

<https://smplolympics.github.io>

1	<b>A Introduction</b>	<b>1</b>
2	<b>B Implementation Details</b>	<b>1</b>
3	B.1 Rewards and Termination Conditions . . . . .	1
4	B.2 Multi-person Sports . . . . .	3
5	B.3 Hyperparamters . . . . .	5
6	B.4 Details about Baselines . . . . .	5
7	<b>C Additional Ablations</b>	<b>5</b>
8	<b>D Broader Social impact</b>	<b>6</b>

## 9 A Introduction

10 In the appendix, we provide comprehensive implementation details for SMPLlympics, including  
11 the reward designs for each sport environment, training procedures, and hyperparameters. Extensive  
12 qualitative results can be accessed on our [supplement site](#), where we provide visualizations of all  
13 sports environments and training results based on our preliminary reward designs. Baseline results  
14 (PPO, AMP, PULSE, PULSE+AMP) are presented to support the quantitative findings discussed in  
15 the main paper. Furthermore, we offer visualizations of the reference motion extracted from in-the-  
16 wild videos. For our pipeline to acquire the human demonstration in the SMPL format, we conduct  
17 an ablation study evaluating the impact of employing a motion imitator (PHC [2]) as a refinement  
18 step. Code, videos, and asset attributions can also be found in our supplementary materials.

## 19 B Implementation Details

### 20 B.1 Rewards and Termination Conditions

21 **High Jump.** For high jump, the humanoid’s task is to leap over a horizontal bar positioned 20m  
22 ahead and 6m to the left of its starting point. The humanoid aims to reach the goal point  $p^{\text{g-high jump}} =$   
23  $(22, 6, 1)$  located 2 m behind the bar. The reward function is defined as follows:

$$\mathcal{R}^{\text{high jump}}(s_t^p, s_t^{\text{g-high jump}}) \triangleq \begin{cases} 1 \times r_t^p & \text{if } p_{t,x}^p \leq 19.5\text{m}, \\ 1 \times r_t^p + 1 \times r_t^h & \text{if } 19.5\text{m} < p_{t,x}^p < 20.5\text{m}, \\ 1 \times r_t^p & \text{if } 20.5\text{m} \leq p_{t,x}^p. \end{cases} \quad (1)$$

where  $\mathbf{p}_{t,x}^p$  denotes the x-axis position. The height reward,  $r_t^h = \mathbf{p}_{t,z}^p$ , with  $\mathbf{p}_{t,z}^p$  representing the z-axis position, incentivizes the humanoid to jump higher. The position reward,  $r_t^p = \|\mathbf{p}_{t-1}^p - \mathbf{p}^{\text{g-high jump}}\|_2 - \|\mathbf{p}_t^p - \mathbf{p}^{\text{g-high jump}}\|_2$  (clamped to  $[0,1]$ ), motivates the humanoid to reach the goal. An episode is terminated if the humanoid falls down, fails to leap over the bar, or moves beyond the designated run-up area.

**Long Jump.** In the long jump environment, the humanoid has a 20-meter runway before the jump line, which its feet should not exceed. The humanoid’s goal is to reach the goal position,  $\mathbf{p}^{\text{g-long jump}} = (30, 0, 1)$ . The training reward is defined as follows:

$$\mathcal{R}^{\text{long jump}}(\mathbf{s}_t^p, \mathbf{s}_t^{\text{g-long jump}}) \triangleq \begin{cases} 1 \times r_t^p + 0.01 \times r_t^v & \text{if } \mathbf{p}_{t,x}^p \leq 20\text{m}, \\ 1 \times r_t^p + 0.01 \times r_t^v + 0.1 \times r_t^h + 30 \times r^l & \text{if } 20\text{m} < \mathbf{p}_{t,x}^p. \end{cases} \quad (2)$$

The position reward,  $r_t^p = \|\mathbf{p}_{t-1}^p - \mathbf{p}^{\text{g-long jump}}\|_2 - \|\mathbf{p}_t^p - \mathbf{p}^{\text{g-long jump}}\|_2$  (clamped to  $[0,1]$ ) encourages the humanoid to reach the goal point. The velocity reward,  $r_t^v = \mathbf{v}_{t,x}^p$  prompts the humanoid to reach higher speed along the x-axis. The jump height reward  $r_t^h = \mathbf{p}_{t,z}^p$  encourages the humanoid to jump higher after reaching the jump line. The jump length reward  $r_t^l = \mathbf{p}_{t,x}^p - 20$  promotes longer final jump length. Each episode terminates if the humanoid falls or runs off the track.

**Hurdling.** In the hurdling task, the humanoid aims to reach a finish line 110m ahead while jumping over 10 hurdles, each 1.067m high. The first hurdle is placed 13.72m from the start, with subsequent hurdles spaced every 9.14m. The reward function is defined as  $\mathcal{R}^{\text{hurdling}}(\mathbf{s}_t^p, \mathbf{s}_t^{\text{g-hurdling}}) \triangleq r_t^{\text{distance}}$ , which encourages the agent to run towards the finish line and clear each hurdle.

$$\mathcal{R}^{\text{hurdling}}(\mathbf{s}_t^p, \mathbf{s}_t^{\text{g-hurdling}}) \triangleq 1 \times r_t^{\text{distance}} \quad (3)$$

The distance reward,  $r_t^{\text{distance}} = \|\mathbf{p}_{t-1}^p - \mathbf{p}^{\text{g-hurdling}}\|_2 - \|\mathbf{p}_t^p - \mathbf{p}^{\text{g-hurdling}}\|_2$ , is clamped to  $[0, 1]$  and encourages the humanoid to get closer to the goal point. We terminate each episode if the character falls or runs off the track.

**Golf.** In the golf task, the humanoid is equipped with a golf club of dimensions of  $0.05\text{m} \times 0.025\text{m} \times 0.02\text{m}$ . The target location for the golf ball is positioned to the left of the humanoid, in the direction of the x-axis, at a distance ranging from 0m to 20m. The reward function is defined as follows:

$$\mathcal{R}^{\text{golf}}(\mathbf{s}_t^p, \mathbf{s}_t^{\text{g-golf}}) \triangleq 1 \times r_t^p + 1 \times r_t^c + 1 \times r_t^g + 1 \times r_t^{\text{pred}} \quad (4)$$

The position reward,  $r_t^p \triangleq \|\mathbf{p}_{t-1}^{\text{ball}} - \mathbf{p}_t^{\text{tar}}\|_2 - \|\mathbf{p}_t^{\text{ball}} - \mathbf{p}_t^{\text{tar}}\|_2$ , clamped such that  $0 < r_t^p < 1$ , encourages the ball to get closer to the target. The contact reward  $r_t^c$  encourages swinging the golf club to hit the ball, defined as:

$$r_t^c = \begin{cases} 1 \times \exp(-100 \times \|\mathbf{p}_t^{\text{ball}} - \mathbf{p}_t^{\text{club}}\|^2) & \text{if } C_{\text{cb}} = 0, \\ 1 & \text{if } C_{\text{cb}} = 1. \end{cases} \quad (5)$$

Here,  $C_{\text{cb}} = 0$  indicates that the club has not made contact with the ball and  $C_{\text{cb}} = 1$  indicates the club has made contact. The goal reward,  $r_t^g = \exp(-0.1 \times \|\mathbf{p}_{t,xy}^{\text{ball}} - \mathbf{p}_{t,xy}^{\text{tar}}\|^2)$ , encourages the ball to reach the target position in the x-y plane. In addition, we predict the ball’s trajectory and provide a dense reward  $r_t^{\text{pred}} = \exp(-0.1 \times \|\mathbf{p}^{\text{land}} - \mathbf{p}_{t,xy}^{\text{ball}}\|^2)$  based on the distance between the predicted landing point and the goal on the x-y plane [7]. The landing position,  $\mathbf{p}^{\text{land}} = (x^{\text{land}}, y^{\text{land}})$ , can be calculated using the initial position and velocity as follows ( $g$  is gravity):

$$x_{\text{land}} = x_0 + v_{0,x} \left( \frac{v_{0,z} + \sqrt{v_{0,z}^2 + 2gz_0}}{g} \right), \quad y_{\text{land}} = y_0 + v_{0,y} \left( \frac{v_{0,z} + \sqrt{v_{0,z}^2 + 2gz_0}}{g} \right) \quad (6)$$

Early termination is triggered if the ball moves backward, does not contact the golf club within 2 seconds, is too close to the humanoid’s body, or the humanoid falls.

**Javelin.** For javelin throw, the humanoid is equipped with a javelin of length 2.7m. Due to the complexity introduced by articulated fingers, the reward function  $\mathcal{R}^{\text{javelin}}$  is applied in three stages: first, the humanoid learns to hold the javelin stably; then, it learns to throw it; finally, the javelin flies

as far as possible. A timer is used to differentiate the three stages. Specifically,  $\mathcal{R}^{\text{javelin}}$  is defined as follows:

$$\mathcal{R}^{\text{javelin}}(s_t^p, s_t^{g-\text{javelin}}) \triangleq \begin{cases} 0.9 \times r_t^{\text{grab}} + 0.1 \times r_t^{\text{js}} & \text{if } t < 0.6s, \\ 0.9 \times r_t^{\text{goal}} + 0.05 \times r_t^s - 0.05 \times r_t^{\text{grab}} & \text{if } 0.6s \leq t < 1.2s, \\ 0.9 \times r_t^{\text{goal}} + 0.1 \times r_t^{\text{js}} & \text{if } 1.2s \leq t. \end{cases} \quad (7)$$

The reward for grasping  $r_t^{\text{grab}} = \exp(-1 \times \|\mathbf{p}_t^{\text{right-hand}} - \mathbf{p}_t^{\text{javelin}}\|^2)$  encourages the hand to stay close to the javelin. The javelin stability reward  $r_t^{\text{js}} = \exp(-1 \times \|\mathbf{q}_t^{\text{javelin}} - \mathbf{q}_t^{\text{javelin-default}}\|^2)$  encourages the 6 DoF pose of the javelin to remain close to the default pose, which faces forward and tilts 30 degrees upward, mimicking a flying pose. The humanoid stability reward,  $r_t^s = \exp(-1 \times \|\mathbf{p}_t^{\text{root}}\|^2)$ , encourages the humanoid to keep its root position fixed. The termination conditions vary according to the stage: during the grasping and throwing stages, the episode terminates if the javelin is too far from the right hand or deviates significantly from the default pose  $\mathbf{q}_t^{\text{javelin-default}}$ . During the flying stage, termination occurs if the javelin is too close to the right hand.

## B.2 Multi-person Sports

**Tennis.** For tennis, each humanoid is equipped with a circular racket with a 15cm radius, positioned 35cm away from the wrist, replacing the right hand. The court measures 23.77m in length and 8.23m in width, mirroring the dimensions and layout of a real tennis court. The net height is 1m, and the simulated ball has a radius of 3.2cm. We design two tasks: a single-player ball return task, where the humanoid trains to hit balls launched randomly, and a 1v1 mode, where the humanoid competes against another humanoid. In the ball return task, the humanoid is positioned at the center of the baseline, with balls launched from the opposite side. The landing location is uniformly sampled on the opposite side and the ball launch velocity is randomly sampled. The reward function is defined as follows:

$$\mathcal{R}^{\text{tennis}}(s_t^p, s_t^{g-\text{tennis}}) \triangleq \begin{cases} 1 \times r_t^{\text{racket}} + 0 \times r_t^{\text{ball}}, & \text{if } C_{\text{rb}} = 0, \\ 0 \times r_t^{\text{racket}} + 1 \times r_t^{\text{ball}}, & \text{if } C_{\text{rb}} = 1. \end{cases} \quad (8)$$

Here,  $C_{\text{rb}} = 0$  indicates that the racket has not made contact with the ball, and  $C_{\text{rb}} = 1$  indicates the racket has made contact.  $r_t^{\text{racket}} = \exp(-1 \times \|\mathbf{p}_t^{\text{racket}} - \mathbf{p}_t^{\text{ball}}\|^2)$  rewards the racket for getting closer to the ball.  $r_t^{\text{ball}} = 1 + \exp(-1 \times \|\mathbf{p}_t^{\text{land}} - \mathbf{p}_t^{\text{tar}}\|^2)$  encourages the predicted landing location of the ball to be close to the target. Similar to the golf task, the landing location of the ball is calculated based on  $\mathbf{p}_t^{\text{ball}}$  and  $\mathbf{v}_t^{\text{ball}}$ , providing a dense reward function to facilitate training [7]. Early termination occurs if the humanoid loses the point, either by failing to catch the ball or by hitting the ball out of bounds. In the 1v1 mode, two humanoids are placed on opposite sides of the court and the first ball is launched from the middle of the court, randomly directed at each player. The same reward function as the ball return task is used. To facilitate 1v1 training, the pre-trained model from the ball return task is used as a warm start. Similarly, the episode terminates if one player fails to catch the ball or returns the ball out of bounds.

**Table Tennis.** For table tennis, each humanoid is equipped with a circular paddle with an 8 cm radius, positioned 12 cm from the wrist, replacing the right hand. The table adheres to standard dimensions, featuring a playing surface 2.74 m in length and 1.525 m in width, standing 0.76 m high. The net is 15.25 cm high, and the table tennis ball has a radius of 2 cm. The setup includes a single-player ball return task and a 1v1 task. The reward function is designed similarly to tennis, except we define the ball reward as  $r_t^{\text{ball}} = 1 + \exp(-1 \times \|\mathbf{p}_t^{\text{land}} - \mathbf{p}_t^{\text{tar}}\|^2) + N_{\text{hit}}$ , where  $N_{\text{hit}}$  counts the number of successful hits in one episode. This formulation is intended to encourage the humanoid to continuously hit the ball effectively. Unlike in golf and tennis, we calculate  $\mathbf{p}_t^{\text{land}}$  when it lands on the table at a height of 0.76 m. For early termination and the warm start in 1v1, we maintain the same setting as in the tennis task.

**Fencing.** For 1v1 fencing, similar to real-world fencing, the two players are confined to a 14m by 2m playground, where stepping out of the bound will reset the game. The fencing reward is structured

104 similarly to the boxing setup in NCP [8]:

$$\mathcal{R}^{\text{fencing}}(s_t^p, s_t^{g\text{-fencing}}) \triangleq 0.1 \times r_t^{\text{facing}} + 0.1 \times r_t^{\text{vel}} + 0.6 \times r_t^{\text{strike}} + 1 \times r_t^{\text{point}}. \quad (9)$$

105 The facing reward,  $r_t^{\text{facing}}$ , penalizes deviation from facing the opponent's root position  $p_t^{\text{opp-root}}$ . The  
 106 velocity reward,  $r_t^{\text{vel}}$ , encourages the x-y plane linear velocity to be directed towards the opponent's  
 107 root position  $p_t^{\text{opp-root}}$ . The strike reward,  $r_t^{\text{strike}} = \exp(-10 \times \argmin \|p_t^{\text{sword}} - p_t^{\text{opp-target}}\|^2)$ , encour-  
 108 ages the swordtip to get closer to the target body parts  $p_t^{\text{opp-target}}$ , which include the pelvis, head, spine,  
 109 chest, and torso. If there is contact with the target body part with sufficient force, a positive reward is  
 110 provided:

$$r_t^{\text{point}} = \begin{cases} 1 & \text{if } \argmin \|p_t^{\text{sword}} - p_t^{\text{opp-target}}\|^2 \leq 0.1 \text{ and contact force} \geq 50\text{Nm}, \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

111 Our fencing agents are trained using competitive self-play, as introduced in the main paper.

112 **Boxing.** For boxing, the humanoid competes in a boxing ring measuring 5m by 5m. The humanoid's  
 113 right hand is replaced with a sphere of 8cm radius. The boxing reward function has the same  
 114 composition as fencing, except that the sword tip position  $p_t^{\text{sword}}$  is replaced by the hand position  
 115  $p_t^{\text{hand}}$ . Our boxing agents are also trained using competitive self-play.

116 **Soccer.** The soccer field measures 32m in length and 20m in width. Each goal is 4m wide and 2m  
 117 tall. The ball has a diameter of 11.5 cm and weighs 450 grams. For the penalty kick task, the reward  
 118 function  $\mathcal{R}^{\text{soccer-kick}}(s_t^p, s_t^{g\text{-kick}}) \triangleq w^{\text{p2b}} r^{\text{p2b}} + w^{\text{b2g}} r^{\text{b2g}} + w^{\text{bv2g}} r^{\text{bv2g}} + w^{\text{b2t}} r^{\text{b2t}} - c_t^{\text{no-dribble}}$  is divided  
 119 into stages based on whether the ball is moving toward the goal. Specifically, we define a "closer to  
 120 goal" variable as  $g_t^{\text{ball-to-goal}} = \|p_t^{\text{goal-target}} - p_{t-1}^{\text{ball}}\|_2 - \|p_t^{\text{goal-target}} - p_t^{\text{ball}}\|_2$ , which indicates whether  
 121 the ball is getting closer to the goal. The full reward function is defined as follows:

$$\mathcal{R}^{\text{soccer-kick}}(s_t^p, s_t^{g\text{-kick}}) \triangleq \begin{cases} 0.4 \times r^{\text{p2b}} - c_t^{\text{no-dribble}} & \text{if } g_t^{\text{ball-to-goal}} \leq 0, \\ 0.1 \times r^{\text{b2g}} + 0.1 \times r^{\text{bv2g}} + 0.8 \times r^{\text{b2t}} - c_t^{\text{no-dribble}} & \text{otherwise.} \end{cases} \quad (11)$$

122 Essentially, if the ball is not moving toward the goal, the humanoid is encouraged to move toward the  
 123 ball; if the ball is moving, the agent is rewarded for shooting the ball toward the target in the goal  
 124 post. The player-to-ball reward,  $r^{\text{p2b}} = \|p_{t-1}^{\text{root}} - p_{t-1}^{\text{ball}}\|_2 - \|p_t^{\text{root}} - p_t^{\text{ball}}\|_2$ , is a point-goal reward [6].  
 125 The ball-to-goal reward  $r^{\text{b2g}} = \|p_t^{\text{goal-target}} - p_{t-1}^{\text{ball}}\|_2 - \|p_t^{\text{goal-target}} - p_t^{\text{ball}}\|_2$  encourages the ball to  
 126 move closer to the goal position. The ball-velocity-to-goal reward  $r^{\text{bv2g}}$  incentivizes the ball velocity  
 127 toward the goal position. The ball-to-target reward  $r^{\text{b2t}}$  predicts the landing position of the ball in  
 128 the net based on its current velocity and position, providing a reward if the ball is close to the target.  
 129 Finally,  $c_t^{\text{no-dribble}}$  penalizes the humanoid if its root position is over the ball's spawning point.

130 The team play (1v1 and 2v2) soccer tasks use similar rewards as the penalty kick task. The reward  
 131 function for team play is  $\mathcal{R}^{\text{soccer-match}}(s_t^p, s_t^{g\text{-soccer}}) \triangleq w^{\text{p2b}} r^{\text{p2b}} + w^{\text{b2g}} r^{\text{b2g}} + w^{\text{bv2g}} r^{\text{bv2g}} + w^{\text{point}} r^{\text{point}}$ ,  
 132 where  $r^{\text{p2b}}$ ,  $r^{\text{b2g}}$  are the same as in the penalty kick.  $r^{\text{point}}$  provides a one-time bonus for scoring.

133 **Basketball.** The basketball environment is similar to soccer except that it utilizes the SMPL-X  
 134 humanoid with articulated fingers. In the free-throwing task, the ball is initialized between the  
 135 humanoid's hands. The free throw reward is defined as:  $\mathcal{R}^{\text{free-throw}}(s_t^p, s_t^{g\text{-soccer}}) \triangleq 0.5 \times r^{\text{ballvel}} +$   
 136  $0.5 \times r^{\text{bv2g}} + r^{\text{basket}}$ . The basketball velocity reward  $r^{\text{ballvel}} = \exp(-0.1 \times \|v_t^{\text{ball}} - v_t^{\text{ball-desired}}\|_2^2)$   
 137 encourages the ball's velocity to be close to the desired velocity to reach the goal. The desired  
 138 velocity,  $v_t^{\text{ball-desired}}$ , is computed using the goal position  $p_t^{\text{goal-target}}$ , and the ball position  $p_t^{\text{ball}}$ , with  
 139 the following physics equations:

$$T_t^{\text{reach}} = \sqrt{\frac{2 \times \|(p_t^{\text{ball}} - p_t^{\text{goal-target}})_z\|_2}{g}}, \quad v_{t,xy}^{\text{ball-desired}} = \frac{\|(p_t^{\text{ball}} - p_t^{\text{goal-target}})_{xy}\|_2}{T_t^{\text{reach}}} \quad (12)$$

$$v_{t,z}^{\text{ball-desired}} = \frac{(p_t^{\text{ball}} - p_t^{\text{goal-target}})_z + 0.5 \times g \times (T_t^{\text{reach}})^2}{T_t^{\text{reach}}}.$$

140 The ball-velocity-to-goal reward  $r^{\text{bv2g}}$  encourages the velocity to be directed towards the goal position.  
 141 The basket reward,  $r^{\text{basket}}$ , provides a one-time reward if the ball passes through the basket.

Table 1: Hyperparameters for training each baseline used in SMPLOlympics. We use the same set of hyperparameters for *each sport*. Notice that AMP and PULSE uses PPO as the optimization method but add respective motion priors (as reward or motion representation).  $\sigma$ : fixed variance for policy.  $\gamma$ : discount factor.  $\epsilon$ : clip range for PPO.  $w_{\text{disc}}$  and  $w_{\text{task}}$ : weights for discriminator and task rewards.

	Batch Size	Learning Rate	$\sigma$	$\gamma$	$\epsilon$	MLP-size	$w_{\text{disc}}$	$w_{\text{task}}$	# of samples
PPO [4]	1024	$5 \times 10^{-4}$	0.05	0.99	0.2	[2048, 1024, 512]	0	1	$\sim 10^9$
AMP [3]	1024	$5 \times 10^{-4}$	0.05	0.99	0.2	[2048, 1024, 512]	0.5	0.5	$\sim 10^9$
PULSE [1]	1024	$5 \times 10^{-4}$	0.3	0.99	0.2	[2048, 1024, 512]	0	1	$\sim 10^9$
PULSE [1] + AMP [3]	1024	$5 \times 10^{-4}$	0.3	0.99	0.2	[2048, 1024, 512]	0.5	0.5	$\sim 10^9$

Team-play basketball has a similar reward design as soccer. The team-play basketball task is highly challenging due to the difficulty of picking the ball up, which is more complex than kicking a ball. Thus, while we support 1v1 and 2v2 team-play basketball, our preliminary reward design does not yield interesting behavior, unlike in soccer.

### B.3 Hyperparamters

Training hyperparameters are provided in Table 1. We use the same set of hyperparameters to train *all* of our sports environments, highlighting the advantage of employing a unified humanoid embodiment for simulated sports.

### B.4 Details about Baselines

For our baseline methods PULSE [1] and AMP [3], we use the official implementations. For PULSE [1], we employ the publicly released model without modification, which is pre-trained on the AMASS dataset. We follow a similar setup for downstream tasks in PULSE, using the frozen prior  $\mathcal{P}_{\text{PULSE}}$ , decoder  $\mathcal{D}_{\text{PULSE}}$ , and residual action representation. Since PULSE only includes trained models for the SMPL-based models, we train SMPL-X humanoid based models following the official code. Specifically, we train a humanoid motion imitator following PHC [2], and distill motor skills into a 48-dimensional latent space (instead of 32-D, to accommodate articulated fingers). PULSE provides an action space for hierarchical RL and can be integrated with AMP. For PULSE+AMP, the AMP reward offers additional style guidance for the humanoid, which is particularly beneficial for tasks such as table tennis. However, we find that the demonstration sequences used for AMP need to be task-specific (*e.g.* contains only a swinging motion); otherwise, the discriminator reward can overpower the task reward and lead to undesired behavior (as seen in the free kick results).

## C Additional Ablations

We conducted an ablation study to evaluate the role of physics-based tracking (w/ PHC) in acquiring human reference motion. Specifically, we used the pose estimation results directly from TRAM [5] as positive samples for the discriminator during policy training (w/ PHC). Our experiments were performed in the context of table tennis. As shown in Table 2, we found that providing video data without PHC leads to significantly lower performance compared to using PHC, similar to the results obtained using only PULSE. We observe that when the quality of the provided reference motion is poor (*e.g.*, with significant noise in position, and drastic velocity changes), the model struggles to effectively utilize the reference motion as style guidance to achieve natural movements. In contrast, employing physics-based tracking to refine pose estimates from in-the-wild videos results in physically plausible motion, which significantly aids in policy learning.

Table 2: Ablation study on PHC.

Table Tennis		
Method	Avg Hits $\uparrow$	Error Dis $\downarrow$
PULSE	0.74	0.19
PULSE+AMP, w/o PHC	0.91	<b>0.18</b>
PULSE+AMP, w/ PHC	<b>1.83</b>	0.23

## D Broader Social impact

We propose SMPLOlympics, a collection of sports environments for simulated humanoids. These environments can be used to benchmark learning algorithms, discover new humanoid behaviors, create animations, and more. The potential negative social impact includes the risk of generating animations that could be used to create DeepFakes. Positive social impact includes the development of intelligent and collaborative agents, advancements in robot learning, discovery of new sports techniques, and the generation of immersive and physically realistic animations.

## References

- [1] Zhengyi Luo, Jinkun Cao, Josh Merel, Alexander Winkler, Jing Huang, Kris Kitani, and Weipeng Xu. Universal humanoid motion representations for physics-based control. *arXiv preprint arXiv:2310.04582*, 2023.
- [2] Zhengyi Luo, Jinkun Cao, Alexander W. Winkler, Kris Kitani, and Weipeng Xu. Perpetual humanoid control for real-time simulated avatars. In *International Conference on Computer Vision (ICCV)*, 2023.
- [3] Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. Amp: Adversarial motion priors for stylized physics-based character control. *ACM Trans. Graph.*, pages 1–20, 2021.
- [4] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL <https://api.semanticscholar.org/CorpusID:28695052>.
- [5] Yufu Wang, Ziyun Wang, Lingjie Liu, and Kostas Daniilidis. Tram: Global trajectory and motion of 3d humans from in-the-wild videos. *arXiv preprint arXiv:2403.17346*, 2024.
- [6] Jungdam Won, Deepak Gopinath, and Jessica Hodgins. Physics-based character controllers using conditional vaes. *ACM Trans. Graph.*, 41:1–12, 2022. ISSN 0730-0301.
- [7] Haotian Zhang, Ye Yuan, Viktor Makoviychuk, Yunrong Guo, Sanja Fidler, Xue Bin Peng, and Kayvon Fatahalian. Learning physically simulated tennis skills from broadcast videos. *ACM Trans. Graph.*, 42:1–14, 2023. ISSN 0730-0301,1557-7368.
- [8] Qingxu Zhu, He Zhang, Mengting Lan, and Lei Han. Neural categorical priors for physics-based character control. *arXiv preprint arXiv:2308.07200*, 2023.