

Learning to Assist: Physics-Grounded Human-Human Control via Multi-Agent Reinforcement Learning

Supplementary Materials

Contents

A Overview of the Supplementary Materials	1
B Additional Qualitative comparison	1
C COM Stability Metric	2
D Additional implementation details	3
E Detailed reward breakdown	4
E.1. Overall reward structure	4
E.2. Task reward decomposition	4
E.3. Caregiver – recipient coupling	5
F. Detailed information about the diffusion planner	5
F.1. Overview	5
F.2. Data representation	6
F.3. Model architecture	6
F.4. Sampling	7

A. Overview of the Supplementary Materials

This supplementary document contains additional details and discussions of our *AssistMimic*. Please also refer to the supplementary video [assistmimic-video.mov](#), which provides an overview of our task setup and problem background, as well as motion-tracking visualizations of the learned assistive behaviors. We highlight reference numbers associated with the main paper in [blue](#), and those associated with this supplementary document in [red](#).

B. Additional Qualitative comparison

Figure 1 visualizes the ablation results of the specialist policies on the Inter-X dataset. Each row corresponds to a different variant, and frames are shown from left to right in chronological order. The first and second rows compare the best AssistMimic policy (w/o dynamic reference retargeting) and the variant without the contact-promoting reward on the *same* help-up motion. The third row shows two separate examples for the variant without weight initialization, because the humanoids quickly lose balance and the episodes terminate early.

From the comparison between the first and second rows, especially in the zoomed-in regions highlighted by the red boxes, we can observe that introducing the contact-promoting reward enables the supporter to discover a correct assistive strategy even under noisy reference motions. In the second row, the supporter over-follows the noisy hand trajectory and ends up pressing down on the recipient from above, which leads to the recipient’s fall. In contrast, the best model maintains stable, supportive contact around the upper body and produces a more realistic help-up behavior.

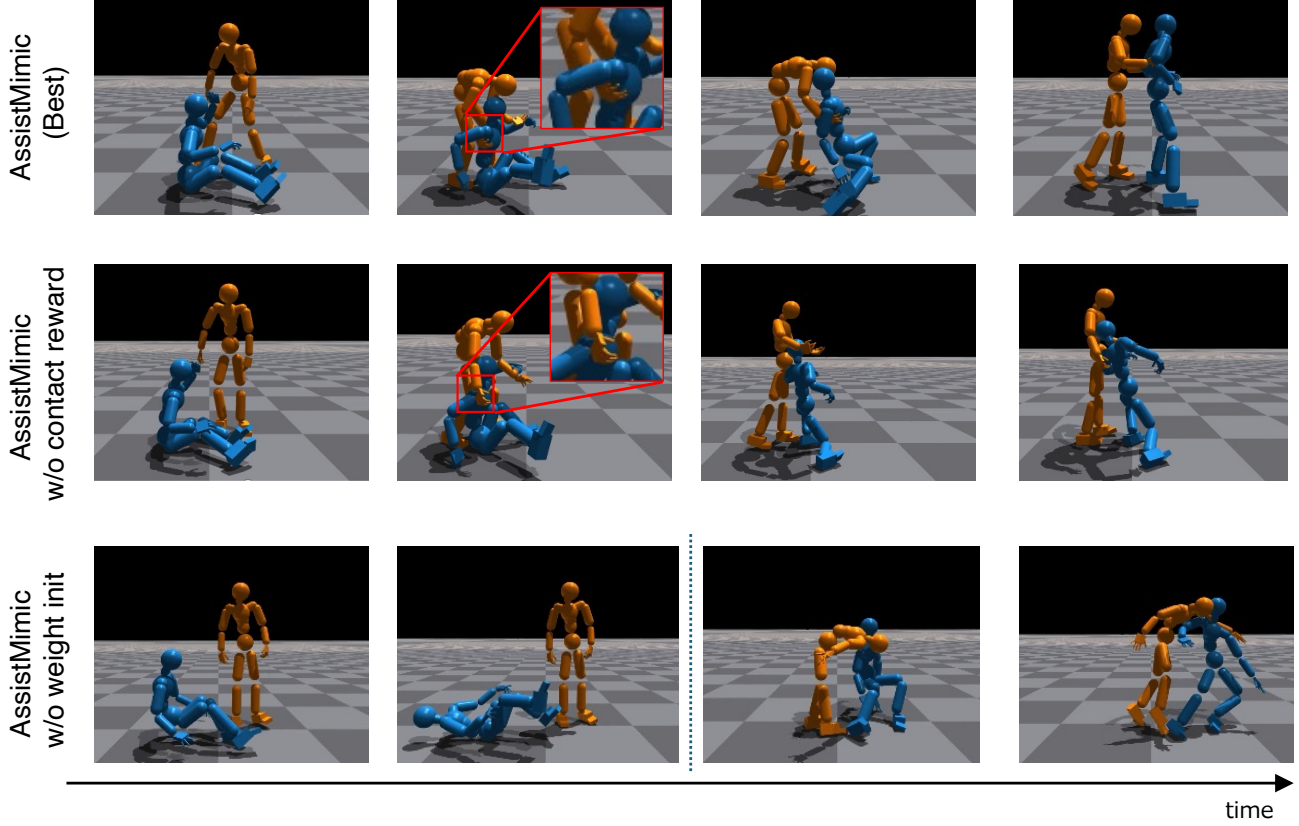


Figure 1. The qualitative comparison of the specialist with Inter-X dataset. Supporter (orange) and Recipient (blue). Left to Right in chronological order.

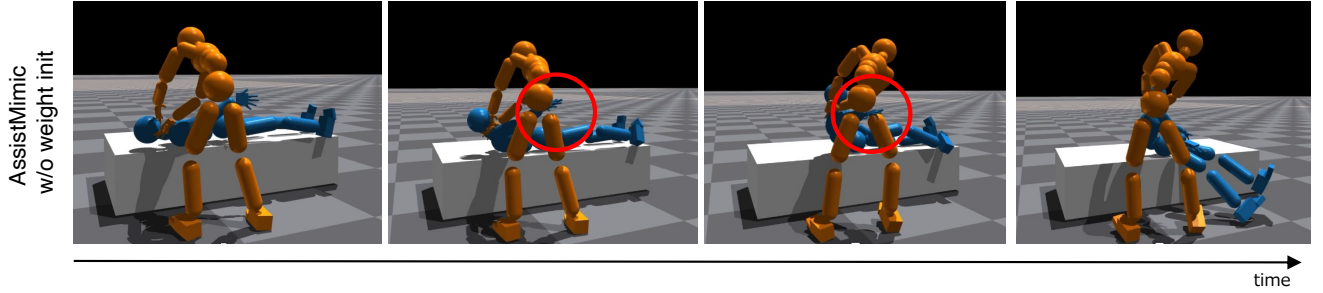


Figure 2. The qualitative comparison with HHI-Assist dataset. Supporter (orange) and Recipient (blue).

For the variant without weight initialization (third row), training fails to progress: near-floor motions cause the characters to lose balance almost immediately in some cases (first and second columns), or, in other cases (third and fourth columns), the supporter approaches and reaches out but then leans its body weight onto the recipient, causing both agents to fall.

Figure 2 shows the results on the HHI-Assist dataset when PHC is *not* used for weight initialization. Compared to the behavior in Figure 4 of the main paper, the recipient here largely ignores the intended hand-tracking objective and instead touches the supporter’s waist, using the resulting reaction to lift its upper body. As also reflected by the low success rate in Table 3, even the episodes that are counted as successful are dominated by such reward-hacking behaviors, indicating that meaningful assistive strategies are not learned in this setting.

C. COM Stability Metric

To quantify the stability of assistive interactions, we measure the temporal variation of the recipient’s center of mass (COM) during each episode.

Table 1. Recipient COM standard deviation (\downarrow)

Model	Seen	Mass $\times 1.5$	Max hip $\tau \times 0.5$
Ours	0.0921	0.0738	0.0865
(-) Dyn Retar	0.1038	0.0902	0.0924
(-) Cont	0.0938	0.0838	0.0849

Definition. Let $\mathbf{c}_t \in \mathbb{R}^3$ denote the 3D position of the recipient’s COM at timestep t , computed from the mass-weighted average of all body segments. We define the COM stability metric as the standard deviation of \mathbf{c}_t over time:

$$\sigma_{\text{COM}} = \sqrt{\frac{1}{T} \sum_{t=1}^T \|\mathbf{c}_t - \bar{\mathbf{c}}\|^2}, \quad \bar{\mathbf{c}} = \frac{1}{T} \sum_{t=1}^T \mathbf{c}_t, \quad (1)$$

where T is the number of timesteps in the episode.

Evaluation protocol. To ensure a fair comparison, we compute σ_{COM} only on motion sequences where all compared methods successfully complete the task. This avoids bias due to early termination or failure cases, which would otherwise result in shorter trajectories and artificially lower variance.

Interpretation. A lower σ_{COM} indicates more stable assistance, as the recipient’s body is maintained with less oscillation or unintended movement. This metric is particularly relevant for bed-based caregiving scenarios (e.g., HHI-Assist), where maintaining a steady posture is critical.

Results. As summarized in Table 1, AssistMimic achieves lower recipient COM standard deviation compared to the ablated variants on the HHI-Assist when evaluated on motions successfully completed by all policies. Together with the results in Table 3, this indicates that our approach achieves both high success rates and stable assistance.

D. Additional implementation details

Optimization setup. For the multi-agent RL optimization, we use Proximal Policy Optimization (PPO) [7] for both the supporter and the recipient policies. For the specialist comparisons in Table 2, 3, policies are trained for 2k iterations, which we found sufficient for these focused subsets. For the generalist evaluation in Table 4, our AssistMimic w/o DAGger (1st line) is trained for 9k iterations. Similarly, we train specialist teacher policies for 9k iterations to provide high-fidelity supervision, from which the final generalist policy is distilled via DAGger [5] over 4k epochs. The learning rate for both agents is set to 5×10^{-6} , and we apply a step decay by a factor of 0.1 once training reaches 600 PPO iterations. The critic network is implemented as an MLP that takes the same input as the actor, augmented with a role label indicating whether the agent is the *recipient* or the *caregiver* (supporter).

Motion prior and fine-tuning. AssistMimic inherits weights from a single-human PHC tracking controller [1] through our weight-initialization scheme. However, motions of the recipient near the floor are largely out-of-domain for the original PHC model trained on AMASS [2]. To address this, we perform a small-scale fine-tuning stage using only recipient motion data. After this stage, we filter out clips that can already be successfully tracked by the single-human PHC rollout alone, since such motions do not require meaningful support and should not be treated as assistive examples. Unlike the original PHC work, which trains multiple motion primitives, all of our experiments are conducted with a *single* primitive.

Generalist training. As discussed in the ablation study in the main paper, the dynamic reference retargeting module did not provide benefits on the Inter-X dataset. Therefore, for training the Inter-X generalist policy, we disable this module and first train four specialist policies, each on a subset of Inter-X help-up motions. We then perform online policy distillation using DAGger [5], where the generalist policy is trained to imitate the actions of these specialists. For this distillation stage, we optimize the generalist for 1k epochs using a squared-error loss between the generalist and specialist actions.

Evaluation protocol for the HHI-Assist bed setting. In many HHI-Assist bed sequences, the supporter steps away from the recipient near the end of the motion after the assistance is completed. Because the recipient is no longer supported during this phase, falls occurring in this stage should not be considered assistive failures. Thus, for quantitative evaluations (Table 3, Table 1), we discard the final 15 frames of each 30 FPS motion clip and compute all metrics on the remaining frames.

Caregiving on a chair. The chair-based caregiving behaviors shown in Figure 1 of the main paper are trained using reference motions from the HHI-Assist dataset. During this training, we apply the same physical limitations to the recipient as those listed in Table 1, analogous to the bed-care setting.

E. Detailed reward breakdown

In this section, we describe the reward details that could not be included in the main paper due to space limitations.

E.1. Overall reward structure

Our method builds upon the Adversarial Motion Priors (AMP) [4] framework. The per-timestep reward for agent $m \in \{S, R\}$ (supporter/recipient) combines a task reward and a discriminator reward:

$$r_t^{(m)} = \lambda_{\text{task}} r_{\text{task},t}^{(m)} + \lambda_{\text{disc}} r_{\text{disc},t}, \quad (2)$$

where $r_{\text{disc},t}$ is the AMP discriminator reward that encourages physically plausible motion, and $\lambda_{\text{task}}, \lambda_{\text{disc}} > 0$ are scalar weights (set to 0.5 each in our experiments).

E.2. Task reward decomposition

The task reward is further decomposed into tracking, power penalty, and assistive terms:

$$r_{\text{task},t}^{(m)} = \lambda_{\text{track}} r_{\text{track},t}^{(m)} + \lambda_{\text{power}} r_{\text{power},t}^{(m)} + \lambda_{\text{assist}} r_{\text{assist},t}^{(m)}, \quad (3)$$

where $\lambda_{\text{track}}, \lambda_{\text{power}}, \lambda_{\text{assist}} > 0$ are scalar weights.

Tracking reward. The tracking reward measures how closely agent m follows its reference motion:

$$r_{\text{track},t}^{(m)} = \frac{1}{J} \sum_{j=1}^J \exp(-k_{\text{track}} \cdot d_j(\hat{\mathbf{q}}_{j,t}^{(m)}, \mathbf{q}_{j,t}^{(m)})), \quad (4)$$

where J is the number of joints, $d_j(\cdot, \cdot)$ is the distance metric for joint j (combining position and rotation differences), $\hat{\mathbf{q}}_{j,t}^{(m)}$ is the current state, $\mathbf{q}_{j,t}^{(m)}$ is the reference state, and k_{track} is a scaling factor.

Power penalty. The power penalty discourages excessive joint actuation by penalizing the mechanical power:

$$r_{\text{power},t}^{(m)} = -\lambda_{\text{power}}^{(m)} \sum_{j=1}^J |\tau_{j,t}^{(m)} \cdot \dot{q}_{j,t}^{(m)}|, \quad (5)$$

where $\tau_{j,t}^{(m)}$ is the torque applied to joint j at time t , $\dot{q}_{j,t}^{(m)}$ is the joint velocity, and $\lambda_{\text{power}}^{(S)} = 0.0015$ for the supporter and $\lambda_{\text{power}}^{(R)} = 0.002$ for the recipient.

Assistive reward. The assistive term encourages effective support:

$$r_{\text{assist},t}^{(m)} = \alpha_{\text{head}} r_{\text{head},t}^{(R)} + \alpha_{\text{torque}} r_{\text{torque},t}^{(R)}, \quad (6)$$

where $r_{\text{head},t}^{(R)} = \min(h_t^{(R)}, h_{\text{max}}^{(R)})$ rewards a higher head height of the recipient (clamped by the maximum achievable height $h_{\text{max}}^{(R)}$), and $r_{\text{torque},t}^{(R)} = \exp(-\|\boldsymbol{\tau}_t^{(R)}\|_1 / \sigma_{\text{torque}})$ rewards reductions in the recipient's joint torques.

Category	Symbol	Description	Value
Overall reward	λ_{task}	Weight for task reward $r_{\text{task},t}^{(m)}$ in the per-timestep reward	0.5
	λ_{disc}	Weight for discriminator reward $r_{\text{disc},t}$ from AMP.	0.5
Task reward decomposition	λ_{track}	Weight for tracking term	1.0
	λ_{assist}	Weight for assistive term $r_{\text{assist},t}^{(m)}$.	1.0
	k_{track}	Scaling factor in the exponential tracking reward	100
	$\lambda_{\text{power}}^{(S)}$	Power penalty coefficient for the supporter	0.0015
	$\lambda_{\text{power}}^{(R)}$	Power penalty coefficient for the recipient	0.002
Assistive reward	α_{head}	Weight for head-height term $r_{\text{head},t}^{(R)}$ in the assistive reward	1.0
	α_{torque}	$r_{\text{assist},t}^{(m)} = \alpha_{\text{head}} r_{\text{head},t}^{(R)} + \alpha_{\text{torque}} r_{\text{torque},t}^{(R)}$ Weight for torque-reduction term $r_{\text{torque},t}^{(R)}$.	0.0 (Inter-X) 0.5 (HHI-Assist)
	$h_{\text{max}}^{(R)}$	Normalization constant for recipient head height in $r_{\text{head},t}^{(R)} = \min(h_t^{(R)} / h_{\text{max}}^{(R)}, 1.0)$.	2.0
	σ_{torque}	Scaling factor in the torque reduction term $r_{\text{torque},t}^{(R)} = \exp(-\ \tau_t^{(R)}\ _1 / \sigma_{\text{torque}})$.	150
Caregiver–recipient coupling	$\frac{1}{2}$ (mixing weight)	Mixing coefficient in the final supporter reward $r_t^{(S)} = \frac{1}{2} \tilde{r}_t^{(S)} + \frac{1}{2} \tilde{r}_t^{(R)}$.	
Dynamic hand reference retargeting	τ_{dist}	Distance threshold for activating reference retargeting in the gating term $G_t = \mathbb{I}(\ p_{\text{root},t}^{(S)} - p_{\text{root},t}^{(R)}\ _2 \leq \tau_{\text{dist}})$.	1.3
Contact-promoting reward (supporter)	d_{th}	Distance threshold for proximity indicator $\chi_{i,t} = \mathbb{I}(d_{i,t} \leq d_{\text{th}})$, where $d_{i,t}$ is the minimum distance between supporter wrist i and recipient upper-body joints.	0.4
	α	Distance scaling factor in the contact term $\beta f_{i,t} \exp(-\alpha d_{i,t})$.	2.5
	β	Force scaling factor in the contact term $\beta f_{i,t} \exp(-\alpha d_{i,t})$.	0.5
	b_{contact}	Sparse bonus added when contact is established in the contact-promoting reward.	0.05
	f_{th}	Force threshold in the saturated contact-force aggregation $f_{i,t} = \sum_{\ell \in H_i \setminus \{i\}} \min(\exp(\ f_{\ell,t}\ _2 - f_{\text{th}}), 1)$.	1.0

Table 2. Hyperparameters used in the reward design and dynamic hand reference retargeting.

E.3. Caregiver – recipient coupling

We first compute the per-agent rewards $\tilde{r}_t^{(S)}$ and $\tilde{r}_t^{(R)}$ according to Eqs. (2)–(6). The final rewards used for optimization are then given by

$$r_t^{(R)} = \tilde{r}_t^{(R)}, \quad r_t^{(S)} = \frac{1}{2} \tilde{r}_t^{(S)} + \frac{1}{2} \tilde{r}_t^{(R)}, \quad (7)$$

so that the caregiver (supporter) is explicitly encouraged to maximize not only its own reward but also the overall reward of the recipient.

Table 2 summarizes the hyperparameters for the proposed contact-promoting reward and dynamic reference retargeting, as well as the additional reward terms introduced in this section.

F. Detailed information about the diffusion planner

F.1. Overview

Our diffusion planner is a denoising diffusion transformer that auto-regressively generates joint positions for both the supporter and the recipient, conditioned on text descriptions. During inference, we prompt the model with unseen descriptions from the ”Help-up” category to synthesize novel motion sequences, which are subsequently tracked using AssistMimic.

F.2. Data representation

We train the planner using motion sequences from the Inter-X dataset labeled as "Help-up" (a small subset is held out for testing). Each sequence contains SMPL-X parameters for both individuals along with a corresponding text description. These sequences are converted into a representation suitable for diffusion-based generation.

Before constructing the motion representation, each motion sequence is preprocessed to ensure a canonical orientation. Specifically, the average displacement vector between the root joints of the supporter and recipient over the sequence is computed, and the entire motion is rotated such that this vector is aligned with the x -axis.

For each individual at frame i (subscripts c for supporter and r for recipient), we extract the joint positions in the global-frame via forward kinematics:

$$\mathbf{p}_c^i, \mathbf{p}_r^i \in \mathbb{R}^{22 \times 3}.$$

Joint velocities are computed using finite differences:

$$\mathbf{v}_c^i = \mathbf{p}_c^i - \mathbf{p}_c^{i-1}, \quad \mathbf{v}_r^i = \mathbf{p}_r^i - \mathbf{p}_r^{i-1} \in \mathbb{R}^{22 \times 3}.$$

The left and right-hand SMPL-X axis-angle poses are converted to the continuous 6D representation, yielding

$$\mathbf{h}_{c,L}^i, \mathbf{h}_{c,R}^i, \mathbf{h}_{r,L}^i, \mathbf{h}_{r,R}^i \in \mathbb{R}^{15 \times 6}.$$

Thus, the motion feature vector at frame i for each individual is

$$\mathbf{x}_c^i = [\mathbf{p}_c^i, \mathbf{v}_c^i, \mathbf{h}_{c,L}^i, \mathbf{h}_{c,R}^i] \in \mathbb{R}^{312},$$

$$\mathbf{x}_r^i = [\mathbf{p}_r^i, \mathbf{v}_r^i, \mathbf{h}_{r,L}^i, \mathbf{h}_{r,R}^i] \in \mathbb{R}^{312}.$$

Concatenating both individuals produces the full representation:

$$\mathbf{x}^i = [\mathbf{x}_c^i, \mathbf{x}_r^i] \in \mathbb{R}^{624}.$$

A motion window is defined as

$$\mathbf{x}^{i:i+N} = \{\mathbf{x}^i, \mathbf{x}^{i+1}, \dots, \mathbf{x}^{i+N-1}\}.$$

The accompanying text description is encoded using a frozen DistilBERT encoder [6], producing

$$\mathbf{z}_{\text{text}} \in \mathbb{R}^{N_{\text{tokens}} \times 768}.$$

F.3. Model architecture

Our implementation of the diffusion planner G is based on MDM [8]. It is trained to predict the clean motion window $\hat{\mathbf{x}}_0^{i:i+N}$ from a noisy version $\mathbf{x}_t^{i:i+N}$, conditioned on the previous window and the text embedding. The diffusion timestep is denoted by t :

$$\hat{\mathbf{x}}_0^{i:i+N} = G(\mathbf{x}_t^{i:i+N}, t, \mathbf{x}^{i-N:i}, \mathbf{z}_{\text{text}}).$$

During training, the windows $\mathbf{x}^{i-N:i}$ and $\mathbf{x}_t^{i:i+N}$ are concatenated and processed with self-attention, while the text embedding is incorporated via cross-attention in the transformer decoder layers. The motion windows along with the text embedding are all projected to the same latent dimension to be compatible with the attention layers.

We optimize the standard $\mathcal{L}_{\text{simple}}$ loss together with a loss for temporal consistency \mathcal{L}_{vel} , weighted by $\lambda_{\text{vel}} = 25$:

$$\mathcal{L}_{\text{vel}} = \frac{1}{N-1} \sum_{i=1}^{N-1} \|(\mathbf{x}_0^{i+1} - \mathbf{x}_0^i) - (\hat{\mathbf{x}}_0^{i+1} - \hat{\mathbf{x}}_0^i)\|_2^2.$$

$$\mathcal{L} = \mathcal{L}_{\text{simple}} + \lambda_{\text{vel}} \mathcal{L}_{\text{vel}}.$$

Specifically, our model uses a transformer decoder with 10 layers and 8 attention heads, a feedforward dimension of 2048, and a latent dimension of 768, with dropout set to 0.1. The prediction window is $N = 20$ frames. During training, the conditioning variables are masked with a probability of 0.1 to enable unconditional generation. The diffusion process uses 100 timesteps with a cosine beta schedule. We optimize the model using AdamW with a learning rate of 2×10^{-4} , betas (0.9, 0.95), and weight decay 10^{-3} . Training is performed for 8 hours on an NVIDIA A6000 GPU.

F.4. Sampling

During inference, the model is given a previously unseen text description of the intended motion and auto-regressively generates 180 frames in chunks of 20 frames. Each subsequent window is conditioned on the text description and the 20 frames generated in the previous step. The first 20 frames are generated using only the provided text as conditioning.

Finally, the SMPL-X parameters are recovered from the diffusion output via inverse kinematics using VPoser [3]. This generated motion sequence is then tracked using AssistMimic.

References

- [1] Luo, Z., Cao, J., Winkler, A., Kitani, K., Xu, W.: Perpetual humanoid control for real-time simulated avatars. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 10895–10904 (2023)
- [2] Mahmood, N., Ghorbani, N., Troje, N.F., Pons-Moll, G., Black, M.J.: Amass: Archive of motion capture as surface shapes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 5442–5451 (2019)
- [3] Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3d hands, face, and body from a single image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10975–10985 (2019)
- [4] Peng, X.B., Ma, Z., Abbeel, P., Levine, S., Kanazawa, A.: Amp: Adversarial motion priors for stylized physics-based character control. *ACM Transactions on Graphics* **40**(4), 1–20 (2021)
- [5] Ross, S., Gordon, G., Bagnell, D.: A reduction of imitation learning and structured prediction to no-regret online learning. In: Proceedings of the fourteenth international conference on artificial intelligence and statistics. pp. 627–635. JMLR Workshop and Conference Proceedings (2011)
- [6] Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv abs/1910.01108* (2019)
- [7] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017)
- [8] Tevet, G., Raab, S., Gordon, B., Shafir, Y., Cohen-or, D., Bermano, A.H.: Human motion diffusion model. In: International Conference on Learning Representations (2023), <https://openreview.net/forum?id=SJ1kSyO2jwu>