ACE: Adversarial Correspondence Embedding for Cross Morphology Motion Retargeting from Human to Nonhuman Characters

TIANYU LI, Georgia Tech, USA
JUNGDAM WON, Seoul National University, South Korea
ALEXANDER CLEGG, Meta AI, USA
JEONGHWAN KIM, Georgia Tech, USA
AKSHARA RAI, Meta AI, USA
SEHOON HA, Georgia Tech, USA

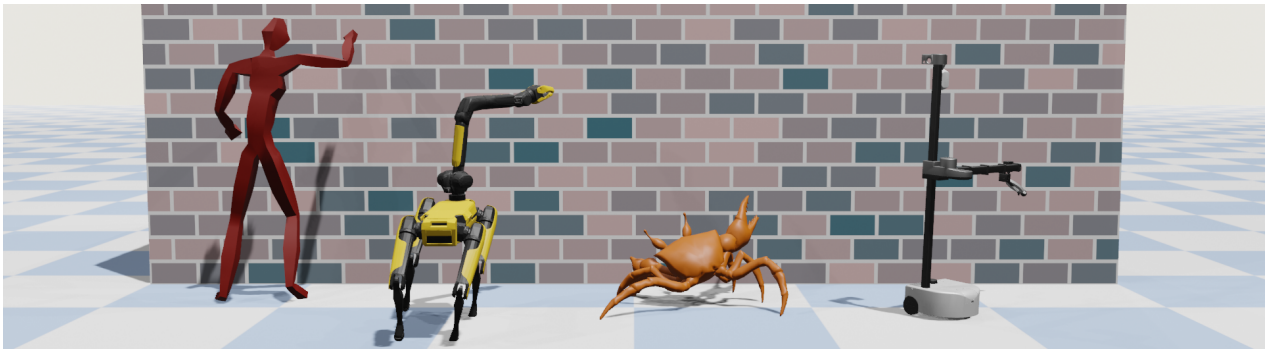


Fig. 1. We propose a motion retargeting framework, Adversarial Correspondence Embedding (ACE), to retarget human motions to characters with significantly different morphologies. This figure illustrates the retargeted wall-washing motions from human (1st from left) to Spot (2nd), Crab (3rd), and Stretch (4th).

Motion retargeting is a promising approach for generating natural and compelling animations for nonhuman characters. However, it is challenging to translate human movements into semantically equivalent motions for target characters with different morphologies due to the ambiguous nature of the problem. This work presents a novel learning-based motion retargeting framework, Adversarial Correspondence Embedding (ACE), to retarget human motions onto target characters with different body dimensions and structures. Our framework is designed to produce natural and feasible character motions by leveraging generative-adversarial networks (GANs) while preserving high-level motion semantics by introducing an additional feature loss. In addition, we pretrain a character motion prior that can be controlled in a latent embedding space and seek to establish a compact correspondence. We demonstrate that the proposed framework can produce retargeted motions for three different characters – a quadrupedal robot with a manipulator, a crab character, and a wheeled manipulator. We further validate the design choices of our framework by conducting baseline comparisons and a user study. We also showcase sim-to-real transfer of the retargeted motions by transferring them to a real Spot robot.

CCS Concepts: • **Computing methodologies** → **Procedural animation**; *Motion processing*; • **Computer systems organization** → Robotics.

Additional Key Words and Phrases: character animation, motion retargeting, adversarial learning

ACM Reference Format:
Tianyu Li, Jungdam Won, Alexander Clegg, Jeonghwan Kim, Akshara Rai, and Sehoon Ha. 2023. ACE: Adversarial Correspondence Embedding for Cross Morphology Motion Retargeting from Human to Nonhuman Characters. In *SIGGRAPH Asia 2023 Conference Papers (SA Conference Papers '23), December 12–15, 2023, Sydney, NSW, Australia.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3610548.3618255

SA Conference Papers '23, December 12–15, 2023, Sydney, NSW, Australia

ACM ISBN 979-8-4007-0315-7/23/12…$15.00
https://doi.org/10.1145/3610548.3618255

1 INTRODUCTION

Animating non-human characters has been a longstanding topic of discussion in computer graphics. Various animation films, movies, and computer games feature beloved characters with various morphologies inspired by everyday objects (e.g., Lumière [Disney 1991]), animals (e.g., Sebastian [Disney 1989]), or imaginary robotic designs (e.g., Wall-E [Pixar 2008], R2D2 [Lucasfilm 1977], and Omnics [Blizzard 2016]). While moving in their own distinctive styles, these characters still need to move somewhat "human-like" to convey human-interpretable semantics. It is possible to use manual design, optimal control, or reinforcement learning approaches to develop motion controllers, but this process may require great manual effort from experts and take multiple iterations to produce effective and human-understandable motions, even for skilled animators.

A more direct way to empower these non-human characters with diverse movements is to translate human motions. This motion retargeting approach not only simplifies the motion design process by avoiding complex cost or reward engineering but also has the

potential to make motions human-interpretable. However, it is not straightforward to adapt motions between characters with very different morphologies due to ambiguity and feasibility issues. For instance, which hand of a human should be mapped onto the single manipulator of a quadrupedal robot? Which types of gaits should be used when the character is following a human's walking pace? It is important to note that this is a question of style, and there is no single right answer. Even worse, some human motions are impossible for characters due to different body dimensions and structures and may cause weird dynamics or self collisions. As a result, there are fewer works that address cross-morphology motion retargeting from human to non-human characters compared to the extensive body of work on human-to-human motion mapping.

The problem of motion retargeting can be approached using a range of methods. Optimization-based motion approaches [Abdul-Massih et al. 2017] allow us to retarget a given motion to a new character by minimizing an objective function, but they may need to be carefully tuned to take different characters or scenarios into consideration. On the other hand, data-driven approaches are able to establish implicit relationships and generalize to large scenarios. Supervised learning offers users the option to build an explicit relationship from a paired dataset [Kim et al. 2022], but it requires precise matching of the motions between the source and target characters, which can be labor-intensive and time-consuming. On the other hand, researchers have demonstrated that recent advances in unsupervised learning can translate images [Zhu et al. 2017] and language [Rashid et al. 2019] across domains. Our work is motivated by these recent advances, where motion retargeting can be formulated as a translation problem between motions existing in two different domains. While the prior works [Aberman et al. 2020] investigated adversarial learning for motion retargeting between similar humanoids, it has not been extensively investigated in the context of cross-morphology motion retargeting.

In this work, we present Adversarial Correspondence Embedding (ACE), a learning-based motion retargeting framework that can translate human motions to a non-human character with significant morphological differences. The goal of our framework is to generate natural, feasible, and semantics-preserving character motions for given human motions. To this end, we build our framework on top of adversarial learning, which simultaneously trains a generator that retargets the given motion and a discriminator that evaluates the naturalness of the generated motion. We further guide the learning process by introducing an additional feature loss that preserves semantic features from the source human motion. We also pretrain motion priors that control the character's motions using a latent embedding, allowing the generator to learn a compact mapping to this latent embedding space instead of the full state of the character.

We demonstrate that our proposed ACE framework can retarget various human motions to three very different morphologies, including a quadrupedal robot with a manipulator (Spot [BostonDynamics 2019]), a crab character that uses two legs as manipulators, and a mobile robot with a telescopic manipulator (Stretch [HelloRobot 2023]). Across such a large range of scenarios, our framework generates compelling retargeted motions that look smooth, natural, and feasible on the target character. We also compare our proposed framework against several baseline approaches through multiple objective metrics and also by conducting a user study. We further demonstrate the flexibility of our work by retargeting motions with different end-effector mappings. Finally, we showcase the sim-to-real transfer of the retargeted motions to a real Spot robot to highlight the physical validity of the proposed method.

2 RELATED WORKS

2.1 Motion Retargeting

Motion retargeting is one of the long-standing challenges in computer animation. One common approach is to formulate it as an optimization problem with different constraints on kinematic properties [Gleicher 1998], end-effector motions [Choi and Ko 2000] or dynamics feasibility [Tak and Ko 2005]. Recently, a differentiable optimal control method, DOC, has been proposed [Grandia et al. 2023] to retarget motion capture data to real robots with various proportions and mass distribution. Although these methods can synthesize natural motions for new characters, the design of objectives and constraints often requires a labor-intensive process.

As large mocap datasets become accessible, data-driven motion retargeting approaches have been proposed. Some researchers [Delhaisse et al. 2017; Jang et al. 2018] train the retargeting function through a small set of paired data via supervised learning, then generalize to new motions. The recent success of approaches using cycle-GANs on unpaired image-to-image translation [Zhu et al. 2017] inspires research on investigating motion retargeting with unpaired datasets. Villegas et al. [2018] propose using a cycle-consistency adversarial objective with a forward kinematics-based recurrent network for motion retargeting. Aberman et al. [2020] propose a skeleton-aware network to process motion such that motion can be retargeted to another character with a differently structured skeleton. However, the aforementioned data-driven methods only work for retargeting animations between humanoid characters. Noam et al. [Aigerman et al. 2022] discusses retargeting between arbitrary meshes while focusing less on the animation of articulated characters.

Several works [Choi et al. 2020; Kim et al. 2022; Rhodin et al. 2014, 2015; Seol et al. 2013; Yamane et al. 2010] have explored retargeting motions from humans to non-humanoid characters, where the skeleton of the target character may greatly differ from the sources in terms of both structures and dimensions. These methods require selecting a few keyframe poses from human-captured motion sequences and manually pairing them with poses [Rhodin et al. 2014; Yamane et al. 2010] or motion sequences [Rhodin et al. 2015; Seol et al. 2013] for the target character, which can be labor-intensive. Abdul-Massih et al. [2017] addresses the cross-morphology motion retargeting problem by defining *Groups of Body Parts* (GBPs) and translating the problem into constrained optimization to preserve the semantics of the original motion. However, the motions beside GBPs need to be designed manually. In this work, we tackle the problem of cross-morphology motion retargeting with an unpaired dataset. Our goal is to transfer the motion of a huma to a nonhuman character, while preserving the semantics.

2.2 Embedding Space Models for Animation and Control

Embedding space models have been explored in controller design for both kinematic and dynamic characters. The idea is to learn

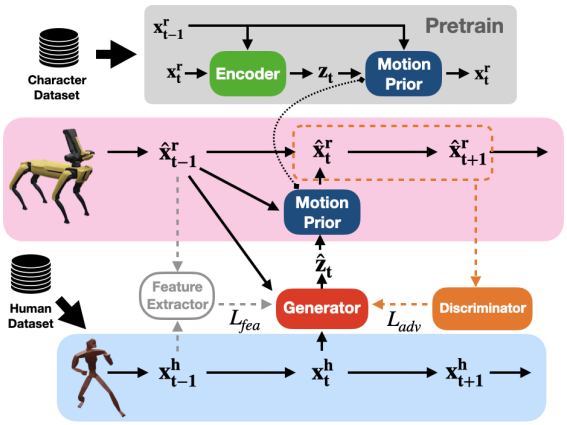


Fig. 2. Overview diagram of Adversarial Correspondence Embedding (ACE). We first pre-train a motion prior that controls the character's state $\mathbf{x}^r$ with a latent variable $\mathbf{z}$. Then we train a motion retargeting Generator that maps the current human state $\mathbf{x}_t^h$ and the previous character state $\mathbf{x}_{t-1}^r$ into the latent variable, and a discriminator that determines whether the state transition is realistic or not. We also introduce an additional feature loss that guides the correspondence learning.

a low-dimensional embedding that encapsulates natural-looking motions, then to use this learned embedding to efficiently construct controllers or further constrain output motions. Given kinematic motions, the models can be trained via supervision with specialized network architectures [Kim et al. 2022; Starke et al. 2022; Zhang et al. 2018] or in an unsupervised manner [Li et al. 2021a; Ling et al. 2020; Rempe et al. 2021]. For physically simulated characters, the embedding models are often trained implicitly while learning imitation controllers via reinforcement learning. For instance, Peng et al. [2019] propose multiplicative compositional policies (MCP), which allow a policy to explore a compact embedding space to activate multiple low-level skills. Won et al. [2021] use similar embedding models to solve multi-agent problems. Recently, generative embedding models that can control simulated characters without conditioning on task-specific inputs have been studied based on conditional VAEs [Won et al. 2022; Yao et al. 2022] and adversarial learning [Peng et al. 2022]. Once the embedding models are trained, a hierarchical controller can be trained to achieve various motions by traversing the embedding. Similar to these works, we also leverage a pre-trained motion prior, but study motion retargeting using the low-dimensional embedding space.

3 OVERVIEW

We present Adversarial Correspondence Embedding (ACE), a framework for retargeting human motions to characters with significantly different morphologies. Our problem takes a human motion dataset Ω^h and a character motion dataset Ω^r as inputs and learns a generator G that translates a human state $\mathbf{x}^h$ (Section 6.1) into a target character state $\mathbf{x}^r$ (Section 6.2). We aim for the generated motions to adapt to the target character's motion patterns while preserving the semantics of the original human motion. However, training such a retargeting function is not straightforward due to the significant morphological differences between the two datasets, as well as the intrinsic ambiguity of the interspecific motion retargeting problem.

We approach this problem using the framework of adversarial learning. We draw inspiration from its recent success in unpaired image-to-image translation [Zhu et al. 2017] to build a correspondence between human and character motions. Unlike supervised learning [Kim et al. 2022], unsupervised learning allows us to learn this correspondence with minimal or zero paired inputs. To further improve the quality of motion, we pre-train a motion prior π for a character and construct a low-dimensional embedding space along with a controller that can generate diverse motions within this space. An overview of our approach is illustrated in Figure 2.

4 PRE-TRAINING OF CHARACTER MOTION PRIOR

Motion retargeting algorithms often directly map the input motion of the source character to a high-dimensional motion of the target character. However, learning such a complex mapping can be less practical due to unstable convergence during training. Instead, we propose training a motion prior, denoted as $\pi(\mathbf{z}_t, \mathbf{x}_{t-1}^r) \mapsto \mathbf{x}_t^r$, which maps the embedded control variable $\mathbf{z}_t$ and the previous character's state $\mathbf{x}_{t-1}^r$ to the character's state at the current step $\mathbf{x}_t^r$. Later, motion retargeting is learned in this embedding space, making cross-morphology motion mapping more stable.

Recent literature [Abdul-Massih et al. 2017; Peng et al. 2022; Zhang et al. 2018] in computer animation has discussed learning-based techniques to obtain such generative motion controllers and may complement the proposed framework. Notably, our framework is agnostic to the choice of the method used to generate the motion prior. In this paper, we use Variational Autoencoder (VAE)-based method [Ling et al. 2020] to learn motion prior π:

$$\mathbf{z}_t = c(\mathbf{x}_{t-1}^r, \mathbf{x}_t^r) \tag{1}$$

$$\arg\min_{\pi} \mathbb{E}_{(\mathbf{x}_{t-1}^r, \mathbf{x}_t^r) \sim \Omega^r} ||\mathbf{x}_t^r - \pi(\mathbf{z}_t, \mathbf{x}_{t-1}^r)|| \tag{2}$$

where $\mathbf{x}_t^r$ is the character's state at time instant t, $\mathbf{z}_t$ is the embedded variable generated by encoding state transition $(\mathbf{x}_{t-1}^r, \mathbf{x}_t^r)$ using the encoder network c. Given the embedded state $\mathbf{z}_t$ and the character's state, the motion prior π aims to reconstruct $\mathbf{x}_t^r$. However, training an effective motion controller that can capture all state transitions while preserving naturalness is a challenging task. To address this, we utilize the Mode-adaptive network (MANN) [Zhang et al. 2018] as the neual network architecture of π, which has been proven effective in previous animation works. For detailed implementation of MANN and the encoder, please refer to Sec 7.1. Through joint training of the encoder c and the motion prior π, we can obtain an embedded variable $\mathbf{z}$ and the motion prior for future usage.

5 ADVERSARIAL CORRESPONDENCE EMBEDDING

In this section, our goal is to develop an effective motion retargeting function. Although the problem of motion retargeting inherently involves ambiguity when dealing with inter-specific morphologies, we aim to achieve two notable properties in the retargeted motion. Firstly, the retargeted motion should look natural on the target character. Secondly, it should preserve the key features of the source motion. To accomplish this dual objective, we approach the problem of retargeting using generative adversarial learning [Aberman et al. 2020; Goodfellow et al. 2020; Villegas et al. 2018]. This involves employing a trained discriminator D to distinguish the motions

retargeted by the generator G from an existing character motion dataset. This encourages the generator to produce motions that are indistinguishable from pre-collected character motions, implying that they are close to the character's natural motion. Additionally, we design a simple feature loss to match the high-level features of the source and target motions. It is important to note that this feature loss plays a critical role in finding meaningful correspondence. Without it, a generator may encounter the issue of "mode-collapse", where it learns to produce only a limited range of motions.

5.1 Problem Formulation

The typical formulation [Aberman et al. 2020] of motion retargeting with Generative Adversarial Networks (GANs) learns a generator G to directly map the motion of the source character $\mathbf{x}^h$ into the motion of the target character $\mathbf{x}^r$. However, such generator network can be difficult to learn due to the high dimensional state spaces of our characters. Instead, we leverage our pre-trained motion prior to learn the correspondence in an embedding space. Our generator (motion retargeting network) $G(\mathbf{x}_t^h, \mathbf{x}_{t-1}^r) \mapsto \hat{\mathbf{z}}_t$ takes the current human pose $\mathbf{x}_t^h$ and the previous character motion $\mathbf{x}_{t-1}^r$ to generate the embedded variable $\hat{\mathbf{z}}_t$. Then, the character state is produced by the pretrained motion prior $\pi(\hat{\mathbf{z}}_t, \mathbf{x}_{t-1}^r) \mapsto \hat{\mathbf{x}}_t^r$. Next a discriminator $D(\mathbf{x}_{t-1}^r, \hat{\mathbf{x}}_t^r) \mapsto [0, 1]$ maps the state transition to the generated dataset (0) or character motion dataset (1).

5.2 Training of Discriminator

We train a discriminator D to distinguish the original state transition $(\mathbf{x}_{t-1}^r, \mathbf{x}_t^r)$ in the character motion dataset Ω^r from generated transition $(\mathbf{x}_{t-1}^r, \hat{\mathbf{x}}_t^r)$ by the generator G, by minimizing a discriminator loss L_D:

$$L_D = -\mathbb{E}_{\Omega^r}[log(D(\mathbf{x}_{t-1}^r, \mathbf{x}_t^r))] - \mathbb{E}_{\Omega^h}[log(1 - D(\mathbf{x}_{t-1}^r, \hat{\mathbf{x}}_t^r))], \quad (3)$$

$$\text{where } \hat{\mathbf{x}}_t^r = \pi(G(\mathbf{x}_t^h, \mathbf{x}_{t-1}^r), \mathbf{x}_{t-1}^r)). \quad (4)$$

However, GAN uses an iterative training formulation for updating the discriminator and the generator, which often causes unstable training dynamics. One reason is non-zero gradients on the manifold of real data samples [Mescheder et al. 2018] due to the approximation error in the discriminator. Thus, we incorporate a gradient penalty regularizer, as used in prior works [Peng et al. 2022] to stabilize the training and improve the quality of the training result. This gradient penalty augments the previous discriminator objective as:

$$\arg\min_D \; L_D + \frac{w^{gp}}{2}\mathbb{E}_{\Omega^r}[||\nabla_\lambda D(\lambda)|_{\lambda=\mathbf{z}}||^2], \quad (5)$$

where we set the weight term w^{gp} to be 0.1 for our experiments.

5.3 Training of Generator

Simultaneously, we train a generator G to produce natural motion transition while preserving semantic features of the source motion. It is trained by minimizing the following objective function:

$$\arg\min_G \; w_{adv}L_{adv} + w_{feat}L_{fea}. \quad (6)$$

Here, L_{adv} is the adversarial loss derived from the discriminator D and encourages G to generate movements that deceive D into classifying them as character motion data. L_{fea} is a feature loss that indicates the preservation of the semantic features from the source motion and is inspired by the concept of the group of body parts (GBP) [Delhaisse et al. 2017]. Specifically, the adversarial loss L_{adv} is calculated by:

$$L_{adv}(G) = -log(D(\mathbf{x}_{t-1}^r, \hat{\mathbf{x}}_t^r)) \quad (7)$$

$$= -log(D(\mathbf{x}_{t-1}^r, \pi(G(\mathbf{x}_t^h, \mathbf{x}_{t-1}^r), \mathbf{x}_{t-1}^r))), \quad (8)$$

and L_{feat} is designed to match selected high level features:

$$L_{fea}(G) = ||\Psi(\mathbf{x}_t^h) - \Psi(\hat{\mathbf{x}}_t^r)|| \quad (9)$$

$$= ||\Psi(\mathbf{x}_t^h) - \Psi(\pi(G(\mathbf{x}_t^h, \mathbf{x}_{t-1}^r), \mathbf{x}_{t-1}^r))||, \quad (10)$$

where Ψ is a manually designed feature function, which includes terms like end-effector positions (details in Section 6.3).

To account for potential differences in the number of end-effectors between the human and target character, we have implemented a mechanism for establishing automatic correspondence between their respective end-effector indices. This is achieved by minimizing the KL-divergence between the character's end-effector position distribution and the human end-effector position distribution:

$$j \mapsto i : \arg\min_i \; KL[\; p(x^{r,j}) \;||\; p(x^{h,i}) \;] \quad (11)$$

here, j is the j-th end-effector of the target character, while i is the i-th end-effector of the human. Besides this auto-mapping, the user can also manually define a mapping, if required.

6 MODEL REPRESENTATION

6.1 Human Representation

We prepare a human motion dataset $\Omega^h = \{\xi_1^h, \xi_2^h, \cdots, \xi_N^h\}$ that serves as an input distribution to adversarial learning. Here $\xi^h = \{\mathbf{x}_1^h, \mathbf{x}_2^h, \cdots, \mathbf{x}_T^h\}$ denotes one human motion trajectory containing states $\mathbf{x}^h$. Each trajectory can have a different length of state sequence depending on the source motion. The state $\mathbf{x}_t^h$ at time t includes features as follows:

- Height of the root from the ground [1 dim].
- Orientation of the root [4 dims].
- Linear and angular velocities of the root [6 dims].
- Position of each joint [51 dims].
- End-effector (foot, hand, head) position [15 dims].
- End-effector (foot, hand, head) velocity [15 dims].

Except for the height of the root, all the features are defined in the *local coordinate* frame of the human character. Specifically, we define the local coordinate frame as follows. First, we select the pelvis of the character as the root node. Then the character's local coordinate frame is defined with its origin on the root node, its x-axis aligned with the direction that the character is facing and its z-axis is aligned with a global up-vector.

6.2 Target Character Representation

Similar as the human motion dataset, we prepare a character motion dataset $\Omega^r = \{\xi_1^r, \xi_2^r, \cdots, \xi_M^r\}$. We define each motion $\xi^r = \{\mathbf{x}_1^r, \mathbf{x}_2^r, \cdots, \mathbf{x}_T^r\}$, where $\mathbf{x}^r$ represents a state vector.

The character's state vector includes the following items:

- Height of the root from the ground [1 dim].

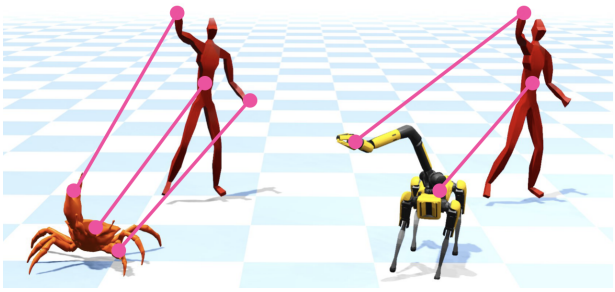


Fig. 3. Illustration of the feature correspondence between the human and the characers. The end-effector correspondence is built in automatically. The feature loss is designed to track the root's and end-effector's motion.

- Orientation of the root [4 dim].
- Relative location of the root from the previous frame [2 dim].
- Orientation of the root in the previous frame [4 dim].
- Linear and angular velocities of the root [6 dim].
- Pose of each joint [joint number dim].
- End-effector positions [(3 * Number of EE) dim].
- End-effector velocities [(3 * Number of EE) dim].

Similar to Section 6.1, the features are defined in the *local frame*, which is centered at the root node. The embedded control variable $\mathbf{z}$ of the character is defined as a latent variable with 32 dims.

6.3 Design of Feature Loss

To preserve the semantic meaning of motions, we add an additional feature loss to the training of the generator. The features selected Ψ for this feature loss are as follows:

- Height of the root.
- Orientation of the root.
- Linear and angular velocities of the root.
- End-effector position.

All terms are normalized according to the character's body length. Figure 3 displays the visualization of the features. The generator is trained to match these features between the source human and target robot motion for all motions. End-effector correspondence is automatically generated using the method introduced in Section 5.3. However, manual mapping is also possible, and we present the results in the result section. Overall, these features can be selected without requiring extensive expert knowledge and have been widely used in previous works such as [Aberman et al. 2020].

6.4 Network Architecture

This works contains 4 networks: the state transition encoder E and motion prior π in the pretraining stage; the generator G and the discriminator D in the training stage. Here we list the their structure:

- Encoder Network c: MLP network with LeakyRelu as activation function and a structure of [512, 512, 512, 512].
- Motion Prior π: MANN [Zhang et al. 2018] structure with 8 experts and 512 as the unit number for each network layers.
- Generator Network G: MLP network with LeakyRelu as activation function and a structure of [512, 512, 512].
- Discriminator Network D: MLP network with SiLU as activation function and a structure of [512, 512, 512].

7 EXPERIMENT AND EVALUATION

In this section, we present qualitative evaluation of our approach on retargeting human motion to different characters. We transfer motion from a human to a Spot robot (a quadruped with a manipulator), a Crab character (a hexapod with two arms), and a Stretch robot (a wheeled robot with a manipulator). The wide range of morphologies that we experiment with further reinforce the generality of our approach, and that it can enable generic motion transfer between inter-specific morphologies. Quantitatively, we compare our approach, ACE, against different baseline methods. In addition, we conduct a user study to further evaluate our method against other approaches. Finally, we transfer the retargeted motion to the real Spot robot.

7.1 Implementation Details and Datasets

Our motion retargeting framework is implemented in PyTorch, and the experiments are performed on a PC equipped with an NVIDIA GeForce RTX 2070 Super and AMD Ryzen 9 3900X 12-Core Processor. We optimize the parameters of the motion prior, discriminator and generator with the loss functions mentioned in Section 5.3 using the Adam optimizer [Kingma and Ba 2014]. Training in total takes about 1 hour without any parallelization, including training the motion prior, generator and discriminator.

To evaluate our method, we constructed a human motion dataset with 200 trajectories that contains 75594 input motion states. The human dataset is from 2 sources: CMU Motion Capture Database [CMU 2002], and Ubisoft La Forge Animation Dataset ("LAFAN1") [Harvey et al. 2020]. The motions in the CMU dataset are around 300 frames (4.8s) while the LAFAN1 Dataset contains large motion trajectories (around 5000 frames, 90s). The selected motions include a variety of human motions including sports, dancing, housework and construction work.

The evaluation of retargeting human motions are conducted on three characters with various morphologies:

- **Spot**: Spot is a quadrupedal robot [BostonDynamics 2019] with a manipulator on its back, developed by Boston Dynamics. The robot has 32 cm long thigh and shank links. It has 12 degrees of freedom (DoFs) for locomotion and 6 DoFs for its manipulator, which results in a total of 18 DoFs.
- **Crab**: Crab is a hexapod character with two arms. With each leg and arm has 3 joints which leads to a 24 DoFs.
- **Stretch**: Stretch is a wheel-based mobile manipulation robot character from Hello Robot [HelloRobot 2023]. The Stretch has a single arm with four prismatic joints.

One crucial component of ACE is the learned motion prior that should be expressive and be able execute a large range of skills. This requires a diverse motion dataset on the target characters. Our character dataset contains different locomotion gaits with base linear and angular velocities ranging from -1.5 to 5m/s and -1 to 1 rad/s. The character motion are generated by rolling out kinematic controller with random target commands. The kinematics controller plans footstep location based on the current and target velocity of the root and uses a parabolic trajectories generator for swing leg motion. We tuned the parameters of the controller to make

the generated motions similar to the dog mocap data. For wheel-based robots, we make simplification by assuming it can walk with arbitrary speeds and directions using low-level controllers. We use a total 200k data points for each character.

7.2 Main Results

We illustrate various retargeted motions of all the characters in Figure 4. Our method successfully retargeted a wide range of dynamic human motions, such as sports and construction activities. In all scenarios, the generated motions look feasible and preserve the original semantics of the human motions without showing significant visual artifacts, such as self-penetration or foot skating. Note that our method is general enough to support very different characters with various numbers of legs and arms.

Our system considers the different capabilities of the human and the character by matching the *normalized* feature vectors. Therefore, the retargeted motions often travel or sweep less than the original human motions. For example, Spot pushes the object for 1.39 m, which is less than 3.48 m of the human (Figure 4a, the third row). This design choice is reasonable considering the height difference between the human ($\approx$ 1.7 m) and the Spot robot ($\approx$ 0.51 m). This scaling factor can also be easily changed based on user preference.

In our design, the characters take their footsteps in their own styles, instead of taking synchronized steps. This allows the characters to exhibit more feasible and natural motions based on their capability: for instance, the *drag* motion of Spot naturally changes the gait from trotting to galloping based on the human's walking speed (*1:56* in the supplemental video). However, we also want to note that this design decision may sacrifice the additional semantics of feet movements.

7.3 Baseline Comparison

To prove the effectiveness of the proposed method and its individual component, we compare our method to the following methods:

- **Neural Kinematic Network (NKN)**: the first baseline we compare is Neural Kinematic Network (NKN) of Villegas et al. [2018]. NKN uses a recurrent neural network structure with a Forward Kinematics layer and adversarial learning with cycle consistency. However, NKN only addresses the morphology with same skeleton structure which can not be adapt to our setting as-is. Since the number of joints is different between the domains, we use the same treatment as proposed by Aberman et al. [2020] which removes NKN's reconstruction loss.
- **ACE without Feature Loss (ACEwoFea)**: the second baseline is ACE but without the feature loss. This baseline aim to evaluate how feature loss affects the result of the training.
- **ACE without Adversarial Loss (ACEwoAdv)**: the third baseline is ACE without the adversarial loss, which corresponds to inverse kinematics only based on manual features. Here, we aim to evaluate the importance of the adversarial loss in training motion retargeting function.

For quantitatively measuring the performance of the approaches, we borrow two evaluation metrics, Diversity and Frechet Inception Distance, from the existing text-to-motion literature [Guo et al. 2022, 2020; Tevet et al. 2022]. We further adopt two additional metrics, a feature loss and the unrealistic frame ratio.

- **Diversity (DIV)**: Diversity measures the variance of generated motion across all source human motions. From a set of all generated motions from different source human motions, two subsets of the same size S_d are randomly picked. Their motion features $\{\Psi(\mathbf{x}^r_1) \cdots \Psi(\mathbf{x}^r_{S_d})\}$ and $\{\Psi(\mathbf{x'}^r_1) \cdots \Psi(\mathbf{x'}^r_{S_d})\}$ are extracted as defined in Sec. 6.3. The diversity of this set of motion is defined as: $DIV = \frac{1}{S_d}\Sigma_{i=1}^{S_d}||\Psi(\mathbf{x}^r_i) - \Psi^r(\mathbf{x'}^r_i)||$. In motion retargeting, it is better to obtain a DIV score similar to that of the dataset. In addition, lower diversity often indicates the degeneration of the synthesized motions.
- **Frechet Inception Distance (FID)**: FID measures the distance between feature vectors computed for two motion datasets, which is a common metric to evaluate the synthesized motion quality. We compute the feature distributions for the character's motions and the retargeted motions, and measure the distribution difference. Lower FID indicates that two motion sets have similar feature distributions.
- **Feature Loss (L_{fea})**:We also measure the feature loss (Equation 9). Feature loss indicates the preservation of 'semantic meaning'. Lower scores indicates better results.
- **Unrealistic Frame Ratio (UFR)**: UFR reflects the realism of the motion. It is defined as the number of generated motion frames containing unrealistic effects divided by the total number of motion frames. The unrealistic effects that we consider include self-collision, foot penetration, and foot sliding.

We evaluate our motion retargeting framework on 16 motions including sports activities, construction tasks, and house chores. The quantitative results are summarized in Table 1. The qualitative results are presented in Figure 4 and can be best seen in the supplementary video.

Table 1. Quantitative results on the Spot.

	DIV $\rightarrow$	FID $\downarrow$	$L_{fea}\downarrow$	UFR $\downarrow$
Dataset	2.254	0.000	N/A	0.258%
ACE(Ours)	**2.483**	**0.489**	0.606	2.071%
NKN	1.718	0.914	0.912	6.213%
ACEwoFea	0.445	0.976	1.975	**0.517**%
ACEwoAdv	3.077	0.736	**0.553**	9.741%

Overall, **ACE** generates natural and compelling retargeted motions on the every chracters, including both realistic leg and arm movements (Figure 4). The generated motions maintain the semantic knowledge in the original motion, and show minimal transfer artifacts, such as foot skating or self-collision, thanks to the pre-trained motion priors and adversarial loss. The robot also shows rich whole-body movements as shown in Volleyball, Tennis and Pushing (Figure 4a). As a result, **ACE** outperforms all the methods in DIV (closest to DIV of the dataset) and FID, while being the second best in L_{fea} and UFR.

Although **ACEwoFea** produces natural motions, the generated motions fall into the repetitive patterns and lose the semantic information of the human motion, as demonstrated by the Diversity

value of 0.445 and L_{fea} value of 1.975 while **ACE** gets a higher diversity and a lower feature loss. This mode collapse phenomenon is very common in adversarial learning settings. The features are used as a regularization term in training to help mitigate mode collapse.

ACEwoAdv is capable of generating reasonable outputs when retargeting certain human motions by preserving the semantic features: it shows the lowest L_{fea} of 0.553. However, there are many scenarios where it fails to produce satisfactory results, despite the use of a pretrained motion prior. This can be demonstrated by its high FID value of 0.736 while **ACE** has 0.489. In addition, it often produces unrealistic motions with self-collision, foot penetration, and foot sliding, which is supported by the worst unrealistic frame ratio (UFR) value of 9.741%.

Our comparison reveals that when generating high-quality retargeted motions, **ACE** outperforms **NKN** in many criteria. Previous work on motion retargeting [Aberman et al. 2020] has noted the crucial role played by the reconstruction loss in **NKN**. Removing this component for supporting cross-morphology scenarios could result in degradation of the motion quality produced by **NKN**.

7.4 User Study

In addition to the aforementioned quantitative evaluation, we conducted a user study to assess how our method performs in terms of visual perceptual quality when compared to other baseline methods. Our study group comprised 20 participants with varying levels of expertise and experience in character animation. Prior to the study, participants were provided a comprehensive explanation of the study, without any information about the underlying method.

In the user study, five different source human motions were randomly selected from the dataset and presented to the subjects, along with the retargeted character motions generated by each method. The selected human motions comprised various sports and other human activities. The subjects are asked to evaluate the generated motion in 0 to 5 scales based on realism and the magnitude alignment to the source human motion. During the study, the user has no access to the method that generates the motion.

The preference results are presented in Table 2. According to the results, most users picked the retargeted motions of **ACE** as the most favorable. On the other hand, **ACEwoFea** received the worst result as it loses all semantic information of the human motions. Our findings also show that **ACEwoAdv** achieved a relatively high score, although **ACE** is still statistically better than **ACEwoAdv** with a p-value of 0.03. This is because **ACEwoAdv** can provide better results compared to **NKN** and **ACEwoFea**. However, it is important to note that unrealistic effects such as foot sliding or foot penetration may not be easily captured by non-expert users. The user study offers strong evidence of the effectiveness of our approach in retargeting motions across different morphologies.

Table 2. User study on scoring the retargeted motions.

ACE(Ours)	NKN	ACEwoFea	ACEwoAdv
4.25 ± 0.39	1.41 ± 0.78 (***)	1.01 ± 0.80 (***)	3.95 ± 0.45 (*)

7.5 Flexibility for Incorporating User's Preference

Although ACE includes an automatic end-effector mapping mechanism, the end-effector correspondence can also be manually set according to the user's preference. In the previous section, the automatic mapping assigned the manipulator of the Spot robot to the right hand of the human character. However, we can manually assign the manipulator to the left hand of the human and use ACE to produce new retargeted motions. Figure 5 presents the result under the new mapping. In addition to manual end-effector mapping, other heuristics, such as specific joint-level mapping or foot-pattern synchronization, can be incorporated into ACE by modifying the feature loss function.

7.6 Real Robot Experiments

One application of our work is to reproduce the retargeted motions on a real robot via sim-to-real transfer. This is important in robotics because it allows the robot to acquire various motor skills from human movements. Once motion is retargeted, we can use several techniques for executing the given motion, such as motion imitation [Peng et al. 2018] or model-based control [Li et al. 2021b].

We selected the Spot robot as the robotic platform. The vendor-provided Spot API takes as input the base velocity command and the manipulator joint angle. Just for this experiment, we manually define the latent space z as the root and arm commands to directly leverage the vendor-provided controller as the low-level motion prior.

We transferred two motions, *Sword* and *Fencing*, which involve rich full-body motions and rapid arm motions. The Spot robot was able to execute both sequences at 100 % success rates out of five trials. Our ACE framework generates physically plausible motions that are within the region of attraction of the given controller. However, the details of the footsteps were different, particularly when the arm is stretched and starts to affect the balance. In this case, the real Spot robot takes wider steps to recover the balance (Figure 7). The numbers of footsteps are also different: 14 for our kinematically retargeted motions and 26 for the real Spot due to the differences in the controllers as well as the need for balance recovery. However, it is important to note that these experiments are designed to highlight the robotic application but do not indicate any guarantee on sim-to-real transfer.

8 DISCUSSION AND FUTURE WORK

This work presents a learning-based framework, Adversarial Correspondence Embedding (ACE), which retargets a given human motion to another character with significant morphological differences. Our framework leverages adversarial learning to generate natural character motions while guiding the correspondence learning via a feature loss. We also introduce a pre-trained motion prior and learn retargeting in a compact embedding space, which leads to smooth, physically realistic motion on the character. We demonstrate that the proposed framework can generate compelling retargeted motions on three characters, Spot, Crab, and Stretch, with various morphologies. We also conduct baseline comparisons and a user study to justify the design decisions of our framework. Finally,

we highlight the potential robotic application by transferring the retargeted motions to a real Spot robot.

In our experiment, we observed some poorly retargeted results. They are mainly caused by two reasons. The first reason is that our database does not have enough motions to support certain types of human motions. For instance, a fast backward human motion is transferred to a much slower motion of a character due to the lack of corresponding motion. The second reason is the quality of the learned motion priors. Although we use the Mode-Adaptive Network (MANN) to replicate motions in the dataset, it occasionally produces jerky or unexpected motions. To improve the quality of the resulting motions, we can either add more data to the dataset or use more advanced motion synthesis techniques, such as Deep Phase [Starke et al. 2022].

There are several interesting directions that we aim to explore as our future work. Our current implementation does not systematically consider the differences in dynamic capabilities between morphologies. For example, a larger human may be able to turn faster than a smaller character. Exploring solutions for such scenarios may require investigating long-horizon motion planning with reinforcement learning or time-warping. In addition, our trained model is limited to a single robot. We plan to investigate a general motion embedding space that can freely translate motions back and forth betweenvarious morphologies, including human to character, character to human, and character to character. Finally, our framework may be less effective in handling characters without clear notions of arms and legs, such as a shark [Seol et al. 2013] or a caterpillar [Rhodin et al. 2014]. Extending the proposed framework for more general characters will be an interesting future research direction.

ACKNOWLEDGMENTS

This material is based upon work supported by Meta Inc and the National Science Foundation under Grant No. 2222723. Jungdam Won was partially supported by the New Faculty Startup Fund from Seoul National University, ICT(Institute of Computer Technology) at Seoul National University. We would like to thank Jiahan Fan for helping with visualization and Morgan Byrd for helping with video.

REFERENCES

Michel Abdul-Massih, Innfarn Yoo, and Bedrich Benes. 2017. Motion style retargeting to characters with different morphologies. In *Computer Graphics Forum*, Vol. 36. Wiley Online Library, 86–99.

Kfir Aberman, Peizhuo Li, Dani Lischinski, Olga Sorkine-Hornung, Daniel Cohen-Or, and Baoquan Chen. 2020. Skeleton-aware networks for deep motion retargeting. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 62–1.

Noam Aigerman, Kunal Gupta, Vladimir G Kim, Siddhartha Chaudhuri, Jun Saito, and Thibault Groueix. 2022. Neural jacobian fields: Learning intrinsic mappings of arbitrary meshes. *arXiv preprint arXiv:2205.02904* (2022).

Blizzard. 2016. Overwatch. https://overwatch.blizzard.com/

BostonDynamics. 2019. Spot® - The Agile Mobile Robot. https://www.bostondynamics.com/products/spot

Kwang-Jin Choi and Hyeong-Seok Ko. 2000. Online motion retargetting. *The Journal of Visualization and Computer Animation* 11, 5 (2000), 223–235.

Sungjoon Choi, Matt Pan, and Joohyung Kim. 2020. Nonparametric motion retargeting for humanoid robots on shared latent space. *Proceedings of Robotics: Science and Systems (R: SS)* (2020).

CMU. 2002. CMU Graphics Lab Motion Capture Database. https://mocap.cs.cmu.edu

Brian Delhaisse, Domingo Esteban, Leonel Rozo, and Darwin Caldwell. 2017. Transfer learning of shared latent spaces between robots with similar kinematic structure. In *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 4142–4149.

Disney. 1989. The Little Mermaid. https://movies.disney.com/the-little-mermaid

Disney. 1991. Beauty and the Beast. https://movies.disney.com/beauty-and-the-beast

Michael Gleicher. 1998. Retargetting motion to new characters. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. 33–42.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

Ruben Grandia, Farbod Farshidian, Espen Knoop, Christian Schumacher, Marco Hutter, and Moritz Bächer. 2023. DOC: Differentiable Optimal Control for Retargeting Motions onto Legged Robots. *ACM Transactions on Graphics* 42, 4 (2023).

Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. 2022. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5152–5161.

Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. 2020. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*. 2021–2029.

Félix G. Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. 2020. Robust Motion In-Betweening. 39, 4 (2020).

HelloRobot. 2023. Stretch Robot. https://hello-robot.com/product

Hanyoung Jang, Byungjun Kwon, Moonwon Yu, Seong Uk Kim, and Jongmin Kim. 2018. A variational U-Net for motion retargeting. In *SIGGRAPH Asia 2018 Posters*. 1–2.

Sunwoo Kim, Maks Sorokin, Jehee Lee, and Sehoon Ha. 2022. Human Motion Control of Quadrupedal Robots using Deep Reinforcement Learning. *Robotics: Science and Systems* (2022).

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

Tianyu Li, Roberto Calandra, Deepak Pathak, Yuandong Tian, Franziska Meier, and Akshara Rai. 2021a. Planning in learned latent action spaces for generalizable legged locomotion. *IEEE Robotics and Automation Letters* 6, 2 (2021), 2682–2689.

Tianyu Li, Jungdam Won, Sehoon Ha, and Akshara Rai. 2021b. FastMimic: Model-based Motion Imitation for Agile, Diverse and Generalizable Quadrupedal Locomotion. *arXiv preprint arXiv:2109.13362* (2021).

Hung Yu Ling, Fabio Zinno, George Cheng, and Michiel Van De Panne. 2020. Character controllers using motion vaes. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 40–1.

Lucasfilm. 1977. Starwars. https://www.starwars.com/

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. 2018. Which training methods for GANs do actually converge?. In *International conference on machine learning*. PMLR, 3481–3490.

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

Xue Bin Peng, Michael Chang, Grace Zhang, Pieter Abbeel, and Sergey Levine. 2019. Mcp: Learning composable hierarchical control with multiplicative compositional policies. *Advances in Neural Information Processing Systems* 32 (2019).

Xue Bin Peng, Yunrong Guo, Lina Halper, Sergey Levine, and Sanja Fidler. 2022. ASE: Large-Scale Reusable Adversarial Skill Embeddings for Physically Simulated Characters. *arXiv preprint arXiv:2205.01906* (2022).

Pixar. 2008. Wall-E. https://www.pixar.com/feature-films/walle

Ahmad Rashid, Alan Do-Omri, Md Haidar, Qun Liu, Mehdi Rezagholizadeh, et al. 2019. Bilingual-gan: A step towards parallel text generation. *arXiv preprint arXiv:1904.04742* (2019).

Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J Guibas. 2021. Humor: 3d human motion model for robust pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 11488–11499.

Helge Rhodin, James Tompkin, Kwang In Kim, Kiran Varanasi, Hans-Peter Seidel, and Christian Theobalt. 2014. Interactive motion mapping for real-time character control. In *Computer Graphics Forum*, Vol. 33. 273–282.

Helge Rhodin, James Tompkin, Kwang In Kim, Edilson De Aguiar, Hanspeter Pfister, Hans-Peter Seidel, and Christian Theobalt. 2015. Generalizing wave gestures from sparse examples for real-time character control. *ACM Transactions on Graphics (TOG)* 34, 6 (2015), 1–12.

Yeongho Seol, Carol O'Sullivan, and Jehee Lee. 2013. Creature features: online motion puppetry for non-human characters. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 213–221.

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.

Seyoon Tak and Hyeong-Seok Ko. 2005. A physically-based motion retargeting filter. *ACM Transactions on Graphics (TOG)* 24, 1 (2005), 98–117.

Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. 2022. Human motion diffusion model. *arXiv preprint arXiv:2209.14916* (2022).

Ruben Villegas, Jimei Yang, Duygu Ceylan, and Honglak Lee. 2018. Neural kinematic networks for unsupervised motion retargetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 8639–8648.

Jungdam Won, Deepak Gopinath, and Jessica Hodgins. 2021. Control strategies for physically simulated characters performing two-player competitive sports. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–11.

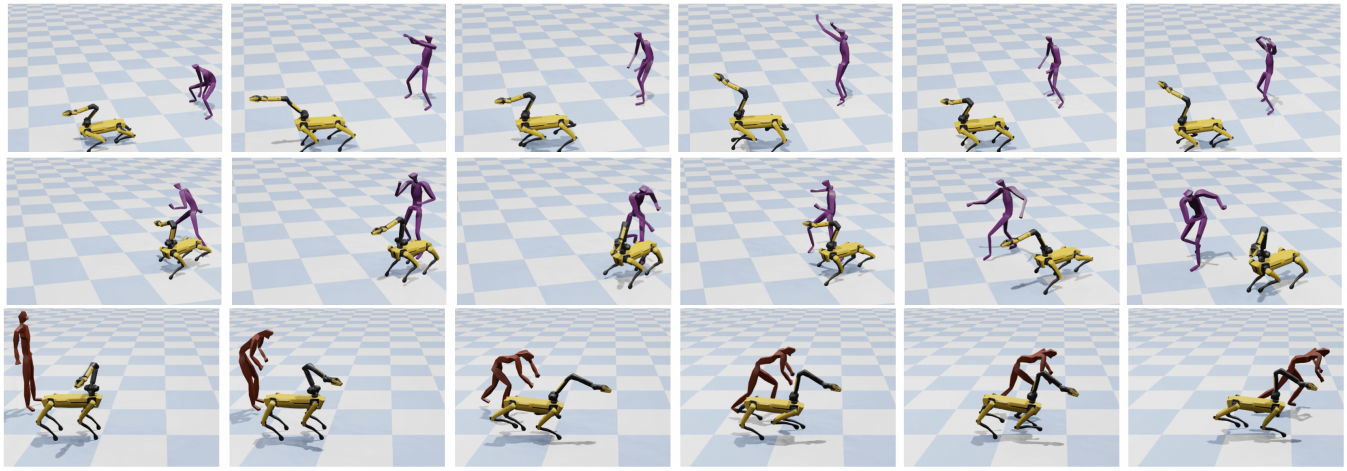
Jungdam Won, Deepak Gopinath, and Jessica Hodgins. 2022. Physics-based character controllers using conditional VAEs. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–12.

Katsu Yamane, Yuka Ariki, and Jessica Hodgins. 2010. Animating non-humanoid characters with human motion data. In *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 169–178.

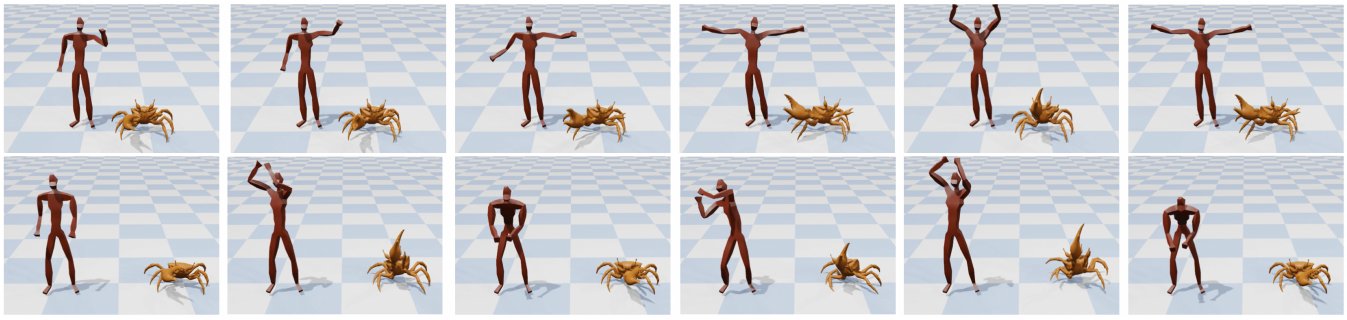
Heyuan Yao, Zhenhua Song, Baoquan Chen, and Libin Liu. 2022. ControlVAE: Model-Based Learning of Generative Controllers for Physics-Based Characters. *ACM Transactions on Graphics (TOG)* 41, 6 (2022), 1–16.

He Zhang, Sebastian Starke, Taku Komura, and Jun Saito. 2018. Mode-adaptive neural networks for quadruped motion control. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–11.

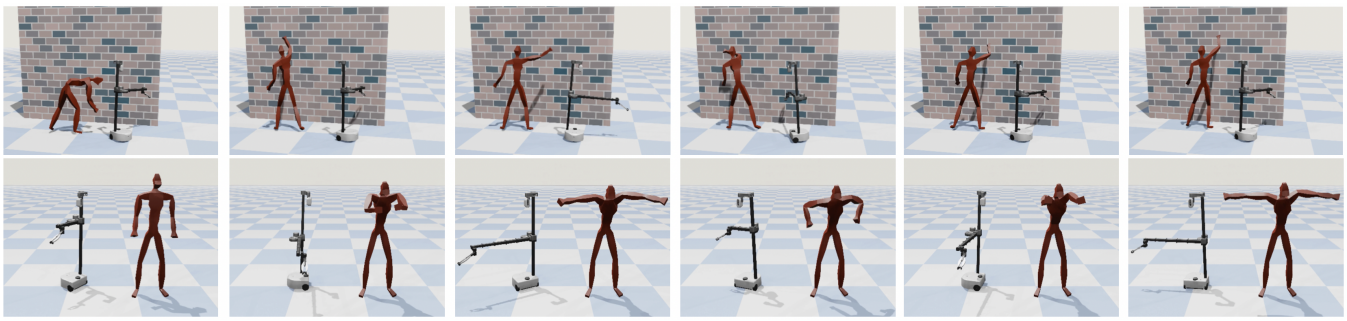
Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.



(a) Spot *Volleyball*, *Tennis*, and *Pushing*.



(b) Crab *Waving* and *Chopping*.



(c) Stretch *Wall Washing* and *Swimming*.

Fig. 4. Retargeted Motions on Spot, Crab and Stretch. Our framework can retarget various human motions while preserving semantic features.

Fig. 5. Different generated Spot *Wave* motions by varying end-effector mapping. The blue uses auto-mapping while the pink one is manually assigning Spot's manipulator with human left arm.

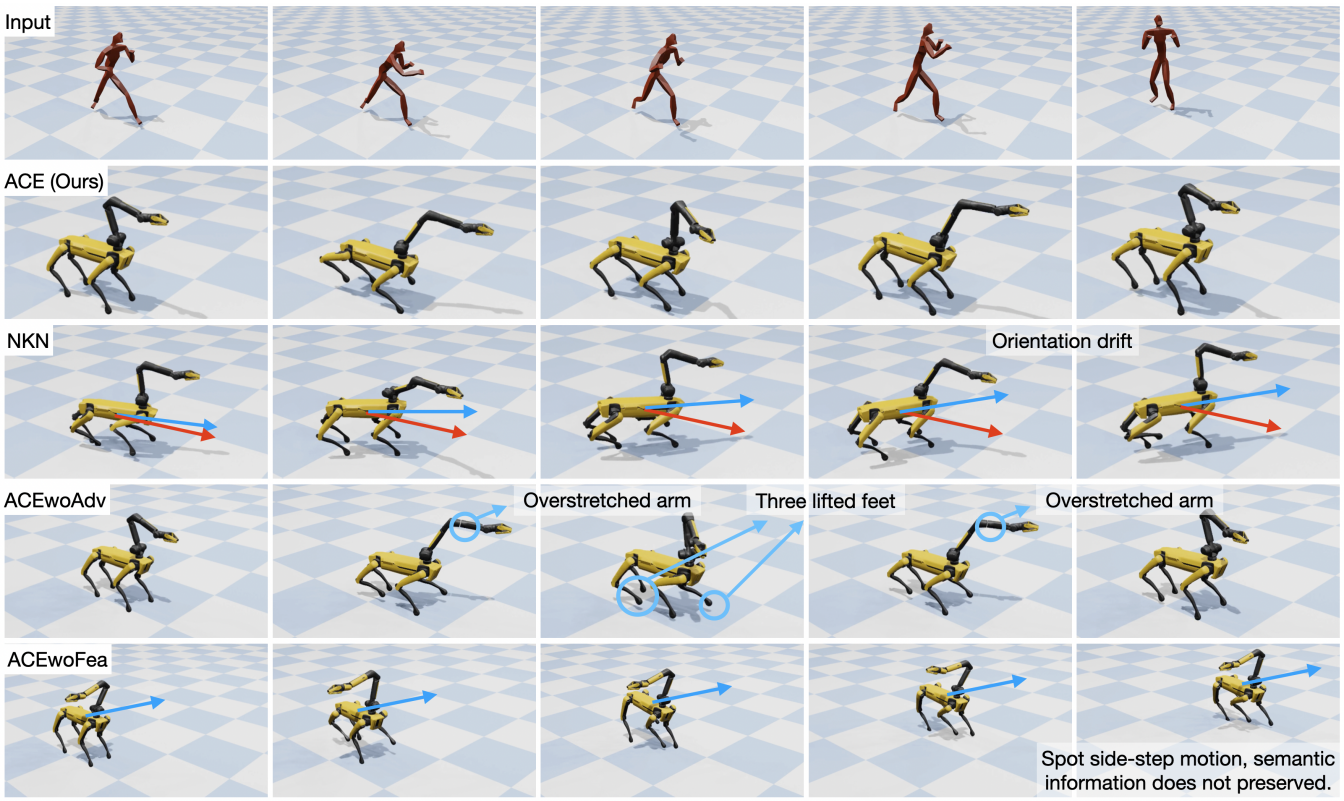


Fig. 6. Comparing our method with the baseline methods, we demonstrate that our approach can generate realistic character motion while preserving the semantic information of the input motion.

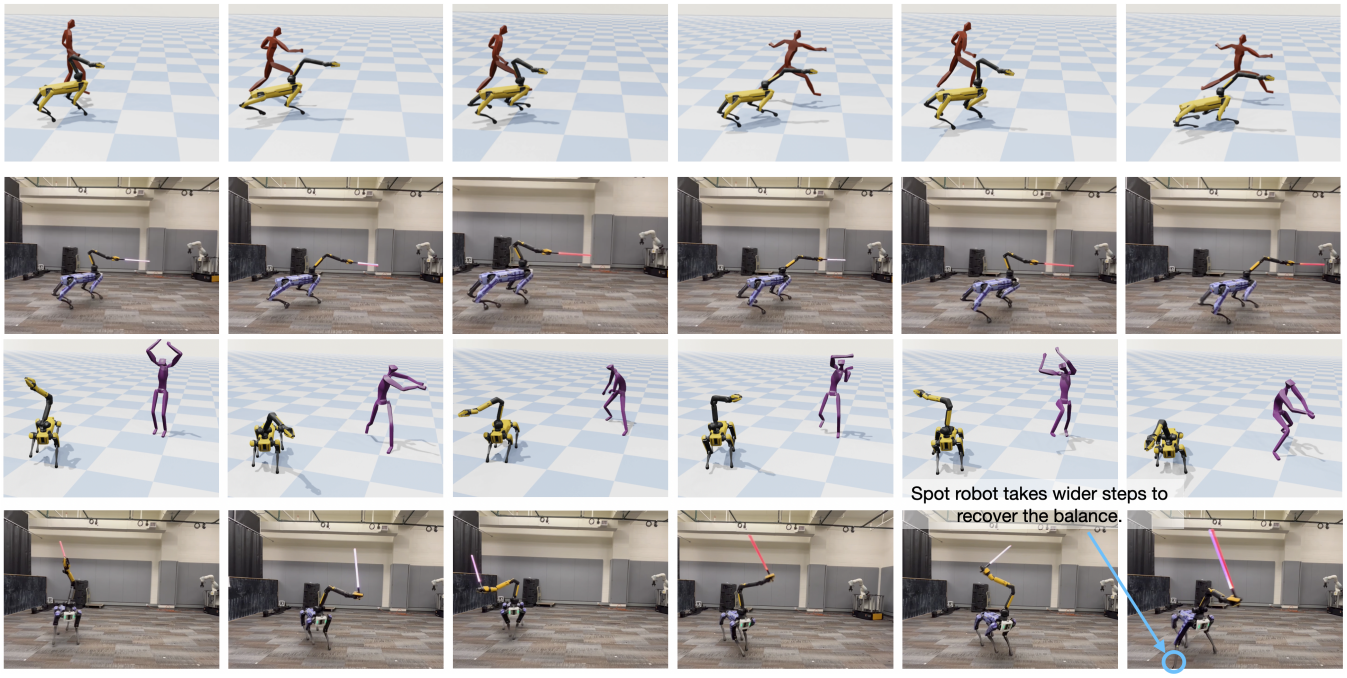


Fig. 7. We successfully transferred two whole-body motions generated for Spot, *Fencing* and *Sword*, to the Spot robot without any failures.