Geometry-Aware Retargeting for Two Skinned Characters Interaction

INSEO JANG, KAIST, Visual Media Lab and KAI Inc., Republic of Korea
SOOJIN CHOI, KAIST, Visual Media Lab, Republic of Korea
SEOKHYEON HONG, KAIST, Visual Media Lab, Republic of Korea
CHAELIN KIM, KAIST, Visual Media Lab, Republic of Korea
JUNYONG NOH*, KAIST, Visual Media Lab, Republic of Korea

Fig. 1. Our method can retarget the interaction motions to multiple characters with various shapes, preserving the semantics in source motions.

Interactive motion between multiple characters is widely utilized in games and movies. However, the method for generating interactive motions considering the character's diverse mesh shape has yet to be studied. We propose a Spatio Cooperative Transformer (SCT) to retarget the interacting motions of two characters having arbitrary mesh connectivity. SCT predicts the residual of root position and joint rotations considering the shape difference between the source and target of interacting characters. In addition, we introduce an anchor loss function for SCT to maintain the geometric distance between the interacting characters when they are retargeted. We also propose a motion augmentation method with deformation-based adaptation to prepare a source-target paired dataset with an identical mesh connectivity for training. In experiments, our method achieved higher accuracy for semantic preservation and produced less artifacts of inter-penetration between the interacting characters for unseen characters and motions than the baselines. Moreover, we conducted a user evaluation using characters with various shapes, spanning low-to-high interaction levels to prove better semantic preservation of our method compared to previous studies.

CCS Concepts: • **Computing methodologies** → **Animation**.

Additional Key Words and Phrases: Character Animation, Motion Retargeting, Two Character Interaction

ACM Reference Format:
Inseo Jang, Soojin Choi, Seokhyeon Hong, Chaelin Kim, and Junyong Noh. 2024. Geometry-Aware Retargeting for Two Skinned Characters Interaction. *ACM Trans. Graph.* 43, 6, Article 203 (December 2024), 17 pages. https://doi.org/10.1145/3687962

0730-0301/2024/12-ART203 $15.00
https://doi.org/10.1145/3687962

1 INTRODUCTION

Motion retargeting is the task of transferring motion from a source character to target characters while maintaining its original semantic context. Because motion capture requires expensive capture devices and laborious post-processing, techniques for motion retargeting have been developed to promote the reusability of motion capture data. Recently, motion retargeting has been utilized in diverse applications, such as films and games. For instance, motion data captured from a performer in a movie can be used for a virtual character in a video game via motion retargeting, reducing both costs and production time.

While techniques for retargeting of one character motion to another have been well-developed, relatively few studies addressed *retargeting interaction motion* between characters because of significant challenges it poses caused by diverse body proportions and shapes that the interacting characters can have, ranging from short to tall and skinny to bulky. Furthermore, interaction motions can span a wide spectrum, from simple interactions that contain a few contacts between characters, such as handshaking and greeting, to complex interactions that include rich contacts, such as dancing, wrestling, and hugging. When contacts are expected, naively applying the same motion parameters, such as joint rotations, to target characters is likely to result in the penetration of a body part of one character to another. One example would be retargeting the motion of a skinny source character to a big-sized target character. In this case, a simple retargeting would make the target character collide with an interacting character due to its bulky belly. To avoid such an artifact, the motion of the target character should be properly adjusted considering the shape of interacting characters.

There are previous approaches that aim to retarget interaction motions between characters by focusing on representations of interactions, such as interaction meshes [Ho et al. 2010] and interaction graphs [Zhang et al. 2023a], which can deal with changes in body proportions but not geometric changes. Furthermore, Jin et al. [2018]

proposed a mesh-level retargeting method that can handle characters with different body shapes. While this method can preserve the geometric semantics of the original motions after the retargeting, it assumes that the source and target characters have an identical mesh connectivity, limiting the generalization to stylized characters with various shapes.

To retarget the interaction motion of two characters having different mesh connectivities, we propose a **Spatio Cooperative Transformer (SCT)** that can retarget source motions to target characters by considering the skeleton structure and mesh information of both the source and target characters, as shown in Figure 1. In addition, we introduce a mesh-agnostic representation, **anchor**, that describes the posed shape of a skinned-character. Furthermore, because interaction data of two characters is difficult to obtain, while a small amount of data leads to overfitting of the network, we augment the training dataset via a **deformation-based motion adaptation** approach to improve the retargeting performance and avoid overfitting from the limited training data.

SCT transfers the motions of a source character to a target character by estimating the root position and joint rotations of the target motion from a source motion. Specifically, the network takes the bone length and the size of the bounding box of the mesh for each joint as input to consider the geometric differences between the source and target characters. SCT learns motion residuals according to the difference of character shape from characters with a uniform mesh connectivity. At inference, SCT generalizes the retargeting scheme to characters with different mesh connectivities.

Retargeting methods based on unsupervised learning [Ruben et al. 2021; Zhang et al. 2023b] often produce jitter or discontinuity of the motion resulting from the lack of ability to estimate the motion from the shape difference between the source and target characters due to the absence of ground truth data. Therefore, SCT is trained in a supervised manner using the dataset obtained in the data preparation step. This approach ensures comprehensive learning of the motion characteristics and facilitates effective retargeting of interaction motions of characters with an arbitrary mesh connectivity.

Anchor is a mesh-independent geometric representation of skinned characters, in the form of a sparse set of points defined on the character's surface. As the character moves its joints, the location of the corresponding geometry can be expressed by anchors assigned to each joint. Therefore, the distance between two anchors from two different joints represents the geometric distance between those two joints. By matching the anchor distance between the interacting source characters to that between the target characters, geometric semantics, such as contact, can be properly maintained. By incorporating an anchor distance loss that minimizes the difference between the source and target anchor distance maps, the network can effectively preserve the semantic context of motions when retargeting of interaction motion is performed between characters with various body shapes.

Finally, in a data preparation step, we utilize a novel motion augmentation method based on deformation-based adaptation. We adjust the motion of target characters having the identical mesh connectivity using a procedural method via a relationship descriptor [Al-Asqhar et al. 2013]. After enriching character data by scaling the skeleton and varying the shape of its mesh, we augment the motion dataset by adjusting the relative distance between interacting characters based on the deformation. By obtaining diverse data for the characters and their interaction motions, our network handles the interaction motions of the characters with various shapes robustly.

We conducted experiments using diverse interaction motions between characters ranging from simple (e.g. greeting, hand-shaking, push-ups) to complex ones (e.g. salsa, hugging). Through qualitative and quantitative evaluations, we confirmed that our method outperforms previous retargeting methods for both meshes with uniform and different connectivities. Specifically, our method excels in preserving the semantic context of the original motion, while avoiding penetration artifacts. Additionally, we conducted a user study to validate the perceptual naturalness of the generated results.

Our contributions can be summarized as follows:

- We propose a SCT designed to retarget interaction motions from the source characters to target characters with different mesh connectivity in a supervised manner using a paired dataset of different characters with various body shapes.
- We introduce the concept of anchor and an anchor distance loss to effectively transfer interaction semantics across characters with diverse shapes.
- We suggest a new data augmentation method to adjust motion by maintaining a semantic context from geometric changes of the character, which is used to collect a paired dataset to train SCT with characters having diverse shapes.

2 RELATED WORK

2.1 Single Character Retargeting

Skeleton-based Retargeting. Early studies on motion retargeting relied on optimization-based approaches. Gleicher [1998] is the pioneer in exploring the methods to adapt the motion of an articulated character to another with constrained space-time optimization. The extension of this approach additionally utilized inverse rate control for online retargeting [Choi and Ko 1999]. Other methods addressed the retargeting for physically plausible motion using Kalman filter [Tak and Ko 2005] or physical controllers based on a linear quadratic regulator [Al Borno et al. 2018].

Recently, data-driven methods have been in the spotlight with the advent of deep-learning. When performing a motion retargeting task, because it is difficult to obtain paired data for all possible motion and character models, an unsupervised learning scheme has been employed to bypass the difficulty of preparing the paired dataset to train a network, using a cycle consistency loss [Villegas et al. 2018], a shared latent space across different characters [Aberman et al. 2020], a pose loss for the same latent feature between input and output poses [Lim et al. 2019], and an adversarial loss [Hu et al. 2023]. Recently, we begin to see approaches that prepare a large-scale paired dataset with the same semantics to train the network in a supervised manner [Kim et al. 2020; Lee et al. 2023]. Unlike these approaches, our method focuses on retargeting the interaction motion between two characters in a supervised manner, considering the geometry of the characters to reduce the penetration and contact-missing artifact.

Geometry-based Retargeting. Transferring motions from one character to another with different geometry has been studied to find a way to maintain visual naturalness after the motion retargeting. Some researchers have used optimization to prevent self-penetration and preserve the contact observed in the original motion [Basset et al. 2019; Ho and Shum 2013; Lyard and Magnenat-Thalmann 2008]. Liu et al. [2018] used a deformation function to transfer the relationship between the body surfaces of two characters with local interaction modeled by a context graph. Extending this idea, our method aims to handle the retargeting of interacting characters with both uniform and different mesh connectivities.

Recent deep learning-based methods have overcome the constraint of requiring the source and target characters to share the uniform mesh connectivity [Cheynel et al. 2023; Ruben et al. 2021; Zhang et al. 2023b]. Because it is difficult to obtain the ground truth data for retargeting, Ruben et al. [2021] utilize a test-time optimization scheme to preserve contacts and Zhang et al. [2023b] use rotation modification when inter-penetration or contact-missing artifacts are detected. Because these approaches focus on a single-character scenario, it is not clear how to determine the motion of a character in relation to the interacting character with these approaches. Our method is designed to handle this case.

2.2 Multiple Character Interaction

Retargeting for Multiple Characters. Ho et al. [2010] suggested a method that can preserve the semantics of motions when retargeting the motions to other deformable articulated characters by minimizing the deformation of the interaction mesh. Zhang et al. [2023a] extended this idea to construct an interaction graph for preserving the skeletal distance between interacting characters in a physical simulator. Jin et al. [2018] suggested the use of a volumetric mesh surrounding the character's skin to preserve the proximity and contact observed in the source motion when generating the corresponding target motion. Unlike these previous studies that do not focus on dealing with meshes with various connectivities, we propose an SCT network that can retarget interaction motion between the characters in a mesh-agnostic manner.

Synthesizing Interaction Motions. Approaches to synthesizing the interaction motions of two characters have been consistently proposed. Liu et al. [2006] modeled the complex interaction motion from the motion sequence of a single character using space-time optimization constrained by a user-specified interactive constraint. Recently, learning-based approaches to synthesizing an interaction motion between two characters have been suggested based on the diffusion models that can generate motion sequences from text [Ghosh et al. 2023; Liang et al. 2024; Shafir et al. 2023]. Goel et al. [2022] proposed a method to synthesize the interacting character's reactive motions from a given motion using conditional hierarchical generative adversarial networks (GAN). Instead of newly synthesizing motions that correspond to specific input constraints, our method focuses on transferring the existing motions to other characters with various shapes, while preserving essential characteristics in the original motion.

3 METHOD

This section describes how we prepare a training dataset and train SCT. To prepare a training character dataset, we utilize characters with a uniform mesh connectivity, which are easy to augment by scaling the ratio of characters and reshaping the model. On the other hand, the test characters used to evaluate SCT have varying bone lengths and diverse mesh connectivities. This two-way approach that uses a common representation of characters is an effective way of preparing the dataset to a train network that can retarget to characters with various shapes.

3.1 Motion Representation

We use a template skeleton that is constructed by 22 skeletal joints without finger joints, so that all characters in the training dataset have the identical skeleton structure. Based on this template skeleton, we extract the configuration C in the rest-pose for each character and motion feature m for each motion. Specifically, the character configuration C involves the bone offsets $\gamma \in \mathbb{R}^{3N}$ and the width, height, and depth of the bounding box $\phi \in \mathbb{R}^{3N}$ for all joints in the rest pose, where N denotes the number of joints. The motion sequence m is a set of poses consisting of T frames. The motion feature from each pose is represented with the global position of the root $p_{root} \in \mathbb{R}^3$ and a 6-dimensional local rotation vector $R \in \mathbb{R}^{6N}$ [Zhou et al. 2019] of the entire joints:

$$C = [\gamma, \phi], \tag{1}$$

$$m = [p^1_{root}, R^1_{joint}, p^2_{root}, R^2_{joint} \cdots, p^T_{root}, R^T_{joint}]. \tag{2}$$

3.2 Dataset Augmentation

This section describes the data augmentation method based on a deformation technique. Without loss of generality, we assume that the deformation is performed from a skinny to a chubby character or a short to a tall character. For the clarity of terminology, we also assume that the deformation is performed from a source character to a target character and these characters are referred to as the main characters. The opponent characters for interaction are referred to as partner characters. Please see Figure 2.

3.2.1 Data Augmentation of Characters with the Identical Mesh Connectivity. This section describes the procedure for preparing the training dataset. We use the SMPL model [Loper et al. 2015] as a template character that shares an identical skeleton structure and mesh connectivity across different instances. Preparing the dataset with a uniform character model has several advantages. First, an adaptation method can be easily used to adjust a pose according to the deformation from source to target. We employ a relationship descriptor method [Al-Asqhar et al. 2013] that can adjust the character poses according to the changes in the surrounding environment. In this work, we consider the deformation of the source character as an environmental change and adjust the poses of the partner character. Second, it is natural for an SMPL character to scale the bone length and reshape the mesh by changing shape parameters, which allows easy augmentation to represent various instances in the dataset. Finally, a number of publicly available datasets [Ionescu et al. 2014; Mahmood et al. 2019; von Marcard et al. 2018] are constructed based on SMPL models, and a large set of motion data can

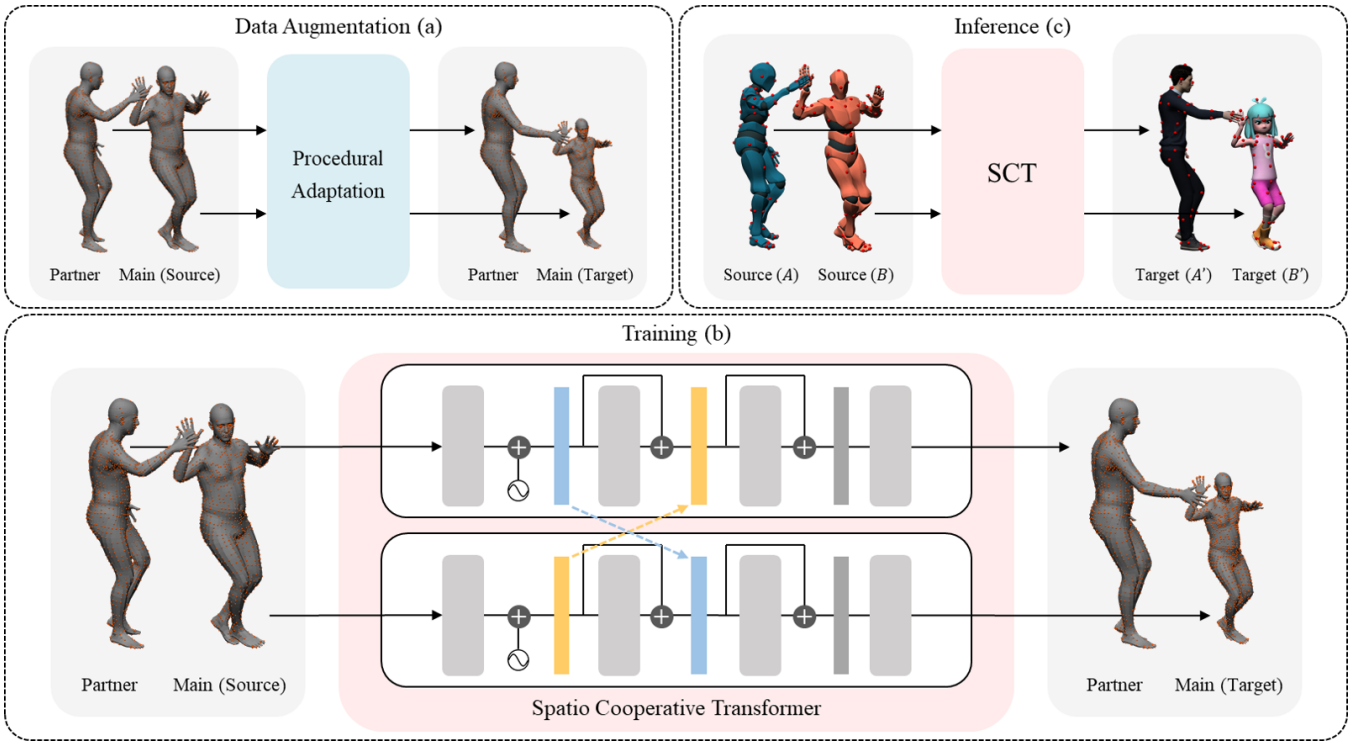


Fig. 2. Overview of the method: In the data augmentation stage (a), we prepare data using characters with the identical mesh connectivity. Orange points on the SMPL model represent the descriptor. In the training stage (b), SCT is trained using the prepared data. In the inference stage (c), retargeting is performed on characters with different mesh connectivities. Red points on the characters represent the anchor points.

be easily obtained from videos by predicting the SMPL parameters using pose estimation approaches [Bogo et al. 2016].

We create a number of target characters by scaling the bone lengths and shape parameters to reshape the body mesh, similar to Won and Lee [2019]. The scale range of characters is determined to cover a wide range of bone length and mesh shape of diverse characters. To scale the bone length, we segment the whole body into three parts: spine, legs, and arms. We then randomly select 10 scaling factors between 0.7 and 1.2, and scale the character accordingly, as shown in Figure 3a. To change the shape of the mesh, we randomly select shape parameters from -10 to 10, as shown in Figure 3b. Additionally, we mirrored the motions with respect to the character's front-facing axis.

3.2.2 Deformation-based Adaptation. During the data preparation step, only one source character is assumed to be changed to the target character, while the partner character remains unchanged as shown in Figure 2(a). The main and partner characters share the same corresponding descriptor points. When training SCT with the augmented dataset, the interaction motions performed by the source characters are used as input to SCT, while the adjusted motions of the target characters are utilized as ground truth data, as shown in Figure 2(b).

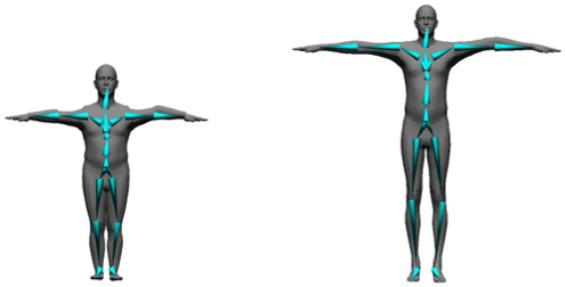
Following the work of Al-Asqhar et al. [2013], we use *descriptor points* to represent the spatial relationship between the main character and the interacting partner character. Because it is defined as a static set of vertex indices on the surface of the mesh, we can sample the same descriptor points from the template character. To distribute the descriptor points evenly throughout the body, we set the number of sampled descriptors proportional to the area of the mesh surface corresponding to each joint. Subsequently, we apply the same descriptor points of the template character to character instances, so that they all share the same descriptor points.

Deformation-based adaptation adjusts the joint position $p_j \in \mathbb{R}^{3N}$ of the partner character to a new position $p'_j \in \mathbb{R}^{3N}$ using the difference of shape between the source and target characters. Figure 4 shows the motion adaptation process occurred depending on the changes from a skinny source character to a chubby target character. The relative vector $\vec{v}_{i,j} = p_j - d_i$ represents a vector starting from a descriptor on the source main character d_i to a joint position of the source partner character p_j. The new joint position of the target partner character p'_j is obtained by the weighted sum of the relational vectors. We set the weight of descriptors to increase exponentially in proportion to the distance r from the descriptor to the joint, as follows:

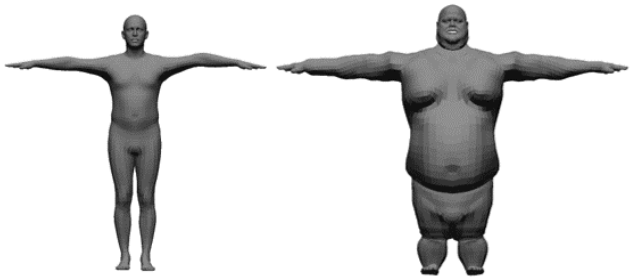
$$w_{i,j} = 1 - \left(\frac{r - r_1}{r_2 - r_1}\right)^{k_{adapt}} \quad \text{if } (r_1 \le r < r_2), \tag{3}$$

$$r_2 = r_1 + h/10, \tag{4}$$

where r_1 is the distance from the joint to the closest descriptor, r_2 is the distance to the descriptor that is a certain distance further from



(a) Characters augmented by bone scaling.



(b) Characters augmented by varying shape parameters.

Fig. 3. Augmented characters. The small and tall characters in the top row are the smallest and tallest one in the training dataset.

r_1 where h is the height of the target main character, and k_{adapt} is the weight that allows for a higher impact for the closer descriptor to the joint.

3.2.3 Procedural Adaptation for Data Augmentation. We procedurally adjust the motion of the source main and partner characters to generate the motion of the target main and partner characters, respectively. Specifically, we first adjust the motion of the partner character and subsequently adjust the motion of the target main character for each frame t. The procedural adaptation process is conducted as follows:

(1) Change the source main character to the target main character.
(2) Copy the motions of the source main and partner characters to those of the target main and partner characters.
(3) Scale the height of the root of the target motion by the ratio between the root height of the source and target characters.
(4) Adjust the pose of the partner character via a deformation-based adaptation and resolve ground penetration artifacts.
(5) Adjust the pose of the target main character.
(6) Resolve the inter-penetration from the motion of both the source main and partner characters.
(7) Apply inverse kinematics (IK) to change the pose representation from the joint positions to rotations.

The main character and partner character are randomly selected, and the same procedure is repeated after swapping their roles.

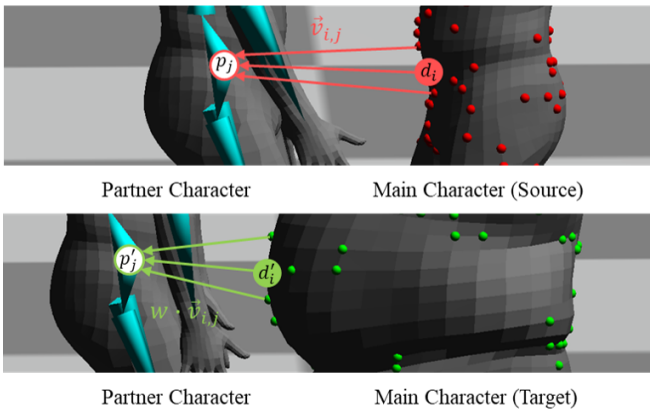


Fig. 4. Deformation-based adaptation adjusts the target pose according to the change of shape. The figures show that the difference of descriptor d_i position on the belly of the source and target characters affects the position of root joint p'_j of the partner character.

3.3 Spatio Cooperative Transformer

This section describes the retargeting method that employs the SCT trained on the prepared dataset. When retargeting two-character interaction motion, there are four characters involved: two source characters and their corresponding target characters. Each source and target character set comprises the main characters, denoted as A and A', and partner characters, denoted as B and B', as illustrated in Figure 2(c).

3.3.1 Input and Output. The input state x consists of the character configuration C of both source and target and motion sequence m of source:

$$x^A = [C^A, C^{A'}, m^A], \tag{5}$$

$$x^B = [C^B, C^{B'}, m^B]. \tag{6}$$

Following Zhang et al. [2023a], the network predicts the residuals y, composed of $resi^A$ and $resi^B$, from the given source motions m^A and m^B, respectively. Thereafter, output motion sequences $m^{A'}$ and $m^{B'}$ of target characters A' and B' can be obtained by applying those residuals to source motions, which are the component-wise addition of the global root position and local rotations of joints:

$$y = [resi^{A'}, resi^{B'}], \tag{7}$$

$$m^{A'} = m^A + resi^{A'}, \tag{8}$$

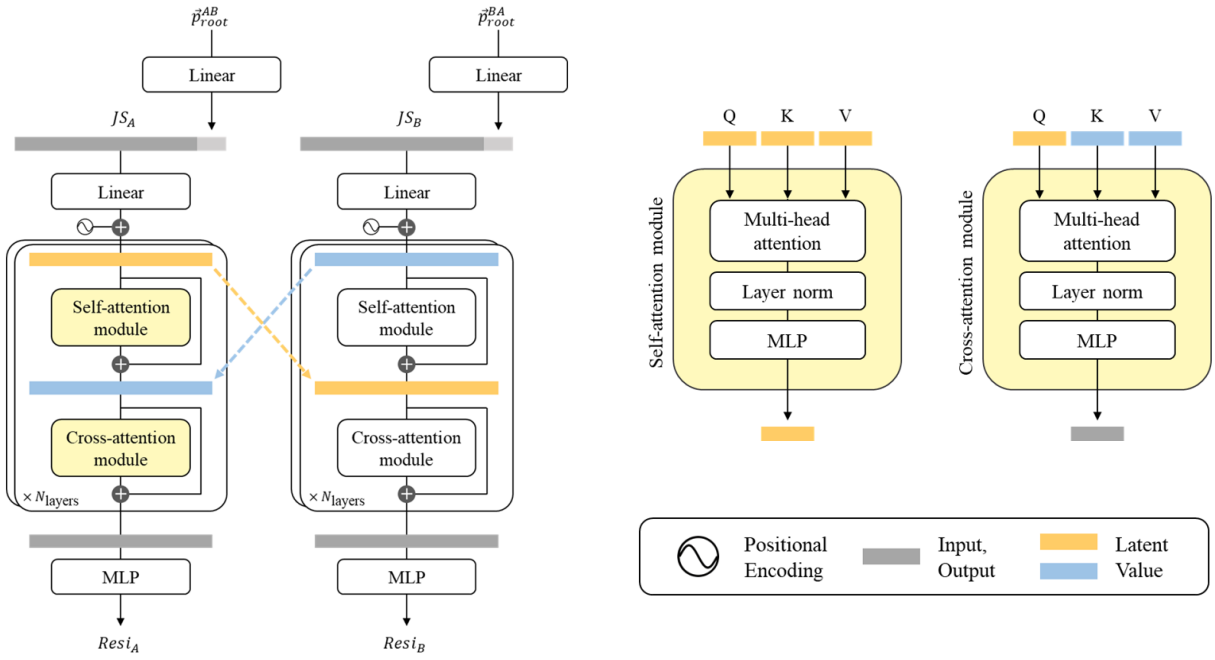
$$m^{B'} = m^B + resi^{B'}. \tag{9}$$

3.3.2 Joint State. A joint state JS_j is embedded into SCT as a token. The joint state consists of the bone offset γ_j, the width, height, and depth of the bounding box ϕ_j, the joint rotation R_j and the root-relative position p_j for the j-th joint:

$$JS_j^A = [\gamma_j^A, \phi_j^A, \gamma_j^{A'}, \phi_j^{A'}, R_j^A, p_j^A], \tag{10}$$

$$JS_j^B = [\gamma_j^B, \phi_j^B, \gamma_j^{B'}, \phi_j^{B'}, R_j^B, p_j^B]. \tag{11}$$

To show the relative displacement from the partner character, the root position of the main character A is represented as the relative



(a) Spatio Cooperative Transform

(b) Attention module

Fig. 5. Overview of Spatio Cooperative Transformer (a) and attention module (b), which illustrates the process in terms of A.

vector from the root position of the partner character B to its root position. The relative root vector $\vec{p}^{BA}_{root}$ is passed through a linear layer ($linear$) to match the dimension with the joint state and is considered as an additional joint token, as shown in the top of Figure 5(a):

$$\vec{p}^{BA}_{root} = p^{A}_{root} - p^{B}_{root}, \tag{12}$$

$$JS^{A}_{N+1} = linear(\vec{p}^{BA}_{root}). \tag{13}$$

Consequently, $N + 1$ joint states are used as the tokens for the transformer:

$$JS^{A} = [JS^{A}_{1}, JS^{A}_{2}, ..., JS^{A}_{N}, JS^{A}_{N+1}], \tag{14}$$

$$JS^{B} = [JS^{B}_{1}, JS^{B}_{2}, ..., JS^{B}_{N}, JS^{B}_{N+1}]. \tag{15}$$

$$\tag{16}$$

3.3.3 Spatio Cooperative Transformer. SCT consists of two symmetric branches of transformer-based networks cooperating to predict motion residual for each character as shown in Figure 5(a). Each branch of the SCT, which consists of N_{layer} transformer layers receives $N + 1$ joint states as input. For each joint state, a linear layer extracts a joint token, which goes through positional encoding. Subsequently, the self and cross-attention modules extract the interaction features considering the main and the partner characters' pose information. Finally, the interaction feature is decoded in the form of the joint rotations and root position by going through an MLP, resulting in the residual value $resi$ of the source motion sequence m.

An attention module consists of N_{head} blocks of multi-head attention, normalization layer, and MLP, as shown in Figure 5(b). The feature is derived by transformer-based multi-head attention [Vaswani et al. 2017] from a query (Q), key (K), and value (V) as follows:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{c_{tk}}})V, \tag{17}$$

where c_{tk} represents the dimension of the joint tokens.

The network shares the interaction features to model the relationship between the motion sequences of two characters. Similar to Liang et al. [2024], SCT utilizes spatial attention that considers both interacting characters in a single frame. Specifically, the first attention layer of SCT extracts its spatial feature for its pose by a self-attention operation where the query, key, and value are computed as follows:

$$Q = W_Q tk^A, \tag{18}$$

$$K = W_K tk^A, \tag{19}$$

$$V = W_V tk^A. \tag{20}$$

W_Q, W_K, W_V are the parameter matrices of query, key, and value. tk^A is the positional encoded joint token. Subsequently, SCT extracts the spatial features of interacting characters through cross-attention operating with its interacting character. The query is from the main

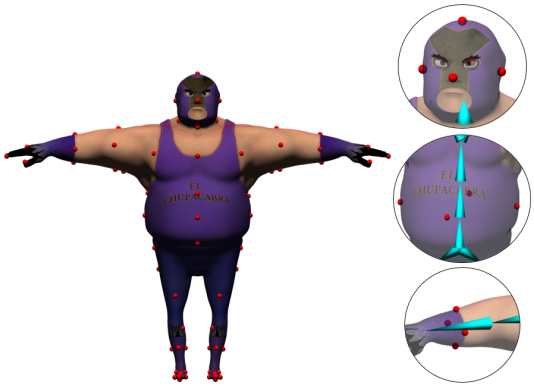


Fig. 6. Anchor positions (red points) for mesh-agnostic characters

character's joint token, and the key and value are from the partner character's token tk^B.

3.4 Anchor

A set of anchors is a sparse and representative vertices on the surface of the character's mesh, which is specified for each joint. A set of anchors is similar to surface correspondences described by Kim et al. [2016] in that they define a relationship between changing geometries. While their surface correspondences are applied to objects, our anchors are defined on characters. As shown in Figure 6, each anchor is located at a representative body part for each character and its vertex index can be different for each character because the characters can have a different mesh connectivity.

3.4.1 Anchor Extraction. We extract the anchor index as follows:

(1) Group each vertex with respect to a joint with its maximum skinning weight for all vertices in a character mesh.
(2) For each grouped vertices, extract an Axis-Aligned Bounding Box (AABB) aligned with the global axis, which spans the vertex group.
(3) Assuming that the up-axis of a joint is the direction from the joint to the child joint, select the plane horizontal to the up-axis and extract the midpoint of the plane.
(4) Define the closest vertex from the midpoint of each plane as the anchor. For each non-end-effector joint, we extract 4 vertices from 4 perpendicular planes as shown in the arm joint in Figure 6.
(5) For each end-effector joint, we extract 5 vertices. The additional vertex is selected from the plane perpendicular to the up-axis. Please refer to the head joint as shown in Figure 6.

Our use of anchors is conceptually similar to Interaction Graph [Zhang et al. 2023a] in that both determine the influence based on the distance between landmarks on characters. The difference is that while Interaction Graph relies on manually pre-specified markers on the body surface, our method automatically identifies representative anchor locations of skinned characters with diverse mesh connectivities and body shapes. This automatic process leads to consistent retargeting results, because the anchor positions remain semantically same across different characters, eliminating the need for each user's judgment when specifying anchors.

3.4.2 Anchor Distance between Interacting Characters. The interaction motion should be retargeted considering the shapes of both the main and the partner characters. If source motions with many close interactions are retargeted to target characters with a chubby body, there easily can be penetrations. To prevent this, the target motions must maintain the same geometric distance that the source motions have. The anchor distance can be used as a sparse representation of the distance between the main and the partner characters. Maintaining the anchor distance between the two interacting characters prevents interpenetration and preserves contact in the target motions.

We introduce an anchor loss for the training of SCT so that the anchor distance between the source characters can be maintained between the target characters after the retargeting. For this training, we use ground truth motions instead of source motions. Training SCT using the ground truth allows the network to experience various shapes of target characters. The ground truth motions maintain the geometric distances of source motions because the ground truth motions are augmented by adaptation based on geometric changes as described in Section 3.2. SCT then accurately transfers the geometric semantics from the ground truth to target, effectively handling unseen characters or motions during training without introducing any artifacts.

3.5 Loss

To train SCT, we employ two loss terms: reconstruction loss and anchor loss. Unlike the previous methods that are trained in an unsupervised manner [Aberman et al. 2020; Lim et al. 2019; Ruben et al. 2021; Villegas et al. 2018; Zhang et al. 2023b], the proposed network is trained in a supervised manner by exploiting the ground truth data acquired as described in Section 3.2. The anchor loss helps SCT to capture important semantics contained in the input interaction motions and to lead those semantics to be preserved in the output motions for unseen characters and motion data. The total loss can be expressed as follows:

$$L(m, \hat{m}) = \lambda_1 L_{rec} + \lambda_2 L_{anchor}. \tag{21}$$

where m and $\hat{m}$ are the input motion and estimated motion, respectively, and λ_1 and λ_2 are the hyper-parameters for the reconstruction and anchor losses, respectively.

3.5.1 Reconstruction Loss. The reconstruction loss consists of the restoration of the root position as well as the rotation and positions of other joints from ground truth. The term for joint position which is computed by Forward Kinematics (FK) helps prevent large positional errors, which can be caused by the error accumulation of joint rotations along the kinematic chain:

$$\begin{aligned} L_{rec}(m^{gt}, \hat{m}) = \lambda_3 ||p^{gt}_{root} - \hat{p}_{root}|| + \lambda_4 ||R^{gt} - \hat{R}|| \\ + \lambda_5 ||FK(m^{gt}, \gamma) - FK(\hat{m}, \gamma)||, \end{aligned} \tag{22}$$

where m^{gt} is the ground truth motions, R^{gt} is the joint rotations of ground truth motions and $FK(\cdot,\cdot)$ is the joint positions in the world-coordinate calculated by forward kinematics given bone offsets and joint rotations.

3.5.2 Anchor Loss. Similar to Zhang et al. [2023a], we extract the distance matrix $D \in \mathbb{R}^{N_{anchor} \times N_{anchor}}$ that measures the distance between anchors of interacting source characters, where N_{anchor} denotes the number of anchors designated on a single character. By utilizing the distance matrices obtained from both the output and ground truth motions, the anchor loss is defined as follows:

$$L_{anchor}(m^{gt}, \hat{m}) = \omega ||D(m^{gt,A}, m^{gt,B}) - D(\hat{m}^{A'}, \hat{m}^{B'})||, \quad (23)$$

$$\omega = 1/exp(-k_{anchor} * D(m^{A,B})), \quad (24)$$

where k_{anchor} determines the sensitivity of weights between anchors, which adjusts the degree of influence based on the proximity of the anchors.

4 EXPERIMENTS

We compared our method with other baselines both quantitatively and qualitatively. We also performed an ablation study to validate the necessity of each component. Finally, to evaluate the perceptual naturalness of resulting motions, we conducted a user study.

Dataset. To gather training data, we retargeted the CMU interaction dataset and HAC dataset [Shen et al. 2020] to the SMPL model. We additionally captured complex interaction motions performed by two people, such as hugging. The initially collected data is consisted of 18,099 frames in 30 FPS. After conducting the data augmentation, the amount of total data became about a length of 6.7 hours, resulting in 723,960 frames. For details of the data augmentation process, please refer to Section 3.2.1.

To evaluate SCT, we collected unseen motion and character sets. We collected 4 characters with different mesh connectivities from Mixamo [Adobe 2021], differing from the mesh connectivity used for training. The unseen motions were created by retargeting the CMU interaction dataset to unseen Mixamo characters with mirroring with respect to the facing axis. We separated the motions into simple and complex interactions, in a similar way to Zhang et al. [2023a]. Characters in simple interaction motions are in contact for a short duration or at a close distance without contact with a partner character, such as handshaking, greeting, jumping, crawling, and punching. Characters in complex interaction motions are continuously in contact with a partner character, such as salsa dance, lift-pushup, and kick-back.

Baselines. We compared our method with various existing motion retargeting methods. These include a simple copy of the source motion to the target character, along with deep-learning-based methods such as PMnet [Lim et al. 2019], NKN [Villegas et al. 2018] and R2ET [Zhang et al. 2023b]. Both PMnet and NKN are skeleton-aware retargeting approaches designed for a single character, while R2ET extends its consideration to mesh information. Additionally, we evaluated our method in comparison to AuraMesh [Jin et al. 2018], an optimization-based approach specifically designed for retargeting two-character interaction motions. Notably, AuraMesh works only for characters with an identical mesh connectivity, specifically SMPL models. To ensure a fair comparison, we exclusively utilized SMPL models when evaluating ours against AuraMesh.

Implementation Details. We set the hyper-parameters as follows: $\lambda_1 = 1.0, \lambda_2 = 10.0, \lambda_3 = 100.0, \lambda_4 = 1.0, \lambda_5 = 100.0, k_{adapt} = 0.1, k_{anchor} = 3.0, N_{layers} = 2, N_{layers} = 2$. We implemented the SCT network based on PyTorch and optimized it using the Adam optimizer [Kingma and Ba 2015] on RTX A5000, spending 12 hours for training the model with 8,000 epochs. To resolve the ground penetration artifacts of the resulting motion, we applied IK as a post-processing. As an alternative, we also tried to incorporate a foot contact loss to maintain the ground contact of the foot joint. We found that the latter did not lead to noticeable qualitative improvement in preserving foot contact compared to the former, making post-processing still necessary. Therefore, we opted to use the first choice for simplicity. For details, please refer to Section 4.6.

4.1 Qualitative Evaluation

We analyzed the results for each motion qualitatively as shown in Figures 7 and 8. We show the effectiveness of SCT in two aspects: different body proportions and mesh shapes. Specifically, to examine the influence of changing the morphology of one character, one source character (A) is retargeted to a character with a similar shape (A'), and the other character (B) is retargeted as a short, chubby, and big-sized characters (B'). To evaluate the effect caused by different body proportions, the motion of source character B was retargeted to a short target character B'. To evaluate the effect caused by different mesh shapes, the motion of source character B was retargeted to a chubby target character B'. Please refer to the supplementary video for the animated results.

Simple Interactions. For rapper-style greeting motion, the source characters bump their hands, feet, and shoulders as shown in Figure 7. To reach the partner character's hand, our method makes character A' bend his knee and lower his arm, while the short character B' raises her arm higher. In contrast, the retargeted motions of the baselines hardly preserve interacting semantics. SCT recognizes the bone offsets of target characters to generate their interaction motions, whereas the baselines only perform retargeting on their own characters, leading to improper interactions.

Complex Interactions. Figure 8 shows the result of retargeting a salsa dance motion in which a character B leans against partner character A. For this experiment, target characters with ordinary shapes A' and various shapes B' are used. In the "Short" column, character A' lowers the right knee so that the short character B' can lean on character A''s knee. In the "Chubby" and "Big" columns, the hand on the waist is likely to penetrate the body of B' if the rotation values of the source characters are simply copied and pasted to the target characters. Because SCT estimates the residual in consideration of the mesh shape of the target characters, the partner character's hand is located on the waist properly. Furthermore, character A' leans the back further to avoid a collision with chubby character B'. This shows that SCT can handle the retargeting of motion robustly from source to target characters with significantly different body shapes. Figure 9 shows the results generated by the baselines

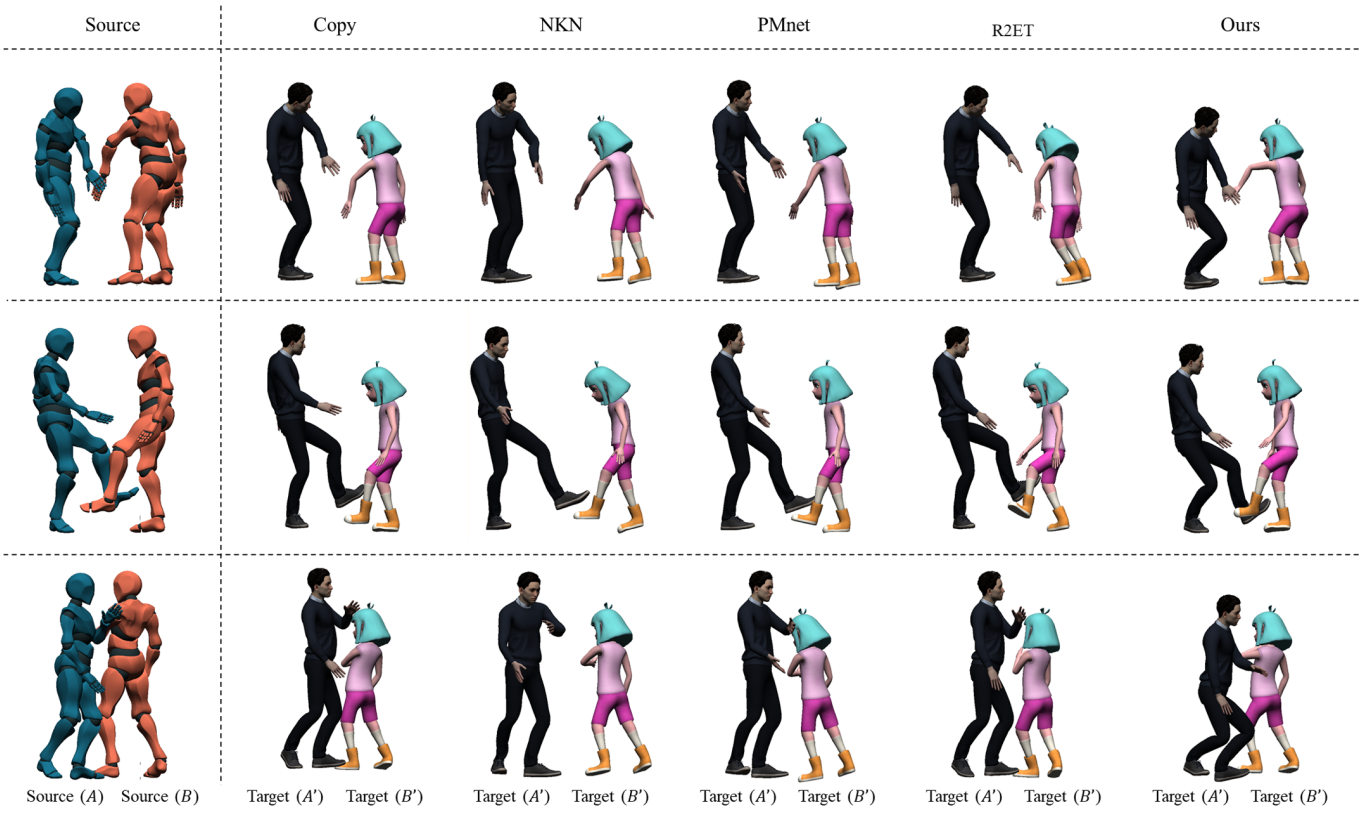


Fig. 7. Simple motions in source characters (left column) are retargeted to ordinary and short target characters to show skeletal changes (right columns).

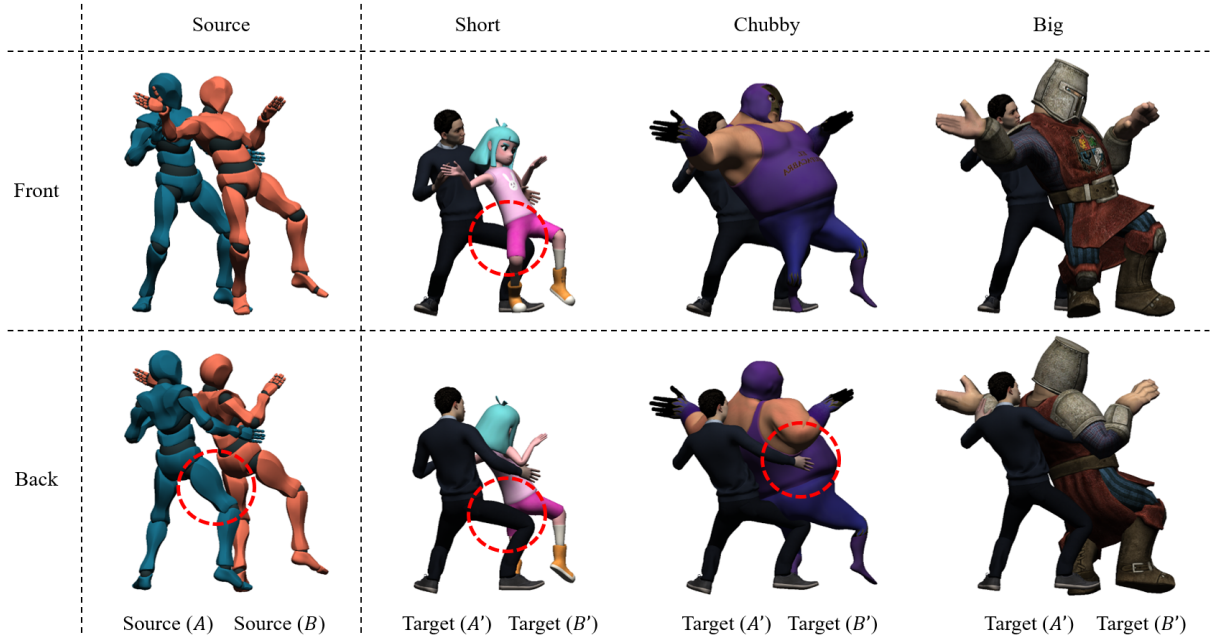


Fig. 8. Complex dance motions in source characters are retargeted to short, chubby, and big characters. The top row is from the front view and the bottom row is from the back view.

and our method with the same salsa dance motion. In contrast to baseline methods that result in overlapping bodies, the interaction motions of two characters from SCT do not overlap. Please refer to the supplementary video for the animated results.

4.2 Quantitative Result

We compared the results of our method to those of the baselines for interaction motion quantitatively, as shown in Table 1. Specifically, we evaluated semantic preservation and artifact avoidance for comparison. The joint-wise and anchor-wise distances were measured to verify if the output motions preserve the semantics of source motions using Equation 23. We assign a larger weight for closer joints or anchors, allowing them to have a greater influence when measuring joint-wise and anchor-wise distances.

We detected contacts between interacting body parts by checking an intersection between the AABBs of the faces for each body part. To evaluate the quality of artifact avoidance in the retargeted motion, we define contact evaluation metrics as follows:

(1) Contact preservation refers to how well the target motions maintain the contact between two body parts observed in the source motions. Contact preservation is represented as true positive (TP).
(2) Contact missing refers to the failure of the target to maintain the contact observed in the source motions. Contact missing is represented as false negative (FN).
(3) Improper contact measures how often contacts are made in the target motions when there is no contact presented in the source motions. Improper contact is represented as false positive (FP).
(4) Non-contact preservation measures how often the target correctly maintains non-contact from the source to target motions. Non-contact preservation is represented as true negative (TN).

From contact evaluation metrics, we calculated precision, recall, and accuracy to compare differences between our model and baselines. The precision measures how accurately the detected contact of the target motions matches that of the source motions. The recall measures how well all contacts from the source motions are transferred to target motions. Finally, the accuracy evaluates the overall performance. Each metric is computed as follows:

$$Precision = \frac{TP}{TP + FP}, \tag{25}$$

$$Recall = \frac{TP}{TP + FN}, \tag{26}$$

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN}. \tag{27}$$

We compared the value of retargeted motion from *ours* with *baselines* (R2ET, PMnet, NKN, and Copy) that deal with retargeting for a single character. As shown in the left column of Table 1, our method achieved better results for semantic preservation compared to the baselines. Specifically, for semantic preservation between joints, our method demonstrated the lowest value 0.036 outperforming the baselines. Similarly, for semantic preservation between meshes, our method also achieved the lowest value of 0.313, further highlighting its effectiveness.

Table 1. Quantitative comparison with baselines for interaction motions and the result of SCT without using anchor and FK loss for ablation. Bold indicates the best value, and underline indicates the second-best value.

	Semantic preservation		Artifact avoidance		
Methods	Skeletal↓	Anchor↓	Precision↑	Recall↑	Accuracy↑
SCT (ours)	<u>0.036</u>	0.313	**0.452**	<u>0.976</u>	**0.894**
R2ET	0.102	0.685	0.203	0.949	0.619
PMnet	0.236	1.037	0.172	**0.980**	0.602
NKN	0.230	1.027	0.216	0.925	0.624
Copy	0.037	0.354	0.348	0.963	0.750
SCT_{woAnc}	**0.033**	**0.301**	<u>0.445</u>	0.975	<u>0.882</u>
SCT_{woFK}	<u>0.036</u>	<u>0.309</u>	**0.452**	0.963	0.881

As shown in the right column of Table 1, we also compared our method with baselines in artifact avoidance. In terms of precision, our method achieved the best precision score of 0.452, indicating that the retargeted motion from SCT preserves the contacts in source motions better than the baselines. For recall, our method shows the best result with a score of 0.976. Regarding accuracy, our method achieved a score of 0.894, demonstrating that the retargeted motion from SCT is effective in accuerately preserving contacts and minimizing unnecessary contacts.

4.3 Comparison with AuraMesh

We conducted a comparative experiment with AuraMesh, dealing with retargeting interaction motions between two characters with the identical mesh connectivity. In Figure 10, a qualitative comparison with simple and complex interaction motions is presented with the main character in red and the partner character in gray. For both simple and complex interaction motions, the performance of our method excelled that of AuraMesh as our method did not suffer from jittering and produced the intended contact properly. AuraMesh determined the extent of updates based on the size of the collided volume between the two character meshes, which is not smooth and continuous across frames thereby creating jittering. Interpolation was applied to alleviate the situation at the cost of often losing details of source motions, such as contact-missing. In contrast, our method better preserves existing contacts between interacting characters, because it was trained on a dataset obtained from continuous adaptation.

Table 2 reports the computation time required by our method and AuraMesh for retargeting interaction motions. Our method is approximately 588 times faster for simple motions of 495 frames and 764 times faster for complex motions of 999 frames. The significantly higher computation time required by Auramesh was due to the need for collision detection between two characters and frame-by-frame optimization. Table 3 shows that while AuraMesh achieves better results in semantic preservation, our method excels in artifact avoidance. This indicates that our method performs comparably to the state of the art method on retargeting motions between characters having an identical mesh connectivity. It is worth noting that our method can retarget interaction motions on characters with different mesh connectivities as well.

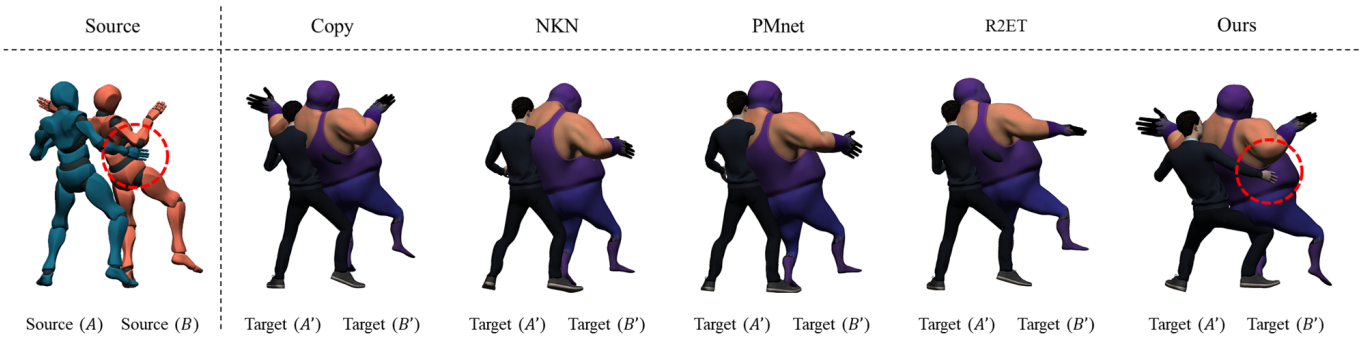


Fig. 9. Complex dance motion retargeted to a chubby character by the baselines and ours.

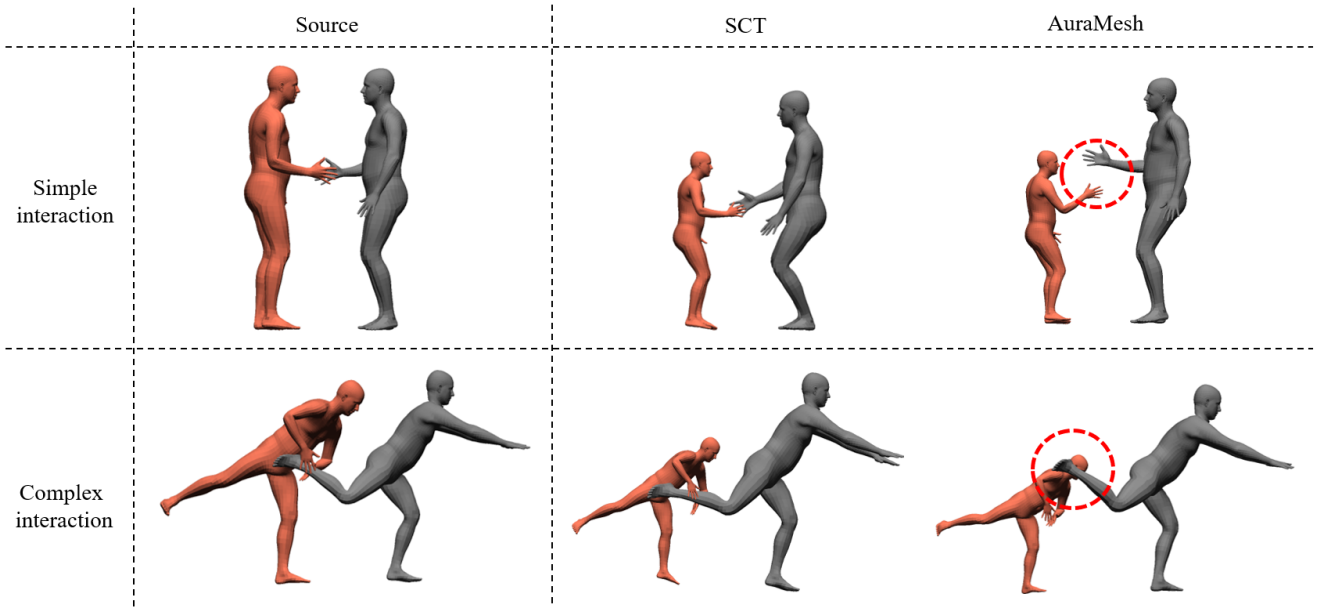


Fig. 10. Comparison of motions retargeted by our method and AuraMesh. Motions retargeted by SCT are of higher quality than those retargeted by AuraMesh without jittering.

Table 2. Computation time comparison with AuraMesh.

Time (second) ↓	Simple motions	Complex motions
SCT (ours)	**0.572**	**1.087**
AuraMesh	336.470	830

Table 3. Quantitative comparison with AuraMesh.

	Semantic preservation		Artifact avoidance		
Methods	Skeletal↓	Anchor↓	Precision↑	Recall↑	Accuracy↑
SCT (ours)	0.036	0.313	**0.452**	**0.976**	**0.894**
AuraMesh	**0.030**	**0.210**	0.343	0.920	0.728

4.4 Ablation study

We performed ablation studies to assess the effectiveness of each loss term. We analyzed the retargeted motions from SCT trained without using the FK loss (SCT_{woFK}) and anchor loss (SCT_{woAnc}), focusing on both qualitative and quantitative evaluations. For quantitative comparison, the results from $SCT(ours)$, SCT_{woAnc} and SCT_{woFK} show no significant difference as shown in the lower part of Table 1. On the other hand, there was a significant difference in the qualitative comparison. Figure 11 shows the visual difference of output motions generated with and without using the anchor loss. The red point is the anchor on the hand of the source character A and target character A'. The blue point is the anchor on the waist of the source character B and target character B'. In source motions, the blue and red points are the closest anchors for the hand of the character A and the waist of the character B, respectively. To preserve the contact

(a) Source motion (b) With anchor loss (c) Without anchor loss

Fig. 11. Source motions (a) with the anchor of character A (red) and that of character B (blue) are retargeted to target motions using SCT trained with the anchor loss (b) and without the anchor loss (c). The target character B' on the right is presented transparently to show interpenetration.

Table 4. Results of user study. Both expert and non-expert users participated to evaluate the retargeted interaction motions. The best result for each column is in bold.

	Expert users	
Methods	Semantic preservation ↑	Artifact avoidance ↑
SCT (ours)	**4.347**	**4.240**
R2ET	1.733	1.740
PMnet	2.827	2.660
NKN	3.033	2.793
Copy	2.900	3.547
	Non-expert users	
Methods	Semantic preservation ↑	Artifact avoidance ↑
SCT (ours)	**4.433**	**4.333**
R2ET	1.733	1.133
PMnet	2.827	2.667
NKN	3.033	2.633
Copy	2.900	3.800

between the source and target characters, the distance between the anchor on the hand (blue) and the anchor on the waist (red) should be maintained in target motions as shown in Figure 11b. For motions produced without using the anchor loss, the hand of character A penetrates into the belly of character B as shown in Figure 11c. Maintaining this anchor distance prevents interpenetration and also makes the proper contacts in the target motions.

4.5 User study

We performed a user study to compare the visual quality of the results produced by SCT against that of the results produced by baselines (R2ET, PMnet, NKN, and Copy). We recruited 20 participants (11 males and 9 females; ages from 23 to 40) including 3 expert animation artists. Participants were asked to evaluate 10 motion sequences for 5 characters so that a total of 50 tasks were performed. We randomly placed the motions that are retargeted by SCT and other baselines. Each participant was asked to answer 2 questions on a 5 Likert scale for each motion sequence: semantic preservation and visual quality without artifacts, such as penetration and jittering.

The result of the user study is summarized in Table 4. We found that SCT gained the highest user preferences compared to other methods in both semantic preservation and artifact avoidance from all participants. In semantic preservation, the retargeted motions of SCT were the most preferred, scoring 4.347 by experts and 4.433 by non-experts. For artifact avoidance, the retargeted motions of SCT were again the most preferred, scoring 4.240 by experts and 4.333 by non-experts. This result demonstrates the effectiveness our method in generating plausible retargeted results on human perception.

4.6 Applying Foot Contact loss

The motions retargeted through SCT sometimes accompany foot penetration into the ground. Applying a foot contact loss can be a remedy for this situation [Ruben et al. 2021]. A set of foot joints, such as heels and toes, in source motions are identified as being in contact with the ground if their heights are below a threshold τ. We applied the thresholds for each toe and heel joints as τ_{toe} and τ_{heel}, respectively. The foot contact loss is applied as follows:

$$L_{foot}(m, \hat{m}) = \lambda_6 \mathcal{M}(p_{foot}) * ||p_{foot} - \hat{p}_{foot}||, \tag{28}$$

where $\mathcal{M}$ is the binary mask, with 1 indicating contact and 0 indicating non-contact. Foot contact is detected by comparing the height of joint position p_{foot} with the ground threshold τ. τ is set to 0.04 for the toe joint and 0.11 for the heel joint, and λ_6 is set to 1.0.

Table 5 shows the foot contact accuracy improvement achieved using the foot contact loss. Despite the quantitative accuracy improvement, post-processing using IK was still required when two characters interacted with each other at a very close distance as shown in Figure 12. This is because the influence of the anchor loss easily becomes more dominant than that of the foot contact loss. As a result, the root shifts downward, making the foot penetrating into the ground.

4.7 Comparison with Procedural Adaptation

Retargeting interaction motion between characters with different connectivities can also be achieved using the procedural adaptation method presented in Section 3.2.2. We compared the results produced by SCT and by the procedural method, focusing on the computation time of the retargeting process and the visual quality of resulting motions. Because characters with different mesh connectivities do not share the same relationship descriptor, we instead utilized the anchor index of each character as the relationship descriptor for procedural adaption.

Figure 13 shows a visual assessment of retargeted motions. Unlike the motion retargeted using SCT, the motion from the adaptation method was not satisfactory. For example, the hands often penetrated through the other character's foot. This artifact occurred because anchors were too sparse to be used as descriptors to represent the mesh, failing to capture the details of body parts, such as

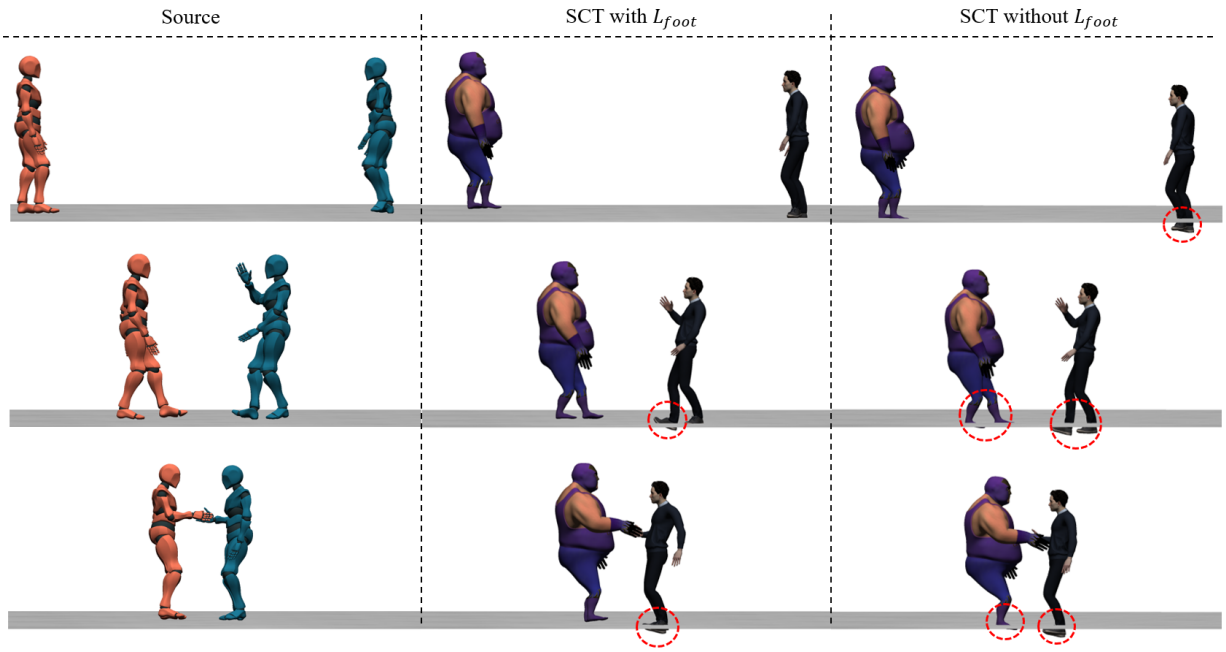


Fig. 12. Retargeted motions through SCT trained with and without a foot contact loss.

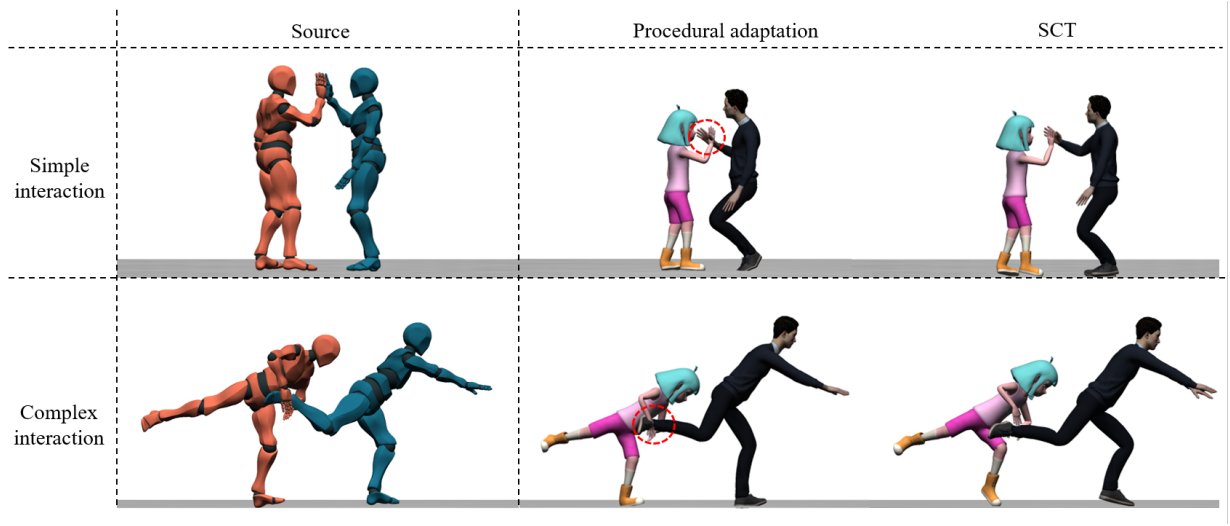


Fig. 13. Visual comparison of retargeted motions produced using procedural adaptation and SCT

Table 5. Comparison of foot contact preservation accuracy produced with and without a foot contact loss.

	Accuracy ↑ (Percent)
SCT with L_{foot}	**0.495**
SCT without L_{foot}	0.339

Table 6. Comparison of the time required for procedural adaptation and inference from SCT. The time is in second.

Time (second) ↓	Procedural adaptation	SCT
Simple interaction	15.883	0.548
Complex interaction	23.499	0.93

the hands. In contrast, the target motions were correctly retargeted using SCT because the SCT was trained using data obtained by dense descriptor points.

Table 6 indicates that the adaptation method requires longer computation time than the inference of SCT. The adaptation method involves nested loops over descriptor points for both the main and partner characters at every frame, resulting in a time complexity of $O(n^3)$. In contrast, the computation of SCT is very fast, enabling it to be applied to real-time interactive applications, such as live motion capture and games.

4.8 Level of Deformation

We investigated the extent of acceptable deformation of the target main character that still results in realistic retargeted motions. We changed the length and volume of a character in a similar way to Won and Lee [2019] and examined the effect of the change on the retargeted motions. For the test set characters, we changed the length of skeletal bones, from short to tall. We also changed the volume of the character's mesh while keeping the skeleton length unchanged, from skinny to bulky.

4.8.1 Height Variation. We examined the naturalness of the motions retargeted to extremely short and extremely tall characters. As mentioned in Section 3.2.1, SCT is trained with the character dataset within the scale range from 0.7 to 1.2. We tested the model using characters scaled by 0.5, 0.6, 1.3, 1.4, which of these are beyond the trained ranges. The short character is shorter than the shortest one and the tall character is taller than the tallest one in the training set. The results are shown in Figure 14a. Characters shorter than the training range tended to float slightly above the ground. In the simple interaction, the character scaled by 0.5 failed to touch the hand of the partner. The characters taller than the training range attempted to touch their partner's hand by bending their knees and their partner raised the hand to maintain contact but failed to do so. Furthermore, in the complex interaction, the tall main characters failed to grab the foot of the partner character. This demonstrates that the contact points of interaction can be preserved only when the retargeting is performed within the trained range.

4.8.2 Volume Variation. We also examined the naturalness of the motions retargeted to extremely skinny and extremely bulky characters. For the extremely skinny character, we scaled down the volume of the normal character's mesh by 0.6 and 0.8. For the extremely bulky character, we scaled up the volume of the bulky character's mesh by 1.2 and 1.4. The results are shown in Figure 14b. When the simple interaction motions are retargeted, the skinny characters bent the knee. This occurs because the skinny characters were not included in the training dataset, and therefore SCT likely misinterprets the skinny characters as small characters from the reduction of the mesh thickness. For the motion retargeted to the bulky characters, the partner's waist bent backward unnaturally to maintain a distance from the bulky main character. In the complex interaction, the skinny characters failed to make proper contact with the partner character's foot. For bulky characters, the partner's foot penetrated into the bulky main characters' belly. Similar to height variation, this demonstrates that the contact points of interaction can be preserved only when the retargeting is performed within the trained range for volume variation.

5 CONCLUSIONS

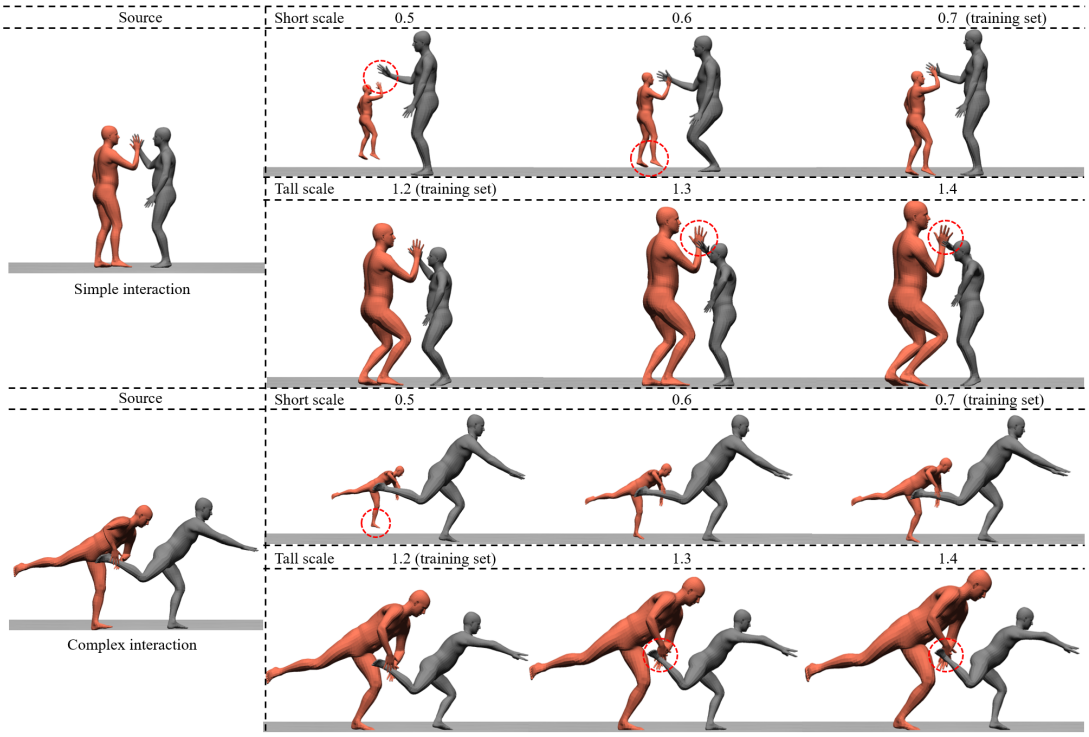
We present a novel network, SCT, for the retargeting of interaction motions between two skinned characters with different mesh connectivities. SCT incorporates the skeletal structures and shape information from both source and target characters to estimate motion residuals, using a novel representation called anchor, which describes the posed shape of a character in a mesh-agnostic manner. To enhance retargeting performance, we propose a data augmentation method based on deformation-based motion adaptation. We show that the proposed method outperforms the baseline methods in preserving semantic context and avoiding artifacts through experiments. Additionally, we validate the effectiveness of each component of SCT through ablation studies.

The retargeting process between two characters can yield equally valid diverse results, because multiple candidate body parts can be adjusted to preserve interaction semantics. Our method specifically addresses the preservation of distances between the closest anchors of two characters. This strategy ensures that the most significant adjustments occur at the closest interaction point. The loss propagates through the skeleton, affecting closer joints more and distant joints less. This distance-based mechanism allows the model to dynamically adjust to different character sizes and shapes, ensuring realistic interactions across characters having different proportions.

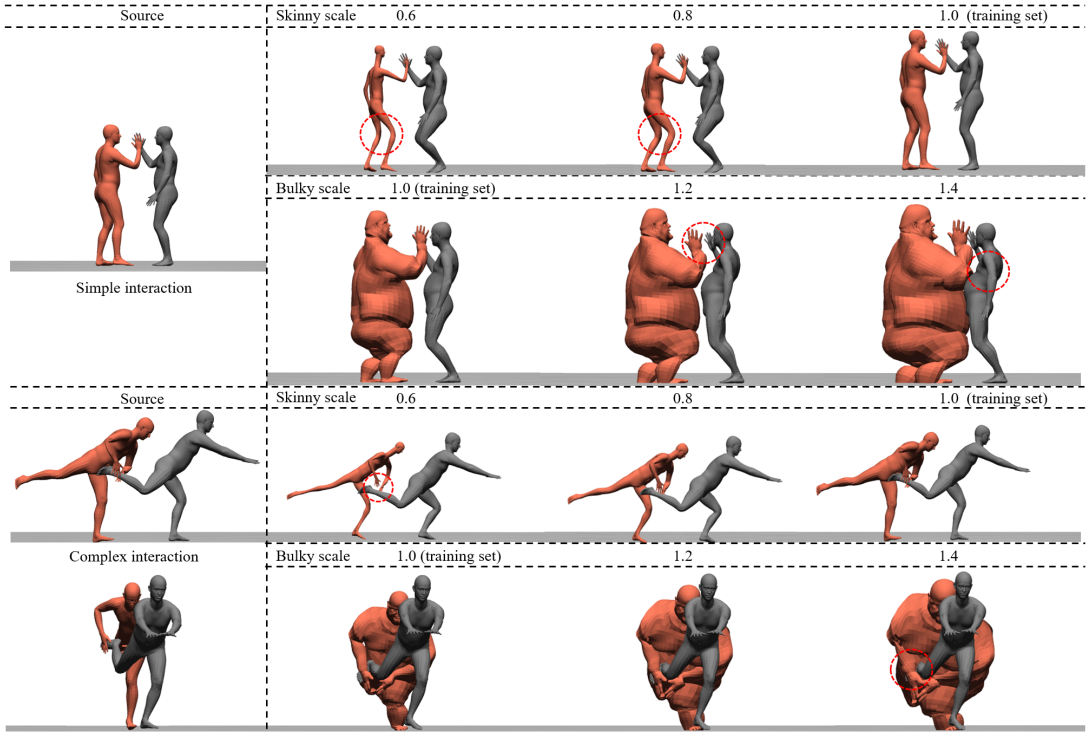
Limitations and Future Work. Retargeting motions of close interactions sometimes produced unnatural results, such as knee bending of the partner character. This artifact does not occur when SCT was trained solely with a reconstruction loss and occurs as the anchor loss starts to exert a significant influence on the partner character. This issue may be addressed by employing a vision-language model to extract high-level motion semantics, similar to Zhang et al. [2024]. Maintaining the semantic meaning of interaction will help SCT retarget motions naturally even for very close interactions.

There are other limitations that remain unresolved in specific scenarios. As shown in Figure 15, when a big target character attempts to crawl through a small space below a partner character, the preservation of anchor distances in the source motion eventually leads the partner character to float in the air, resulting in motions that violate the physical laws. In such cases, incorporating physics simulations alongside the retargeting process can produce physically plausible motions. This approach would help ensure that the motions adhere to the laws of physics, even in challenging scenarios, thereby enhancing the realism and applicability of the retargeted motions.

Even though characters with the identical mesh connectivity can be adjusted by an adaptation method, the characters sometimes cannot reach the target position, because of the fixed bone lengths. As shown in Figure 16, character A' attempts to wrap its arms around character B', while the root position of character A' is adjusted to prevent their bellies from overlapping. However, the arms of character A' are too short to reach the back of character B', leading to inter-penetration occurred between the arms of A' and the body of B'. In such cases, the contact repositioning method [Tonneau



(a) Retargeted motions produced when the target main character (red) changes from the normal character to either an extremely short or extremely tall character.

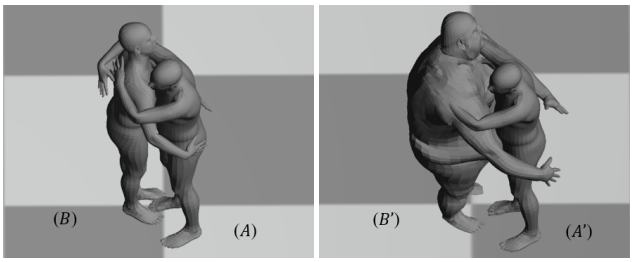


(b) Retargeted motions produced when the target main character (red) changes from the normal character to either an extremely skinny or extremely bulky character.

Fig. 14. The retargeted motions at different levels of deformation.

(a) Source motion (b) Retargeted motion

Fig. 15. Crawling motions for source characters (a) are retargeted to target characters (b). When a large target character tries to crawl under a smaller one, preserving anchor distances causes the character to float in the air unrealistically.



(a) Source motion (b) Retargeted motion

Fig. 16. Hugging motions of source characters (A and B) are retargeted to ordinary (A') and big (B') characters. Retargeted motions of the characters fail to hug each other due to unreachable target positions.

et al. 2016] can be applied to resample the descriptor to a location where the character's arm can reach.

We do not address the relationship of a single character when retargeting interaction motions between characters. This can lead to self-penetration between the limbs and body of a character. Furthermore, the proposed method cannot be applied to characters with different skeletal structures, because SCT requires the skeleton structure to be identical. We expect that by using retargeting methods designed for various skeletons, such as Aberman et al. [2020] and Lee et al. [2023], SCT will be able to retarget motions to characters with diverse skeletal structures.

ACKNOWLEDGEMENT

We appreciate the anonymous reviewers for the valuable comments; Changwook Seo and Hyerin Koo for editing the animation data. This research was supported by the Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2024 (RS-2024-00440434).

REFERENCES

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.

Adobe. 2021. *Adobe Animation Dataset.* https://www.mixamo.com/

Rami Ali Al-Asqhar, Taku Komura, and Myung Geol Choi. 2013. Relationship descriptors for interactive motion adaptation. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Anaheim, California) *(SCA '13)*. Association for Computing Machinery, New York, NY, USA, 45–53. https://doi.org/10.1145/2485895.2485905

Mazen Al Borno, Ludovic Righetti, Michael J Black, Scott L Delp, Eugene Fiume, and Javier Romero. 2018. Robust Physics-based Motion Retargeting with Realistic Body Shapes. In *Computer Graphics Forum*, Vol. 37. Wiley Online Library, 81–92.

Jean Basset, Stefanie Wuhrer, Edmond Boyer, and Franck Multon. 2019. Contact Preserving Shape Transfer For Rigging-Free Motion Retargeting. In *Proceedings of the 12th ACM SIGGRAPH Conference on Motion, Interaction and Games* (Newcastle upon Tyne, United Kingdom) *(MIG '19)*. Association for Computing Machinery, New York, NY, USA, Article 24, 10 pages. https://doi.org/10.1145/3359566.3360075

Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. 2016. Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image. In *Computer Vision – ECCV 2016 (Lecture Notes in Computer Science)*. Springer International Publishing.

Théo Cheynel, Thomas Rossi, Baptiste Bellot-Gurlet, Damien Rohmer, and Marie-Paule Cani. 2023. Sparse Motion Semantics for Contact-Aware Retargeting. (2023).

Kwang-Jin Choi and Hyeong-Seok Ko. 1999. On-line motion retargetting. In *Proceedings. Seventh Pacific Conference on Computer Graphics and Applications (Cat. No.PR00293)*. 32–42. https://doi.org/10.1109/PCCGA.1999.803346

Anindita Ghosh, Rishabh Dabral, Vladislav Golyanik, Christian Theobalt, and Philipp Slusallek. 2023. ReMoS: 3D Motion-Conditioned Reaction Synthesis for Two-Person Interactions. In *arXiv*.

Michael Gleicher. 1998. Retargetting motion to new characters. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*. Association for Computing Machinery, New York, NY, USA, 33–42. https://doi.org/10.1145/280814.280820

Aman Goel, Qianhui Men, and Edmond SL Ho. 2022. Interaction Mix and Match: Synthesizing Close Interaction using Conditional Hierarchical GAN with Multi-Hot Class Embedding. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 327–338.

Edmond S. L. Ho, Taku Komura, and Chiew-Lan Tai. 2010. Spatial relationship preserving character motion adaptation. *ACM Trans. Graph.* 29, 4, Article 33 (jul 2010), 8 pages. https://doi.org/10.1145/1778765.1778770

Edmond S. L. Ho and Hubert P. H. Shum. 2013. Motion adaptation for humanoid robots in constrained environments. In *2013 IEEE International Conference on Robotics and Automation*. 3813–3818. https://doi.org/10.1109/ICRA.2013.6631113

Lei Hu, Zihao Zhang, Chongyang Zhong, Boyuan Jiang, and Shihong Xia. 2023. Pose-aware attention network for flexible motion retargeting by body part. *arXiv preprint arXiv:2306.08006* (2023).

Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. 2014. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, 7 (jul 2014), 1325–1339.

Taeil Jin, Meekyoung Kim, and Sung-Hee Lee. 2018. Aura Mesh: Motion Retargeting to Preserve the Spatial Relationships between Skinned Characters. *Computer Graphics Forum* 37, 2 (2018), 311–320. https://doi.org/10.1111/cgf.13363 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13363

Seong Uk Kim, Hanyoung Jang, and Jongmin Kim. 2020. A variational U-Net for motion retargeting. *Computer Animation and Virtual Worlds* 31, 4-5 (2020), e1947.

Yeonjoon Kim, Hangil Park, Seungbae Bang, and Sung-Hee Lee. 2016. Retargeting Human-Object Interaction to Virtual Avatars. *IEEE Transactions on Visualization and Computer Graphics* 22, 11 (2016), 2405–2412. https://doi.org/10.1109/TVCG.2016.2593780

Diederik Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*. San Diega, CA, USA.

Sunmin Lee, Taeho Kang, Jungnam Park, Jehee Lee, and Jungdam Won. 2023. SAME: Skeleton-Agnostic Motion Embedding for Character Animation. In *SIGGRAPH Asia 2023 Conference Papers* (Sydney, NSW, Australia) *(SA '23)*. Association for Computing Machinery, New York, NY, USA, Article 45, 11 pages. https://doi.org/10.1145/3610548.3618206

Han Liang, Wenqian Zhang, Wenxuan Li, Jingyi Yu, and Lan Xu. 2024. Intergen: Diffusion-based multi-human motion generation under complex interactions. *International Journal of Computer Vision* (2024), 1–21.

Jongin Lim, Hyung Jin Chang, and Jin Young Choi. 2019. PMnet: Learning of Disentangled Pose and Movement for Unsupervised Motion Retargeting. In *British Machine Vision Conference (BMVC)*.

C Karen Liu, Aaron Hertzmann, and Zoran Popović. 2006. Composition of complex optimal multi-character motions. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*. 215–222.

Zhiguang Liu, Antonio Mucherino, Ludovic Hoyet, and Franck Multon. 2018. Surface based motion retargeting by preserving spatial relationship. In *Proceedings of the 11th ACM SIGGRAPH Conference on Motion, Interaction and Games* (Limassol, Cyprus) *(MIG '18)*. Association for Computing Machinery, New York, NY, USA, Article 7, 11 pages. https://doi.org/10.1145/3274247.3274507

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. 2015. SMPL: a skinned multi-person linear model. *ACM Trans. Graph.* 34, 6,

Article 248 (oct 2015), 16 pages. https://doi.org/10.1145/2816795.2818013
Etienne Lyard and Nadia Magnenat-Thalmann. 2008. Motion adaptation based on character shape. *Computer Animation and Virtual Worlds* 19, 3-4 (2008), 189–198.
Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. 2019. AMASS: Archive of Motion Capture as Surface Shapes. In *International Conference on Computer Vision*. 5442–5451.
Villegas Ruben, Ceylan Duygu, Hertzmann Aaron, Yang Jimei, and Saito Jun. 2021. Contact-aware retargeting of skinned motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9720–9729.
Yonatan Shafir, Guy Tevet, Roy Kapon, and Amit H Bermano. 2023. Human motion diffusion as a generative prior. *arXiv preprint arXiv:2303.01418* (2023).
Yijun Shen, Longzhi Yang, Edmond S. L. Ho, and Hubert P. H. Shum. 2020. Interaction-Based Human Activity Comparison. *IEEE Transactions on Visualization and Computer Graphics* 26, 8 (2020), 115673–115684. https://doi.org/10.1109/TVCG.2019.2893247
Seyoon Tak and Hyeong-Seok Ko. 2005. A physically-based motion retargeting filter. *ACM Trans. Graph.* 24, 1 (jan 2005), 98–117. https://doi.org/10.1145/1037957.1037963
Steve Tonneau, Rami Ali Al-Ashqar, Julien Pettré, Taku Komura, and Nicolas Mansard. 2016. Character contact re-positioning under large environment deformation. In *Computer Graphics Forum*, Vol. 35. Wiley Online Library, 127–138.
Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 6000–6010.
Ruben Villegas, Jimei Yang, Duygu Ceylan, and Honglak Lee. 2018. Neural Kinematic Networks for Unsupervised Motion Retargetting. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
Timo von Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. 2018. Recovering Accurate 3D Human Pose in The Wild Using IMUs and a Moving Camera. In *European Conference on Computer Vision (ECCV)*.
Jungdam Won and Jehee Lee. 2019. Learning Body Shape Variation in Physics-based Characters. *ACM Trans. Graph.* 38, 6, Article 207 (2019).
Haodong Zhang, ZhiKe Chen, Haocheng Xu, Lei Hao, Xiaofei Wu, Songcen Xu, Zhensong Zhang, Yue Wang, and Rong Xiong. 2024. Semantics-aware Motion Retargeting with Vision-Language Models. (2024).
Jiaxu Zhang, Junwu Weng, Di Kang, Fang Zhao, Shaoli Huang, Xuefei Zhe, Linchao Bao, Ying Shan, Jue Wang, and Zhigang Tu. 2023b. Skinned Motion Retargeting with Residual Perception of Motion Semantics & Geometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13864–13872.
Yunbo Zhang, Deepak Gopinath, Yuting Ye, Jessica Hodgins, Greg Turk, and Jungdam Won. 2023a. Simulation and Retargeting of Complex Multi-Character Interactions. In *ACM SIGGRAPH 2023 Conference Proceedings* (Los Angeles, CA, USA) *(SIGGRAPH '23)*. Association for Computing Machinery, New York, NY, USA, Article 65, 11 pages. https://doi.org/10.1145/3588432.3591491
Yi Zhou, Connelly Barnes, Lu Jingwan, Yang Jimei, and Li Hao. 2019. On the Continuity of Rotation Representations in Neural Networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

A DEFORMATION-BASED ADAPTATION

Following Al-Asqhar et al. [2013], we update as follow:

$$\vec{v}_{i,j} = p_j - d_i \tag{29}$$

$$w_{i,j} = \begin{cases} 0 & (r \le r_1) \\ 1 - (\frac{r_1}{r_2})^k & \text{if } (r_1 \le r < r_2) \\ 1 & \text{if } (r_2 \le r) \end{cases} \tag{30}$$

$$p'_j = d'_i + \sum_{i \in N_d} w_{i,j} * \vec{v}_{i,j} \tag{31}$$

where N_d is the number of descriptor points, i is the index of the descriptor, j is the index of joints to be updated h is the body height of the character to be updated.

B DATASET REFINEMENT

The source-target motion pairs obtained by procedural retargeting are used as a training dataset. However, because the proposed deformation-based adaptation works based on distances, unexpected updates on frames or body parts may occur. We alleviate this issue by allowing users to designate specific interaction ranges or body parts to be updated. As the source-target paired data is constructed by the user's preference, our SCT can train the desired motion of the user and generate the output motion desired as they want.