A Modular Neural Motion Retargeting System Decoupling Skeleton and Shape Perception

Jiaxu Zhang, Zhigang Tu, *Member, IEEE,* Junwu Weng, *Member, IEEE,* Junsong Yuan, *Fellow, IEEE,* Bo Du, *Senior Member, IEEE*

Abstract—Motion mapping between characters with different structures but corresponding to homeomorphic graphs, meanwhile preserving motion semantics and perceiving shape geometries, poses significant challenges in skinned motion retargeting. We propose M-R^{2}ET, a modular neural motion retargeting system to comprehensively address these challenges. The key insight driving M-R^{2}ET is its capacity to learn residual motion modifications within a canonical skeleton space. Specifically, a cross-structure alignment module is designed to learn joint correspondences among diverse skeletons, enabling motion copy and forming a reliable initial motion for semantics and geometry perception. Besides, two residual modification modules, *i.e.*, the skeleton-aware module and shape-aware module, preserving source motion semantics and perceiving target character geometries, effectively reduce interpenetration and contact-missing. Driven by our distance-based losses that explicitly model the semantics and geometry, these two modules learn residual motion modifications to the initial motion in a single inference without post-processing. To balance these two motion modifications, we further present a balancing gate to conduct linear interpolation between them. Extensive experiments on the public dataset Mixamo demonstrate that our M-R^{2}ET achieves the state-of-the-art performance, enabling cross-structure motion retargeting, and providing a good balance among the preservation of motion semantics as well as the attenuation of interpenetration and contact-missing.

Index Terms—motion retargeting, motion semantics learning, geometry perception, skeleton topology, self-supervision.

1 Introduction

Motion retargeting, as a process of mapping the motion of a source character to a target character without losing plausibility, has been a long-standing problem in the computer vision and computer graphics community. This technique has a wide range of applications in the game and animation industry, serving as a cornerstone for digital avatars and metaverse technologies [1].

Traditional methods utilize optimization-based techniques to refine the source motion with hand-designed and target-related constraints [2], [3], [4]. This process is time-consuming and requires professional knowledge to customize suitable constraints, hindering participation from non-experts. In recent years, learning-based retargeting methods started sparkling in the community. Among them, the neural motion retargeting [1], [5], [6], [7], [8], which has advantages in intelligent perception and stable inference, becomes a new research trend. The previous learning-based methods usually use a *full-motion mapping* structure, which decodes joint rotations of the target skeleton as outputs, with joint positions [1], [5], [7], [8] or joint rotations [6] as inputs. However, due to the gap between the Cartesian coordinate space and the rotation space, the full joint position encoding unavoidably introduces motion distortion.

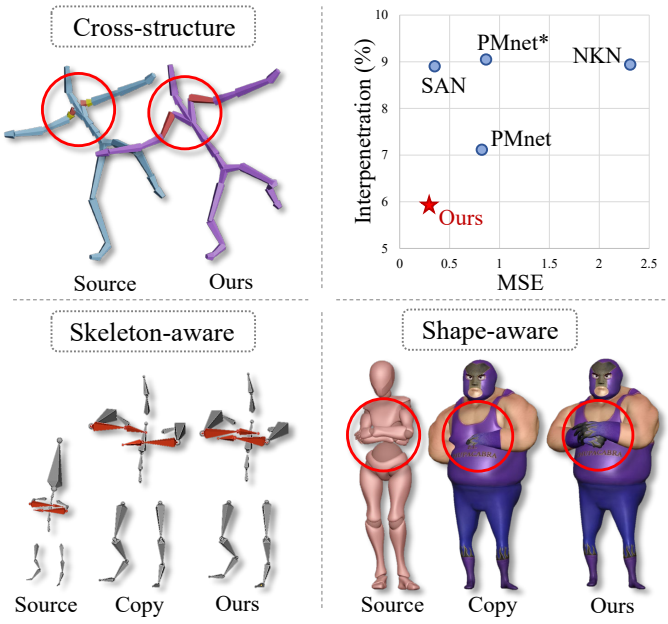


Fig. 1: Our M-R^{2}ET fully considers the source-target differences on the structure, skeleton, and shape geometry levels. The retargeted motion of M-R^{2}ET crosses skeleton structures, preserves motion semantics, eliminates interpenetration, and keeps self-contact without post-optimization.

Meanwhile, the full joint rotation encoding always leads to motion discontinuity in the rotation space [6], [9].

Moreover, the existing learning-based methods tend to

- *Jiaxu Zhang and Zhigang Tu are with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430072, China. Corresponding author: Zhigang Tu (Email: tuzhigang@whu.edu.cn).*
- *Junwu Weng is with Tencent AI Lab, Shenzhen 518063, China.*
- *Junsong Yuan is with the Computer Science and Engineering Department, University at Buffalo, Buffalo, NY 14228, USA.*
- *Bo Du is with the School of Computer Science, Wuhan University, Wuhan 430072, China.*

focus on addressing specific challenges in motion retargeting individually, such as cross-structure retargeting [6], motion semantics preserving [1], [5], [8], and geometry shapes perceiving [7]. However, they lack a unified and comprehensive system that can effectively tackle all three main challenges simultaneously. This limitation hinders their practical application and potential impact in the animation field. Furthermore, these off-the-shelf models cannot be easily merged to work together due to their reliance on different predefined skeleton structures and their inability to perceive geometric shapes end-to-end. Consequently, there is an urgent requirement to explore a comprehensive motion retargeting system that utilizes a unified insight to overcome these limitations.

In animation, we observe that artists normally first align the skeleton topology of the source and target characters, then copy the motion from source to target, and finally modify it manually to preserve motion semantics and avoid translation artifacts, *e.g.* interpenetration, during motion reuse in new characters. Motivated by this observation, different from the *full-motion mapping* structure, we design a *modular* neural motion retargeting system called M-R^{2}ET, which incorporates a *residual* retargeting structure. Our M-R^{2}ET is a comprehensive system that aligns the cross-structure skeletons to a canonical template, initializes the target skeleton with the source motion, and involves neural networks to imitate the modifications from artists. As illustrated in Figure 1, the explored system can perform the entire motion retargeting process in a single inference, encompassing cross-structural skeletal alignment as well as motion semantics and geometry perception. Moreover, in contrast to the *full-motion mapping* structure in previous methods, the proposed *residual* structure offers greater flexibility to our system. It also ensures fidelity and coherence in the results by reducing the search space for retargeting solutions during training. Furthermore, our M-R^{2}ET exhibits robust generalization capabilities, producing plausible results even for unseen characters without retraining the model.

The key insight of our M-R^{2}ET is to align various biped characters into a canonical skeleton space, and utilize residual neural models to cope with two main differences between the source and target characters: 1) the differences in bone length ratio; 2) the differences in body shape geometry. To reach this goal, we explore three modules, *i.e.*, the cross-structure alignment module, the skeleton-aware module, and the shape-aware module. These three modules are decoupled and flexible but work in harmony to achieve physically plausible motion retargeting. It is important to note that the characters discussed in this paper are pre-rigged, which allowing their mesh to be driven with the skeleton motion through the linear blend skinning strategy.

For the source and target characters with various skeleton structures, the *cross-structure alignment* module standardizes them into a canonical skeleton template, thereby copying the motion from the source skeleton to the target skeleton. The canonical skeleton space and the copied initial motion serve as a fundamental pre-step to the perception of both motion semantics and geometry. The alignment process is accomplished by classifying the skeleton joints based on the predefined joint categories on the canonical skeleton template. As a result, we can copy the motion of the same-class joints from the source skeleton to the standardized target skeleton. To handle cases that with mismatched joint numbers, we employ rotation interpolation or multiplication techniques as necessary.

On the skeleton level, the *skeleton-aware* module takes the skeleton configurations as input to assist the transfer of the source motion semantics, such as arm folding and hand clapping, to the target character. To overcome the problem of lacking paired and semantics-correct ground truth, we directly take the supervision signal from the input source motion. The motion semantics is explicitly modeled as a normalized Distance Matrix (DM) of the skeleton joints. Accordingly, the semantics preservation is achieved by aligning the DM between the source and target motions.

On the shape geometry level, the *shape-aware* module senses the compatibility between the skeleton which is adjusted after motion semantics preservation and the target character mesh to avoid interpenetration and contact-missing. To train the module end-to-end, we introduce two voxelized Distance Fields, *i.e.*, the Repulsive Distance Field (RDF) and the Attractive Distance Field (ADF), as measurement tools for interpenetration and contact. We sample the distance of the query vertices on the target character mesh to the body surface in these two fields to estimate the degree of interpenetration and contact. With this design, the whole process is differentiable during training.

In practice, we find there always exists a contradiction between the preservation of motion semantics and the avoidance of interpenetration. We, therefore, propose a *balancing gate* to make a trade-off between the skeleton-level and the geometry-level modifications through learning an adjusting weight. By leaving the weight to the user, our M-R^{2}ET system also accepts interactive fine-grained control from users.

With the above main designs, our M-R^{2}ET enables cross-structure skinned motion retargeting, preserving the motion semantics of the source character, and avoiding interpenetration and contact-missing issues in a single-pass without post-processing. We evaluate our method on various complex motion sequences, different skeleton typologies, and a wide range of character geometries from skinny to bulky. The qualitative and quantitative results show that our M-R^{2}ET outperforms the existing learning-based methods by a large margin. The advantages of our proposed method are summarized in three-fold:

- A novel residual network structure is explored for neural motion retargeting, which involves a skeleton-aware modification module, a shape-aware modification module, and a balancing gate.
- A normalized joint Distance Matrix is presented to guide the training of the skeleton-aware module for explicit motion semantics modeling, and two voxelizd Distance Fields are introduced to achieve differentiable pose adjustment learning.
- Extensive experiments on the Mixamo [10] dataset demonstrate that our method achieves the state-of-the-art performance qualitatively and quantitatively.

This work is an extension of our conference paper [11]. The new contributions include:

- We extend our R^{2}ET to the M-R^{2}ET system, which further exploits the cross-structure retargeting and con-

structs a modular neural retargeting system to comprehensively tackle the pivotal challenges within a canonical skeleton space.
- A cross-structure alignment module is designed to align the homeomorphic skeletons with a canonical template and copy motion among the aligned skeletons, which enables our system to learn semantics and geometry features better in a canonical skeleton space.
- More experiments on cross-structure retargeting data and video motion capture data are conducted to verify the generalization and practicality of our proposed M-R^{2}ET system.

The rest of this paper is organized as follows. In Section 2, we introduce the related works. In Section 3, we provide an overview of our M-R^{2}ET system. In Section 4 and Section 5, we explain the proposed cross-structure alignment module as well as the semantics and shape perception method in detail. In Section 6, we show the ablation studies and the comparisons with state-of-the-arts, where the qualitative and quantitative results are reported comprehensively. Finally, the paper is concluded in Section 7, and the limitations of the proposed system are also discussed.

2 Related Work

Motion Retargeting. Which is pioneered by [12], aims to identify features of the source motion as kinematic constraints and solve the space-time optimization problem. Following [12], many optimization-based motion retargeting methods were proposed successively by introducing specific constraints, e.g., dynamics constraints [3], [13], inverse kinematics [14], [15], joint angle constraints [16], [17], Euclidean distance [18], and trajectory constraints [19]. These traditional methods are widely used in the field of animation, but they are not easy to implement and require professional knowledge from experts.

Recently, there has been a surge of interest in studying deep-learning-based motion retargeting, which is fully data-driven and thus accessible to a wider audience. Jang *et al*. [20] used a deep autoencoder combining the DC-IGN [21] and the U-Net [22] to generate human motions. Villegas *et al*. [1] trained a Neural Kinematic Network for unsupervised motion retargeting. Lim *et al*. [5] developed a novel architecture which separately learns frame-by-frame poses and overall movement. Li *et al*. [8] introduced an iterative method to yield retargeted motions based on a trained motion autoencoder. All these methods were performed on single articulated skeletons while ignoring the skeleton typologies and the shape geometry of characters. Thus, they cannot handle cross-structure retargeting and are prone to produce unrealistic results on skinned motions.

On the other hand, Aberman *et al*. [6] proposed a Skeleton-Aware Network (SAN) for retargeting motion between skeletons with different topologies. They exploited a pooling operation on skeletons to simplify them into a basic graph via an observation that the biped characters' skeletons are usually homeomorphic. However, because of the network is full-mapping structure, SAN needs to be retrained if the skeleton topology changes. Furthermore, SAN also ignores the shape geometry of characters. Villegas *et al*. [7] presented a latent-space optimization method for skinned motions to preserve self-contacts and prevent interpenetration, but this post-processing method is cumbersome and is unsuitable for a stable real-time system.

In contrast to the above-mentioned methods, we present a modular system called M-R^{2}ET, designed for both skeletal and skinned motion retargeting, considering motion semantics and character shapes. Our M-R^{2}ET system can operate in an end-to-end fashion and effectively handle homeomorphic skeletons without the need for retraining.

Geometry-aware Motion Modeling. There is tremendous work on learning geometry-aware deep motion representations [23], [24], [25]. Gomes *et al*. [26] leveraged the human body shape in the retargeting process while considering the physical constraints of the motion in the 2D image domain. Peng *et al*. [27] introduced the neural blend weight fields to reconstruct an animatable human model from a multi-view video. Jiang *et al*. [28] combined explicit and implicit representations to recover spatio-temporal coherent geometries from a monocular human video. These works extract human motion from RGB-based videos or images, while our work focuses on the motion retargeting of humanoid characters in the 3D space.

In the 3D vision field, Jin *et al*. [29] designed a volumetric mesh that surrounds a character's skin to preserve the spatial relationships of humans. Basset *et al*. [30] exploited an optimization-based method to deform the source shape in the desired pose using three energy functions. Liao *et al*. presented a skeleton-free pose transfer model to automatically transfer poses between stylized 3D characters with consideration of their mesh geometry. Peng *et al*. [31] introduced a cage-based representation as deformation prior to deform and animate the implicit field of arbitrary objects.

In this work, we construct two neural residual modules with distance-based losses to learn the motion semantics and the character geometrics for motion retargeting. Our method mainly copes with the animated characters with various articulated skeletons, but can be also extended to retarget the motion from Skinned Multi-Person Linear Model (SMPL) [32] estimated from RGB videos.

Node Classification. Node classification is a fundamental task in the field of graph representation learning and plays a crucial role in various real-world applications, e.g., social network analysis [33], [34], recommendation systems [35], [36], and bioinformatics [37], [38]. In this work, we treat the alignment of various skeleton joints to a canonical skeleton template as a node classification problem of dynamic graphs. The goal of node classification is to predict the label or class of each node in a given graph based on its structural and relational information with other nodes. One of the key challenges in this area is dealing with the irregular and non-Euclidean nature of graphs, as traditional neural networks are designed for regular grid-like data [37]. To tackle this, researchers have proposed several innovative approaches, which can be broadly categorized into three main paradigms: graph convolutional networks (GCNs) [39], graph attention networks (GATs) [40], and transformer-based methods [41].

Graph convolutional networks, introduced by Kipf *et al*. [39], have been one of the pioneering techniques for graph node classification. Later, Hamilton *et al*. [42] proposed GraphSAGE that utilizes a sampling strategy to generate

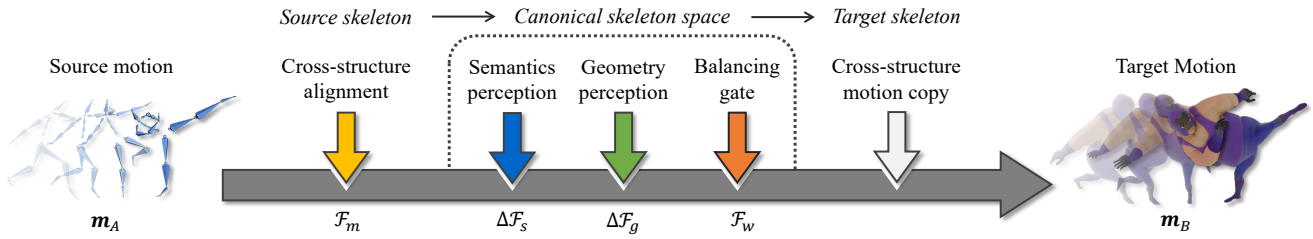


Fig. 2: Overview of the proposed M-R^{2}ET system, which has three decoupled modules, *i.e.*, the cross-structure alignment module $\mathcal{F}_m$, the skeleton-aware module $\Delta\mathcal{F}_s$, the shape-aware module $\Delta\mathcal{F}_g$, and an extra balancing gate $\mathcal{F}_w$. Each of these modules can be seamlessly and flexibly embedded in the system as needed, enabling end-to-end motion retargeting.

node representations, allowing it to scale to larger graphs effectively. Defferrard *et al*. [43] presented a spectral-based graph convolutional network called ChebNet, which contains a fast localized spectral graph filter derived from Chebyshev polynomial. GIN introduced in [44] operates on fixed-size embeddings and incorporates message passing with pooling operations, allowing it to capture global graph properties.

In M-R^{2}ET, we leverage a GCN-based cross-structure alignment module to classify the skeleton joints into predefined categories of a canonical template. This process enables us to copy motion from the source skeleton to a standardized target skeleton, facilitating seamless motion retargeting between different skeletal typologies.

3 Overview

In this section, we will first provide an overview of two key preliminaries used in M-R^{2}ET, *i.e.*, GCN and Transformer, respectively. Then, we will present the overview framework of our M-R^{2}ET system and introduce the notations used throughout this paper.

3.1 Preliminaries

We use GCN in our cross-structure alignment module. According to [39], the layer-wise feature propagation rule in GCN is formulated as:

$$H^{l+1} = \sigma(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}H^l W^l), \tag{1}$$

where H is the graph feature. $\widetilde{A} = A + I_N$ is the adjacency matrix of the graph. I_N is the identity matrix with $N \times N$ dimension. N is the number of the graph nodes. $\widetilde{D}_{i,i} = \sum_j \widetilde{A}_{i,j}$ and W^l is the layer-specific trainable weights.

We use the self-attention mechanism in our shape-aware module. The self-attention function in Transformer exploited in [45] can be described as mapping a query and a set of key-value pairs to an output, where the query (Q), key (K), value (V), and output are all feature vectors. The output is computed as a weighted sum of V, where the weight assigned to each V is calculated by a correlation function of Q with the corresponding K. In practice, the self-attention function is defined as:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V, \tag{2}$$

where d_k is the dimension of K. Integrating multiple self-attention heads, the multi-head self-attention can be formulated as:

$$Multihead(Q, K, V) = Concat(head_1, ..., head_h)W^O, \tag{3}$$

where $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$. W represents the trainable weights.

3.2 Proposed System: M-R^{2}ET

The overview of our system is illustrated in Figure 2. Given the motion sequence $\boldsymbol{m}$ of the input character A, our system mainly employs three sequential modules, complemented by an additional balancing gate, to retarget the source motion $\boldsymbol{m}_A$ to the target character B, and obtaining the output $\boldsymbol{m}_B$. The core step in this pipeline involves using the cross-structure alignment module $\mathcal{F}_m$ to harmonize the source skeleton $\boldsymbol{\gamma}_A$ and the target skeleton $\boldsymbol{\gamma}_B$ into the canonical skeleton space γ^T. Subsequently, the skeleton-aware module $\Delta\mathcal{F}_s$ and the shape-aware module $\Delta\mathcal{F}_g$ are utilized to perceive the motion semantics and the geometry, respectively. Following this, the balancing gate $\mathcal{F}_w$ is applied to strike a balance between the semantics and geometry. Finally, a cross-structure motion copy approach is explored to transfer the canonical skeleton to the target skeleton, completing the motion retargeting process.

The motion sequence $\boldsymbol{m}$ used in our pipeline involves the global root motion, *i.e.*, velocities and rotations $\{\boldsymbol{v}^t\}_{t=1}^T$, $\boldsymbol{v} \in \mathbb{R}^4$, and the local joint rotation quaternions $\{\boldsymbol{q}^t\}_{t=1}^T$, $\boldsymbol{q} \in \mathbb{R}^{N\times 4}$. N and T indicate the number of joints and the sequence length. The time-index t is ignored in the following for simplification. The global root movement is simply processed by normalizing and denormalizing with respect to the heights of the source and target characters. The local joint rotation is translated framewisely, which will be introduced in detail in the following sections. In contrast to [1], [5] which take joint positions as input, we focus on retargeting the motions in the rotation space as [6].

4 Cross-structure Alignment

A unified skeleton typology serves as a fundamental prerequisite to the perception of both motion semantics and geometry in our M-R^{2}ET. However, the skeletal joint number and structure differ among various characters. For instance, in the case of a zombie character, artists may incorporate additional spine or shoulder joints to facilitate stylized motion. Consequently, conducting motion copy on these

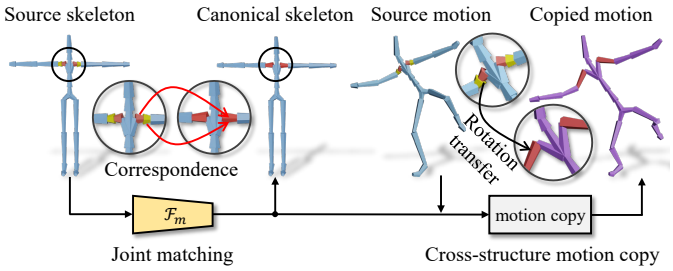
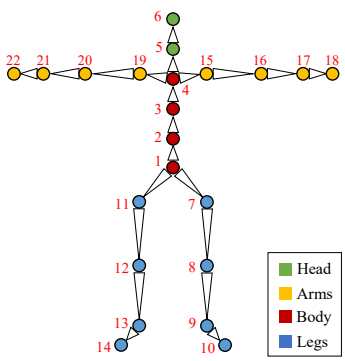


Fig. 3: Illustration of the cross-structure alignment module.



1	Root	12	RightLeg
2	Spine	13	RightFoot
3	Spine1	14	RightToeBase
4	Spine2	15	LeftShoulder
5	Neck	16	LeftArm
6	Head	17	LeftForeArm
7	LeftUpLeg	18	LeftHand
8	LeftLeg	19	RightShoulder
9	LeftFoot	20	RightArm
10	LeftToeBase	21	RightForeArm
11	RightUpLeg	22	RightHand

Fig. 4: Illustration of the canonical skeleton template and the joint categories.

disparate skeletons becomes challenging. To address the issue of heterogeneity in skeleton structures, we develop a straightforward yet effective cross-structure alignment module. This module leverages a GCN to learn the correspondence among diverse skeletons with a canonical topology template, then the skeleton joints can be aligned and processed in a unified manner. Additionally, we introduce a cross-structure motion copy technique to transfer the rotations on the matched joints from source skeletons to target skeletons. Following, we will investigate the inner working mechanism and elaborate the functionality of the cross-structure alignment module in-depth.

4.1 Joint Matching

The establishing of a standardized skeletal topology forms the bedrock of perceiving motion semantics and geometry within our M-R^{2}ET system. Hence, we present a joint matching approach that harmonizes skeletons of diverse humanoid characters with a canonical template. In Figure 4, this template comprises 22 joints, acting as the primary controllers for executing motions in a humanoid character. The joints naturally shape a graph structure based on their physical connections. Taking inspiration from the graph node classification technique [46], we treat the 22 joints of the skeleton template as distinct bone categories and leverage a GCN-based joint matching module, denoted as $\mathcal{F}m$, to classify various skeleton joints into these predefined categories. As depicted in the left part of Figure 3, the joint matching module $\mathcal{F}m$ takes the source or target skeleton, which may deviate from the template, as input and assigns each joint to a specific category. With the joint matching

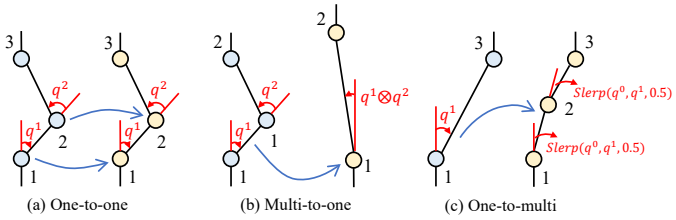


Fig. 5: The three cases in our cross-structure motion copy strategy.

Algorithm 1 Generate training skeleton samples

Input: The skeleton template γ^T
Output: A training skeleton sample $\gamma^{'}$
initial $\gamma^{'} = \gamma^T$;
for j in joints; except Root joint; **do**
  Randomly scale j's length to 0.5 - 2.0 times;
  if j in arms or legs **then**
    Operate symmetrically on the other side;
  end if
end for
for g in groups; **do**
  Randomly split a joint into two
  or merge two joints into one;
  if g is arms or legs **then**
    Operate symmetrically on the other side;
  end if
end for

module, we can align the joints of any skeleton, regardless of their joint number and structure, with the bone categories of the template. As a result, seamless motion copy between these homeomorphic skeletons becomes feasible.

The training procedure of the joint matching module does not necessitate real skeleton data from various characters and manually annotated joint labels. Instead, all the training data can be automatically generated from the skeleton template. To achieve this, we divide the 22 joints of the skeleton template into four distinct groups, as depicted in Figure 4, representing the head, arms, body, and legs, respectively. Subsequently, the training skeleton samples are generated by using Algorithm 1. The GCN-based joint matching module is optimized via the Cross-Entropy loss:

$$\mathcal{L}_{ce}(\boldsymbol{p}, \hat{\boldsymbol{p}}) = -\frac{1}{NC}\sum_{i=1}^{N}\sum_{c=1}^{C} p_{ic} log(\hat{p}_{ic}), \tag{4}$$

where p_{ic} and $\hat{p}_{ic}$ are the label and the predicted label of joint i on class c, respectively.

4.2 Cross-structure Motion Copy

Once we have aligned the structurally diverse skeleton with the canonical skeleton template, we can create a proxy skeleton for the target character by adjusting the original target skeleton to match the template's topology meanwhile preserving the relative length of the bones. This process involves merging the skeleton joints that share the same class to eliminate redundancy and adding missing joints between their parent and child joints. Subsequently, we

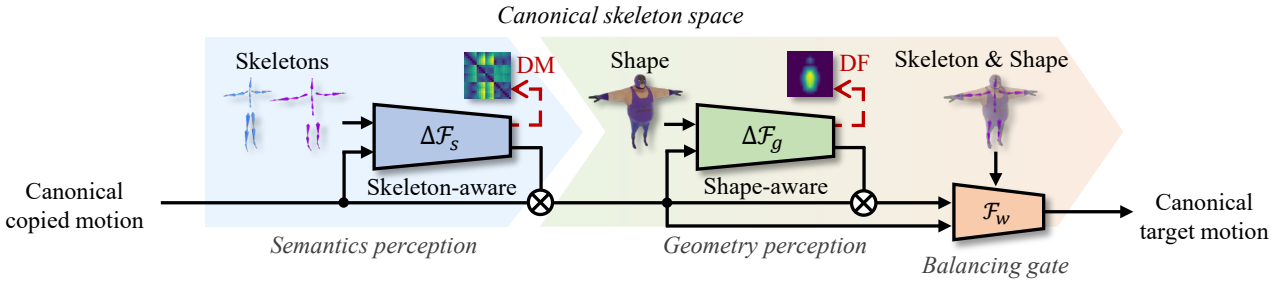


Fig. 6: Overview of the semantics and geometry perception in the canonical skeleton space. The skeleton-aware module, shape-aware module, and balancing gate are all operated in the canonical skeleton space.

transfer the motion from the source skeleton to the proxy skeleton through a cross-structure motion copy strategy, illustrated in Figure 5, which comprises three cases: (a) one-to-one, (b) multi-to-one, and (c) one-to-multi.

In the one-to-one case, we copy the joint rotation from the original skeleton to the proxy skeleton directly, based on their joint correspondence. For the multi-to-one scenario, we combine the rotations of multiple joints onto their corresponding joint using the Hamilton Product of quaternions. In the one-to-multi situation, we utilize Spherical Linear Interpolation (Slerp) to evenly distribute the rotation of one joint to multiple corresponding joints. The Slerp process can be formulated as:

$$Slerp(\boldsymbol{q}^0, \boldsymbol{q}^j, 1/K) = \frac{sin(\beta - \frac{\beta}{K})\boldsymbol{q}^0 + sin(\frac{\beta}{K})\boldsymbol{q}^j}{sin(\beta)}, \quad (5)$$

where $\boldsymbol{q}^0 = (1, 0, 0, 0)$ and β can be calculated by the dot product $\boldsymbol{q}^0 \cdot \boldsymbol{q}^j = cos(\beta)$. K is the number of the corresponding joints. Indeed, we have noticed that humanoid characters generally share similar skeletal topologies, with variations mainly occurring in the number of joints around the spine or shoulder regions, typically involving minor rotations. Thus, our cross-structure motion copy strategy enables seamless motion transfer between the source skeleton and the proxy skeleton meanwhile preserving the characteristics of the motion. As illustrated in Figure 2, after the residual modification of semantics and geometry, we extra use this motion copy strategy to transfer the motion in the canonical skeleton space to the real target skeleton.

5 Semantics & Geometry Perception

Inspired by the creation process in animation, we design a residual modification structure to automatically achieve skeleton-aware and shape-aware local motion retargeting. In particular, our M-R^{2}ET takes the copied motion $\boldsymbol{q}_T$ in the canonical skeleton space as an initialization. A skeleton-aware modification module $\Delta\boldsymbol{q}_s = \Delta\mathcal{F}_s(\cdot)$ is introduced to maintain motion semantics, and a shape-aware modification module $\Delta\boldsymbol{q}_g = \Delta\mathcal{F}_g(\cdot)$ is involved to tackle the interpenetration and contact-missing issues. A balancing gate $\mathcal{F}_w$ is located at the end of this pipeline to balance the two motion modifications. The whole process is then formulated as:

$$\boldsymbol{q}_B = \mathcal{F}_w(\, \boldsymbol{q}_T,\ \Delta\boldsymbol{q}_s,\ \Delta\boldsymbol{q}_g\,). \quad (6)$$

Base Losses. One of the challenges in the neural motion retargeting is that there is always no paired ground truth as target motion supervision. Following [5], we utilize the self-reconstruction principle and adversarial learning as the basic training rules to train $\Delta\mathcal{F}_s$, $\Delta\mathcal{F}_g$ and $\mathcal{F}_w$ in an unsupervised way.

During training, the self-reconstruction regularization is conducted in two ways: 1) reconstructing the exact source motion in the source character (see Section 5.1), 2) minimizing the pose modifications in pose adjustment (see Section 5.2). To avoid the quaternion ambiguity and reduce the position error accumulation along the kinematic chain, the joint rotations and positions are reconstructed simultaneously. Accordingly, the reconstruction loss is defined as:

$$\mathcal{L}_{rec}(\boldsymbol{q}, \hat{\boldsymbol{q}}) = \left\|\boldsymbol{q} - \hat{\boldsymbol{q}}\right\|_2^2 + \left\|f_K(\boldsymbol{q}, \boldsymbol{\gamma}^T) - f_K(\hat{\boldsymbol{q}}, \boldsymbol{\gamma}^T)\right\|_2^2, \quad (7)$$

where $\boldsymbol{q}$ is the input rotation and $\hat{\boldsymbol{q}}$ is the estimated one. f_K denotes a Forward Kinematics (FK) layer [1] that maps the local joint rotations to the global joint positions by referring the rest-pose configuration $\boldsymbol{\gamma}^T \in \mathbb{R}^{N\times 3}$ in the canonical skeleton space.

To achieve realistic motion retargeting, a discriminator $\delta(\cdot)$ is introduced to differentiate the translated motion sequences from the genuine ones. A motion discrimination loss is designed based on the adversarial training [47]:

$$\begin{aligned}\mathcal{L}_{adv}(\hat{\boldsymbol{q}}) = \mathbb{E}_{\boldsymbol{m}\sim p_{real}}\big[\log \delta(\boldsymbol{m})\big] + \\ \mathbb{E}_{\boldsymbol{m}\sim p(\hat{\boldsymbol{q}})}\big[\log\big(1 - \delta(\boldsymbol{m})\big)\big],\end{aligned} \quad (8)$$

in which, $p(\cdot)$ represents the distribution of the *real* motions or *fake* retargeted motions controlled by $\hat{\boldsymbol{q}}$.

Besides, a rotation constraint loss [1] is introduced to constrain the y-axis Euler angles within a range, avoiding excessive joint twisting:

$$\mathcal{L}_{rot}(\hat{\boldsymbol{q}}) = \left\| \max\left(\boldsymbol{0},\ |\epsilon_y(\hat{\boldsymbol{q}})| - \alpha\right)\right\|_2^2. \quad (9)$$

The function $\epsilon_y(\cdot)$ converts the input quaternion to the Euler angle of y-axis, and α is the angle limitation bound. $\max(\cdot)$ is an element-wise function that returns the maximum number between the two inputs.

With these definitions, the base loss is thereafter defined by weighted summarizing the above losses as follows:

$$\mathcal{L}_{base}(\boldsymbol{q}, \hat{\boldsymbol{q}}) = \mathcal{L}_{rec} + \lambda\mathcal{L}_{adv} + \mu\mathcal{L}_{rot}, \quad (10)$$

where λ and μ are the loss balancing factors.

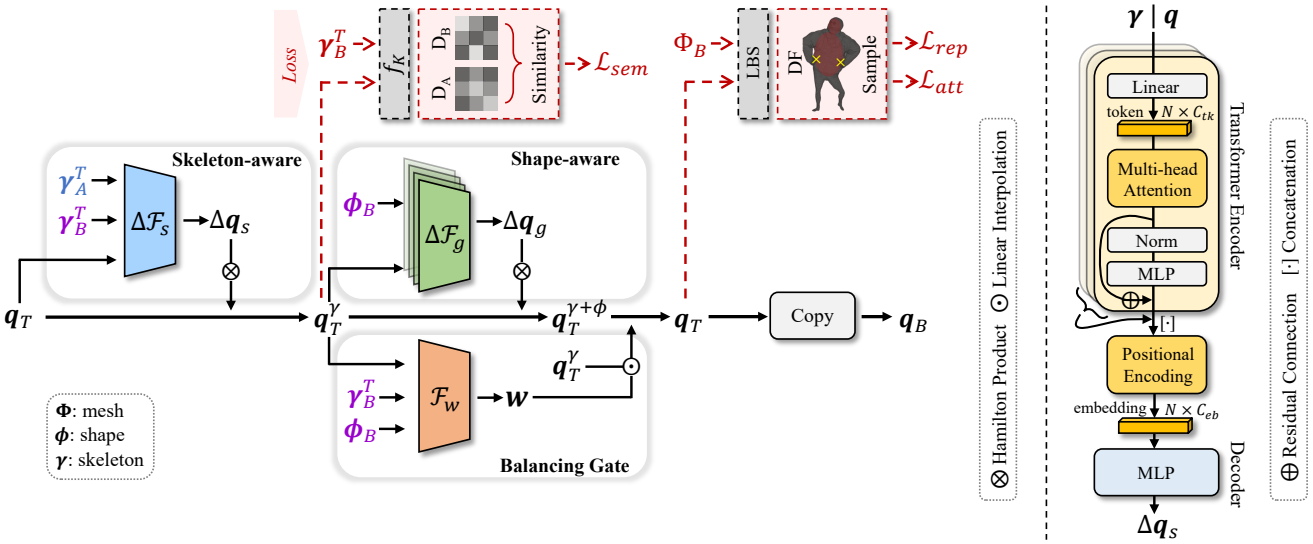


Fig. 7: The detailed description of the proposed semantics and geometry residual modification pipeline and the transformer-based skeleton-aware module (the right one).

5.1 Skeleton-aware Residual Adaptation

The motion residual design can help maintain the motion coherence from the source character and meanwhile provides a good initialization for motion translation. However, owing to the differences of bone length and skeleton proportion between the source and target characters, motion copy $\boldsymbol{q}_T = \boldsymbol{q}_A$ may ignore the source motion semantics. In this section, we will introduce the skeleton-aware module $\Delta\mathcal{F}_s$ for motion semantics preservation.

The skeleton-aware module takes the rest-pose skeletons $\boldsymbol{\gamma}_A^T$ and $\boldsymbol{\gamma}_B^T$ as well as the copied motion $\boldsymbol{q}_T$ as input, and outputs the semantics-oriented quaternion modification $\Delta\boldsymbol{q}_s \in \mathbb{R}^{N\times 4}$. With this estimated motion modification, a Hamilton Product is applied to modify the motion copy $\boldsymbol{q}_T$ and thereafter to preserve the motion semantics in the output $\boldsymbol{q}_T^\gamma$, namely:

$$\begin{aligned}\boldsymbol{q}_T^\gamma &= \Delta\boldsymbol{q}_s \otimes \boldsymbol{q}_T \\ &= \Delta\mathcal{F}_s(\boldsymbol{\gamma}_A^T, \boldsymbol{\gamma}_B^T, \boldsymbol{q}_T;\ \boldsymbol{\theta}_s) \otimes \boldsymbol{q}_T,\end{aligned} \tag{11}$$

where θ_s represents the parameter of module $\Delta\mathcal{F}_s$.

Semantics Preservation. There is no paired ground truth as strong semantics supervision. By utilizing the property of motion retargeting, we here take the supervision signals from the source motion. We model the motion semantics preservation as the maintaining of the source normalized pair-wise joint distances in the target character pose, frame by frame. We introduce a normalized Distance Matrix (DM) $\boldsymbol{D} \in \mathbb{R}^{N\times N}$ to represent the motion semantics, *i.e.*, the pair-wised joint distances as shown in Figure 8. The columns of the matrix indicate the query joints, and the rows represent the reference joints. The element $d_{i,j}$ of $\boldsymbol{D}$ denotes the Euclidean distance from the query joint i to the reference joint j. We extract the pose DM from the source character and regard it as a supervision signal to guide the learning of the target pose DM. With this design, a Semantics Similarity loss is then defined as:

$$\mathcal{L}_{sem} = \left\| \eta\left(\frac{\boldsymbol{D}_A}{h_A}\right) - \eta\left(\frac{\boldsymbol{D}_B}{h_B}\right) \right\|_2^2, \tag{12}$$

where h is the height of the skeleton. $\eta(\cdot)$ is an $L1$ normalization performed on each row of the distance matrix. This normalization operation eliminates the difference of bone lengths and heights between the source and target skeletons.

To support the pair-wise joint relationship learning, we introduce a Transformer structure [45], whose attention mechanism is suitable for pair-wise learning, to build the skeleton-aware module. As shown in the right of Figure 7, our Transformer-based structure consists of two Transformer encoders and one MLP decoder. The Transformer encoders process $\boldsymbol{\gamma}$ and $\boldsymbol{q}$ independently. In this process, N joint features are treated as N tokens with C_{tk} channels, and they are encoded by a Multi-head Attention and a Layer Normalization. Then, the feature of $\boldsymbol{\gamma}$ and $\boldsymbol{q}$ are concatenated and position-encoded to obtain an embedding with C_{eb} channels. In the end, a MLP is shared within N joints to decode the rotation modifications $\Delta\boldsymbol{q}_s$ for these joints. With the Semantics Similarity loss and the Transformer-based model structure introduced above, the skeleton-aware module can be trained by:

$$\min_{\boldsymbol{\theta}_s}\ \mathcal{L}_{base}(\boldsymbol{q}_T, \boldsymbol{q}_T^\gamma) + \nu\mathcal{L}_{sem}, \tag{13}$$

where ν is the loss balancing factor. The reconstruction loss is applied when the source and the target characters are the same. We sample the target character as the source one with a probability of 0.5.

5.2 Shape-aware Residual Adaptation

We introduce a shape-aware module $\Delta\mathcal{F}_g$ in this section to ensure the retargeted skinned motion is interpenetration-free and contact-preserved, as illustrated in the middle part of Figure 7. The shape-aware module takes the shape

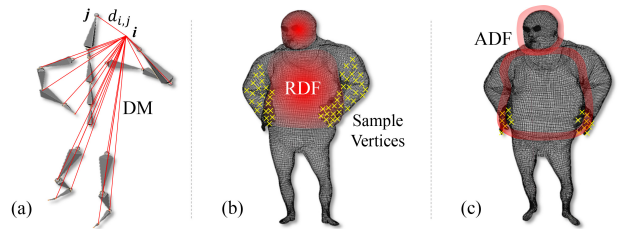


Fig. 8: Illustration of the distance measurements. DM is the normalized Distance Matrix of the skeleton joints. RDF and ADF are the Distance Fields inside and outside the main body.

information $\boldsymbol{\phi}$ of each body part in the target character as well as the $\Delta\mathcal{F}_s$-modified local joint rotation $\boldsymbol{q}_T^{\gamma}$ as input, and outputs the geometry-oriented quaternion modification $\Delta\boldsymbol{q}_g \in \mathbb{R}^{N\times 4}$. $\boldsymbol{\phi} \in \mathbb{R}^{N\times 3}$ is represented by the edge-lengths of the body part bounding box corresponding to each joint in the rest-pose. As we observed, most of the interpenetration and contact-missing issues occur between the limbs and the main body. We, therefore, choose to only adjust the rotations of four target limbs and introduce four MLPs to estimate the rotation modifications for them, independently. With the estimated rotation modifications, the adjusted joint rotation $\boldsymbol{q}_T^{\gamma+\phi}$ is defined as:

$$\begin{aligned}\boldsymbol{q}_T^{\gamma+\phi} &= \Delta\boldsymbol{q}_g \otimes \boldsymbol{q}_T^{\gamma} \\ &= \Delta\mathcal{F}_g(\boldsymbol{\phi}_B, \boldsymbol{q}_T^{\gamma};\ \boldsymbol{\theta}_g) \otimes \boldsymbol{q}_T^{\gamma},\end{aligned} \tag{14}$$

where θ_g represents the parameter of module $\Delta\mathcal{F}_g$.

Penetration-free & Contact-preserving. To achieve differentiable pose adjustment learning with respect to mesh geometry, we introduce two truncated distance fields, *i.e.*, the Repulsive Distance Field (RDF) ψ_R inside the main body and the Attractive Distance Field (ADF) ψ_A around the body surface. These two fields are illustrated in Figure 8 (b, c). RDF assists the network to force the penetrating vertices to be apart from the interpenetration area, and the ADF attracts the near-contact vertices adjusted after motion semantics preservation to be close to the body surface. With the adjusted rotation $\boldsymbol{q}_T^{\gamma+\phi}$, we first deform the mesh vertex set of the target character Φ_B by utilizing Linear Blend Skinning (LBS). Hereafter, ψ_R and ψ_A are estimated by voxelizing the deformed target mesh. Each node on the voxel grid records its distance to the body surface from inside or outside, and we can thus measure the body-surface deviation of each query vertice e on the deformed target mesh by interpolating four node distances on its surrounding tight voxel grid. With this mechanism, our model can be trained to handle the interpenetration and contact-missing problem in an end-to-end manner, and translate a source motion to a plausible target motion during inference without post-processing. These two fields are embedded into two losses, *i.e.*, the Repulsive loss and the Attractive loss, to achieve end-to-end training, and they are defined as:

$$\mathcal{L}_{rep} = \frac{1}{N_l}\sum_{e\in E_l}\psi_R(e), \quad \mathcal{L}_{att} = \frac{1}{N_h}\sum_{e\in E_h}\psi_A(e), \tag{15}$$

where $E_l = \{e_i\}_{i=1}^{N_l}$ and $E_h = \{e_i\}_{i=1}^{N_h}$ are the vertices set of the deformed target mesh's limbs and hands, respectively.

TABLE 1: Detailed architectures of the shape-aware module $\Delta\mathcal{F}_g$ and the balancing gate $\mathcal{F}_w$. The keep probability of the Dropout layers is set as 0.8.

Name	Layer	Channels	Activation
$\Delta\mathcal{F}_g$	Linear	$154 \rightarrow 256$	ReLU
	Dropout	-	-
	Linear	$256 \rightarrow 256$	ReLU
	Dropout	-	-
	QLinear	$256 \rightarrow 88$	-
$\mathcal{F}_w$	Linear	$220 \rightarrow 512$	ReLU
	Dropout	-	-
	Linear	$512 \rightarrow 512$	ReLU
	Dropout	-	-
	Linear	$512 \rightarrow 256$	ReLU
	Dropout	-	-
	Linear	$256 \rightarrow 22$	Sigmoid

N_l and N_h are the corresponding numbers of vertices. $\psi(e)$ samples the ψ value for each vertex e in a differentiable way. $\Delta\mathcal{F}_g$ consists of four independent networks corresponding to the four limbs of the character. Accordingly, these four networks are optimized by four related $\mathcal{L}_{rep}^{E_l}$ as:

$$\min_{\boldsymbol{\theta}_g}\ \mathcal{L}_{base}(\boldsymbol{q}_T^{\gamma}, \boldsymbol{q}_T^{\gamma+\phi};\ \boldsymbol{\theta}_g) + \kappa\sum_{i=1}^{4}\mathcal{L}_{rep}^{E_l^i}(\ \cdot\ ;\ \boldsymbol{\theta}_g^i), \tag{16}$$

where κ is the balancing hyper-parameter, $\boldsymbol{\theta}_g = [\boldsymbol{\theta}_g^i]_{i=1}^4$. As repulsing and attracting the mesh vertices simultaneously would cause unstable training convergence, we do not involve $\mathcal{L}_{att}$ here but leave it in the next Balancing module.

5.3 Balancing Gate

In practice, it is challenging to learn the motion semantics preservation at the skeleton level and meanwhile train the network to tackle the issues of interpenetration as well as contact-missing at the shape-geometry level. Let's take the bottom of Figure 1 as an instance. When retargeting motion from a thin character to an obese character, if only the relative positions of the joints are maintained, it will inevitably lead to interpenetration. On the other side, if only the target shape is considered, the retargeted motion may lose motion semantics. To overcome this problem, we introduce an additional MLP module $\mathcal{F}_w$ to balance the influence between the two modifications $\Delta\boldsymbol{q}_s$ and $\Delta\boldsymbol{q}_g$ by a learned balancing factor $\boldsymbol{w} \in \mathbb{R}^N$. This balancing process is achieved by a linear interpolation between q_T^{γ} and $q_T^{\gamma+\phi}$:

$$\boldsymbol{q}_T = (\mathbf{1} - \boldsymbol{w}) \cdot \boldsymbol{q}_T^{\gamma} + \boldsymbol{w} \cdot \boldsymbol{q}_T^{\gamma+\phi}. \tag{17}$$

in which, $\boldsymbol{w} = \mathcal{F}_w(\boldsymbol{\gamma}_T, \boldsymbol{\phi}_B, \boldsymbol{q}_T^{\gamma}; \theta_w)$, θ_w indicates the parameters of $\mathcal{F}_w$. Each element of $w \in \boldsymbol{w}$ is ranged from 0 to 1, and the symbol "$\cdot$" indicates scaling each row of $\boldsymbol{q}$ via an element of $\boldsymbol{w}$. By leaving the vector $\boldsymbol{w}$ to the user, we can also manually adjust its value at each joint to finely control the retargeted results. To reach an optimized balancing, $\mathcal{F}_w$ can be trained by:

$$\min_{\boldsymbol{\theta}_w}\ \mathcal{L}_{base}(\boldsymbol{q}_T^{\gamma+\phi}, \boldsymbol{q}_T) + \kappa\mathcal{L}_{rep} + \iota\mathcal{L}_{att} + \tau\mathcal{L}_{reg}, \tag{18}$$

where κ, ι, and τ are hyper-parameters. $\mathcal{L}_{reg}$ is a $L2$ regularization loss for w.

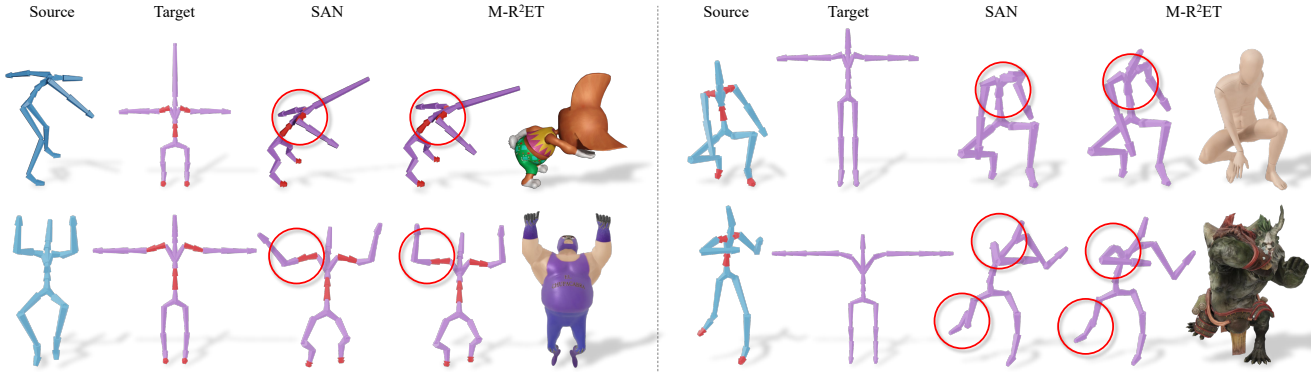


Fig. 9: Qualitative results of cross-structure motion retargeting. Joints highlighted in red indicate a different topology from the canonical skeleton template.

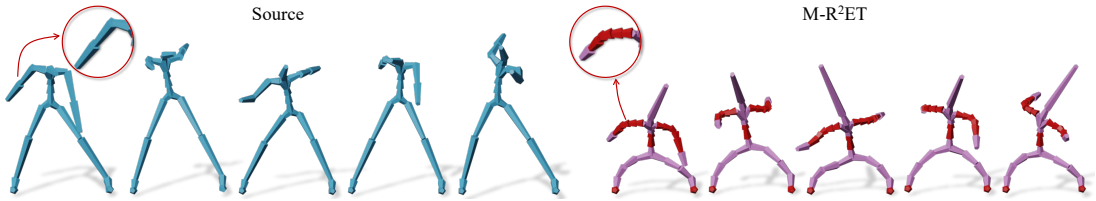


Fig. 10: Retargeted results of a difficult structure. Joints highlighted in red indicate a different topology from the canonical skeleton template.

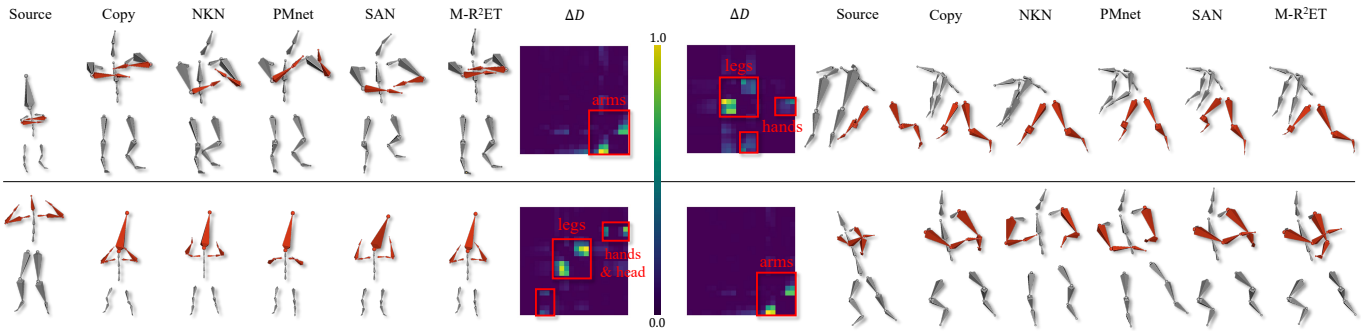


Fig. 11: Qualitative results of skeletal motion retargeting. ΔD indicates the DM difference between the motion copy and our result.

6 Experiments

In this section, we evaluate the retargeting results of M-R^{2}ET system, compare them to other motion retargeting methods, and demonstrate the efficiency of the four proposed modules in our system. Furthermore, we present the findings from a user study.

6.1 Experimental Setup

Datasets. We evaluate our M-R^{2}ET on the Mixamo dataset [10], which is an animation repository performed by multiple 3D virtual characters with different skeletons and shapes. For training, we collect 1952 non-overlapping motion sequences of seven characters and randomly sample 60 frames from each sequence. For testing, we collect 800 motion sequences of 11 characters and each sequence has 120 frames. We have unseen character (UC), unseen motion (UM), seen character (SC), and seen motion (SM) so that four splits UC+UM, UC+SM, SC+UM, SC+SM are considered in the experiment. Around 3/4 of the test samples are unseen. The Mixamo dataset does not provide clean Ground Truth (GT): many of the motions may have interpenetration or contact-missing issues which makes geometry learning challenging. For a fair comparison, we follow the spirit of [1] to implement experiments. The input motion for the modified skeleton is then generated by applying the cross-structure motion copy strategy to its ground-truth motion. It is worth noting that the target skeletons in this evaluation are all canonical skeletons, ensuring the accurate calculation of quantitative metrics.

Implementation details. The cross-structure alignment module (refer to Section 4) consists of a five-layer GCN with a $softmax$ activation function. The architectures of the shape-aware module $\Delta\mathcal{F}_g$ and the balancing gate $\mathcal{F}_w$ are detailed in Table 1. "QLinear" is a Linear Layer that

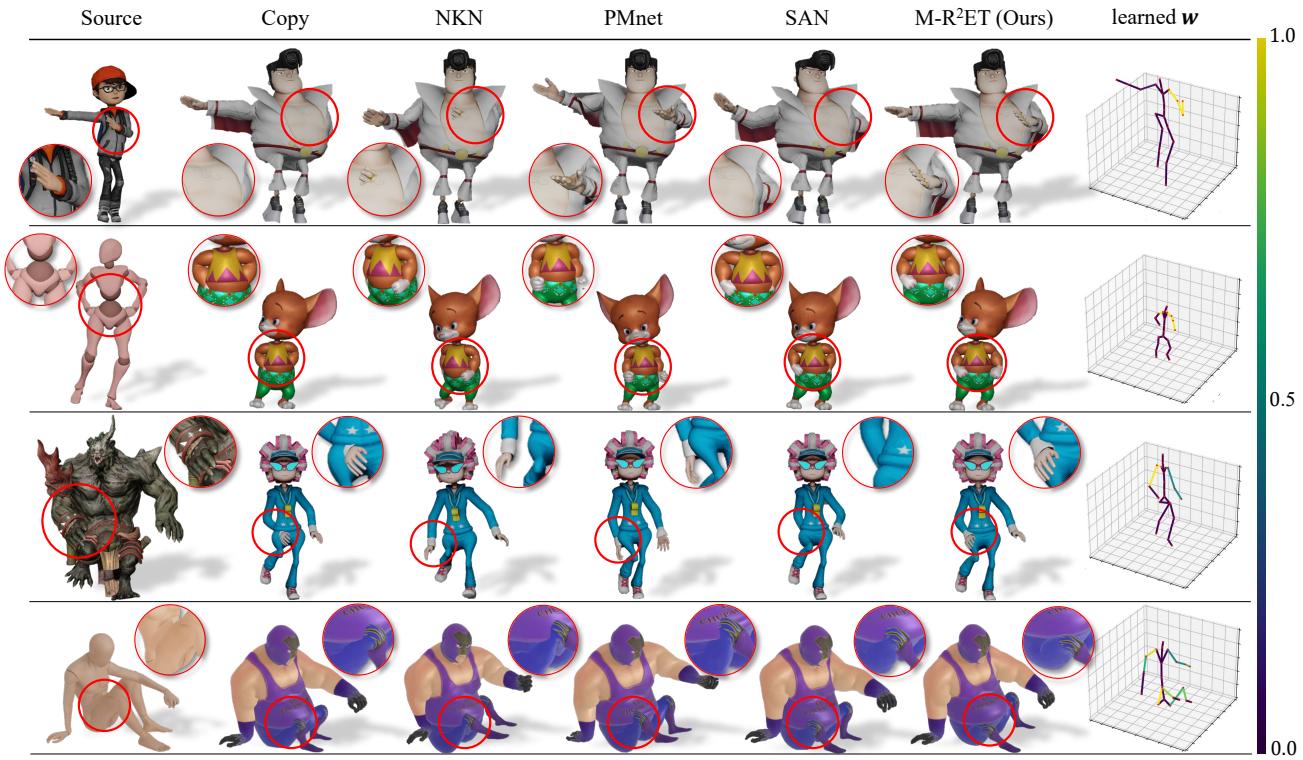


Fig. 12: Qualitative results of skinned motion retargeting. $\boldsymbol{w}$ is the adjusting weight learned by the balancing gate.

outputs quaternions, and its bias is initialized as a unit quaternion. The hyper-parameters λ, μ, ν, κ, ι, and τ in loss functions are set as 2.0, 10.0, 100.0, 0.5, 0.5, and 0.005, respectively. The margin factor α in the Rotation Constraint loss is defined as 100. We implement our model based on the PyTorch framework [48] and apply the Adam optimizer [49] to train the network. We use a single NVIDIA Tesla V100 GPU (16GB) and the training process is divided into three stages: firstly, train the cross-structure alignment module independently. Secondly, train the skeleton-aware module, and then freeze its parameters and train the shape-aware module and the balancing gate.

To train the cross-structure alignment module, the learning rate is set as 0.01, the number of training epochs is set as 100 and the batch size is 128. To train the skeleton-aware module, the learning rate is set as 0.001, the number of training epochs is set as 30 and the batch size is 32. To train the shape-aware module and the balancing gate, the learning rate is set as 0.0001, the number of training epochs is set as 50 and the batch size is 16. Notably, in the third stage, we set the balancing gate $\boldsymbol{w}$ to $\mathbf{1}$ with a probability of 0.3, which makes the learning process of our shape-aware module is stable.

The canonical skeleton template consists of 22 joints as Figure 4 shows. We observe that 22 joints are enough to visually present the humanoid motion. The joints we used include Hips, Spine, Spine1, Spine2, Neck, Head, LeftUpLeg, LeftLeg, LeftFoot, LeftToeBase, RightUpLeg, RightLeg, RightFoot, RightToeBase, LeftShoulder, LeftArm, LeftForeArm, LeftHand, RightShoulder, RightArm, RightForeArm, and RightHand.

The voxelizing process of RDF and ADF is implemented as follows: First, we rescale the deformed mesh into a tight box such that the vertice coordinates are ranged in $[-1, 1]$. Then, we uniformly sample 32 points in this box and calculate the distance from each point to the surface as the value of a voxel. Finally, for the RDF, we set the values of the voxels outside the mesh to 0. For the ADF, if a voxel is inside the mesh or its value is larger than 0.2, we set its value to 0.

6.2 Qualitative Results

Structures. Figure 9 presents the visualization results obtained by our M-R^{2}ET on cross-structure motion retargeting. Comparing to the results of SAN [6], our M-R^{2}ET specifically addresses the cross-structure issue. The joints highlighted in red signify a different topology from the canonical skeleton template. To ensure a fair comparison, we employ the same skeleton topology as the one provided in the code of SAN. This choice is made as the pre-trained SAN is not capable of handling various skeleton topologies beyond its pre-defined one. The comparison results strongly demonstrate the superiority of our M-R^{2}ET in effectively retargeting motion across different skeleton topologies, with the retargeted results exhibiting superior alignment with the source motion. It is essential to emphasize that our M-R^{2}ET system, enabled by the cross-structure alignment module that aligns skeletons to a canonical space, allows for motion retargeting between humanoid characters of arbitrary topology without the need for retraining the model. This capability significantly enhances the versatility and practicality of our approach for a wide range of applications.

In Figure 10, we showcase the retargeted results of our M-R^{2}ET on a challenging target skeleton structure. This

TABLE 2: Comparison with the state-of-the-arts on intra-structure skinned motion retargeting. MSE^{local} is the local MSE. M-R^{2}ET$_{w/oGW}$ is the model with the skeleton-aware module only. M-R^{2}ET$_{w/oW}$ is the model without the balancing gate. Copy† is the motion copy without the global motion normalization.

Methods	Input	MSE$_\downarrow$	$\text{MSE}^{local}{}_\downarrow$	Penetration$^{\%}_\downarrow$	Contact$^{cm}_\downarrow$
GT	-	-	-	9.02	4.92
NKN [1]	Position	2.298	0.575	8.96	4.42
PMnet [5]		0.806	0.281	7.11	14.7
CAR [7]		0.745	0.232	5.72	4.11
ItMRnet [8]		0.653	0.220	8.55	10.3
Copy	Rotation	0.267	0.060	9.23	4.95
Copy†		3.087	0.060	9.23	4.95
SAN [6]		0.321	0.118	8.91	4.86
PMnet*		0.374	0.120	9.03	5.24
M-R^{2}ET$_{w/oGW}$	Rotation.	**0.297**	**0.094**	9.09	4.93
M-R^{2}ET$_{w/oW}$		0.378	0.178	**4.68**	5.31
M-R^{2}ET (Ours)		0.318	0.116	5.94	**3.57**

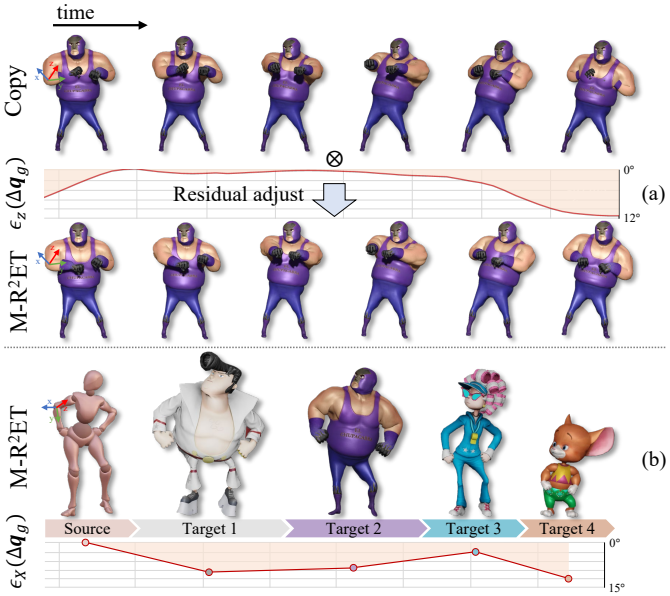


Fig. 13: (a) The change of the geometry-oriented modification $\Delta \boldsymbol{q}_g$ of a motion sequence on time domain. (b) Our results of retargeting one source to multi-targets.

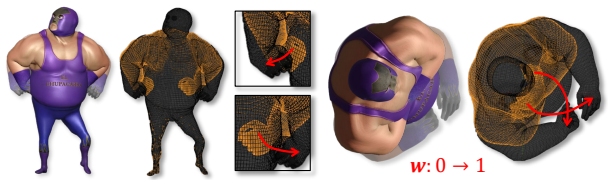


Fig. 14: Manually adjusting the balancing gate $\boldsymbol{w}$ can obtain smooth motion adjustment.

skeleton structure is characterized by significant differences in topology and proportions compared to the canonical skeleton. Unlike common humanoid characters, which typically feature three arm bones, the character in this instance has seven arm bones, posing a challenge for SAN to learn the joint pooling strategy. Leveraging the robust cross-structure alignment module, our M-R^{2}ET effectively handles this intricate structure, producing reasonable retargeted results.

Semantics. Figure 11 visualizes the effect of the skeleton-aware module of M-R^{2}ET on motion semantics preservation in skeletal motion retargeting. M-R^{2}ET$_{w/oGW}$ means our method with only the Skeleton-aware Module equipped. We retarget the motion of the bones among small, medium, and large skeletons with different bone length ratios. The skeleton-aware module can well preserve the motion semantics according to the characteristics of the target character's skeleton. For example, in the top-left row of Figure 11, the "arm folding" pose is retargeted from a small character to a large one, but the existing methods cannot accurately preserve the semantics, and the result of motion copy is more like "hand clapping". Our M-R^{2}ET can perceive the key differences between these two skeletons, i.e., the bone length ratio of the arms to the forearms, and adaptively adjust the copied rotations to generate a reasonable retargeted motion. The following three cases further demonstrate that our method, driven by the Semantics Similarity Loss, can well perceive the skeleton motion semantics and translate it between skeletons with large differences.

Geometry. Figure 12 shows the results of skinned motion retargeting among characters with different shapes. The target characters of the last two rows are unseen characters. The existing methods barely consider the shape geometry of the target characters, and their results suffer from severe issues of interpenetration and contact-missing. In contrast, our results, which are based on the semantics-preserved motion and adjusted by the shape-aware module as well as the balancing gate, can well reduce these implausible problems meanwhile maintaining motion semantics as much as possible without the post-processing. Figure 12 also visualizes the $\boldsymbol{w}$ of each joint predicted by the balancing gate. The predicted $\boldsymbol{w}$ have higher responses on the joints whose succeeding body parts are interpenetrated, and have lower responses when the motion semantics of the corresponding parts need to be preserved.

Figure 13 (a) shows the change of the geometry-oriented

TABLE 3: Comparison with the state-of-the-arts on cross-structure skinned motion retargeting. $\text{MSE}^{cano.}$ is calculated on the joints of the canonical skeleton. M-R^{2}ET$_{w/oSGW}$ is the model without the semantics and geometry perception modules.

Methods	$\text{MSE}^{cano.}_{\downarrow}$	$\text{Penetration}^{\%}_{\downarrow}$	$\text{Contact}^{cm}_{\downarrow}$
SAN [6]	1.012	9.36	6.44
M-R^{2}ET$_{w/oSGW}$	**0.324**	9.59	5.93
M-R^{2}ET (Ours)	0.372	**5.31**	**4.57**

TABLE 4: Ranking results of the user study. We invite 100 users to compare our retargeting results to that of the recent methods from three aspects, i.e., overall quality (Q), semantics preservation (S), and motion details (D).

Methods	Skeletal Motion			Skinned Motion		
	Q$_\downarrow$	S$_\downarrow$	D$_\downarrow$	Q$_\downarrow$	S$_\downarrow$	D$_\downarrow$
Copy	1.88	1.83	1.84	1.84	1.84	1.93
NKN [1]	3.37	3.45	3.40	3.44	3.44	3.42
PMnet [5]	3.06	3.06	3.06	3.10	3.07	3.00
M-R^{2}ET (Ours)	**1.69**	**1.67**	**1.70**	**1.63**	**1.65**	**1.64**

modification $\Delta \boldsymbol{q}_g$ with time of a motion sequence in the Mixamo dataset. The vector $\Delta \boldsymbol{q}_g$ is converted to the average z-axis Euler angle value of two arms for simple illustration. Our shape-aware module can accurately perceive the poses with interpenetration problems as the change of time and apply reasonable adjustments to them. For the poses that are not suffering from interpenetration, our M-R^{2}ET hardly adjusts them, so as to keep the original motion semantics as much as possible. At the same time, the modification changes smoothly as time goes on, which ensures the coherence and naturalness of the retargeted motion. Figure 13 (b) shows our results of retargeting one source motion to multiple targets. Targets 1,2,3 are unseen characters. For characters with different body shapes, our M-R^{2}ET can sensitively perceive their geometries and make precise adjustments to the source motion. Overall, Figure 13 demonstrates that our M-R^{2}ET is robust to a variety of poses and characters.

An automatic algorithm can provide visual results that follow the pre-defined learning constraints designed by engineers, such as the avoidance of interpenetration and contact-missing, but cannot always satisfy the aesthetic needs of the animators. Thanks to the flexible balancing gate, our M-R^{2}ET overcomes this drawback as Figure 14 shows. By gradually scaling $\boldsymbol{w}$, we can get results that vary smoothly from the arm pose that preserves the motion semantics to the one that avoids interpenetration, thereby interactively selecting the visually best results.

6.3 Quantitative Results

Comparison with the state-of-the-arts. Table 2 shows the comparison between our method and the state-of-the-arts on intra-structure skinned motion retargeting. Considering that the Mixamo dataset may create a new character with an archived motion by using motion copy, the "Copy" has the lowest MSE and local MSE. However, this does not mean that the motion copy is the best choice (See Figure 11 and

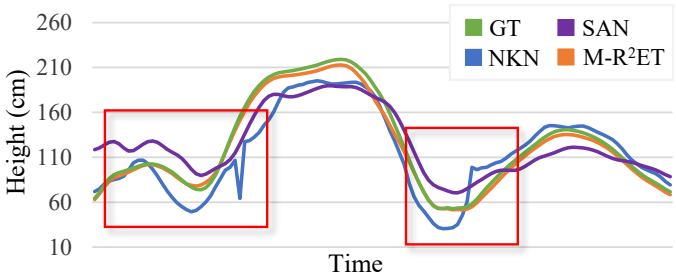


Fig. 15: The height change of the left-hand end-effector in a retargeted motion.

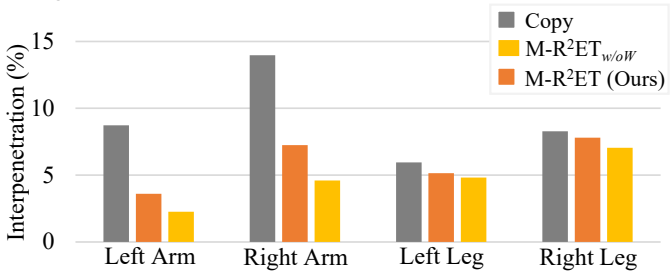


Fig. 16: Comparison of the interpeneration rates of different limbs.

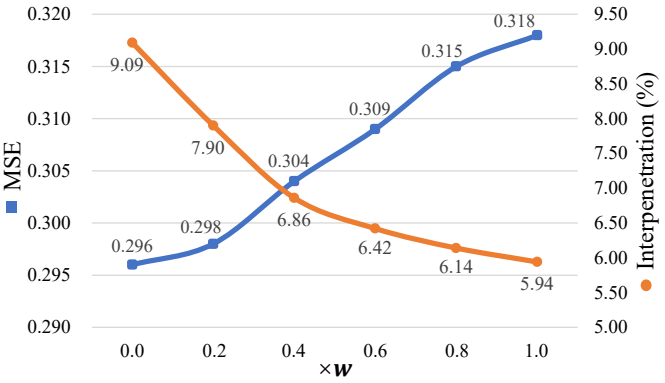


Fig. 17: The curves of MSE and penetration rate of our results as the balancing weight $\boldsymbol{w}$ is varied.

Figure 12). We here just treat MSE as an auxiliary reference metric for comparison. Compared with NKN, PMnet, and SAN which focus on skeletal motion retargeting, our M-R^{2}ET$_{w/oGW}$ reduces the MSE by 87% (0.297 vs 2.298), 63% (0.297 vs 0.806), and 7% (0.297 vs 0.321), respectively. The MSE of the PMnet with rotation input (PMnet*) is lower than the PMnet with position input but is also worse than ours. The above results show that our method can well reconstruct the source motion while adjusting the local motion according to the skeletal configurations to make it more in line with the motion semantics.

As shown in Table 2, the GT of the Mixamo dataset bears the issues of interpenetration and contact-missing. Our M-R^{2}ET$_{w/oW}$ system, with the help of the shape-aware module, can perceive the geometry of characters and reduce the interpenetration effectively. Compared with the GT, our M-R^{2}ET$_{w/oW}$ system reduces the penetration rate by more than 48% (4.68 vs 9.02). Without the balancing gate equipped, the contact can not be well maintained but the interpenetration is still reduced. Our full model *i.e.*, M-R^{2}ET,

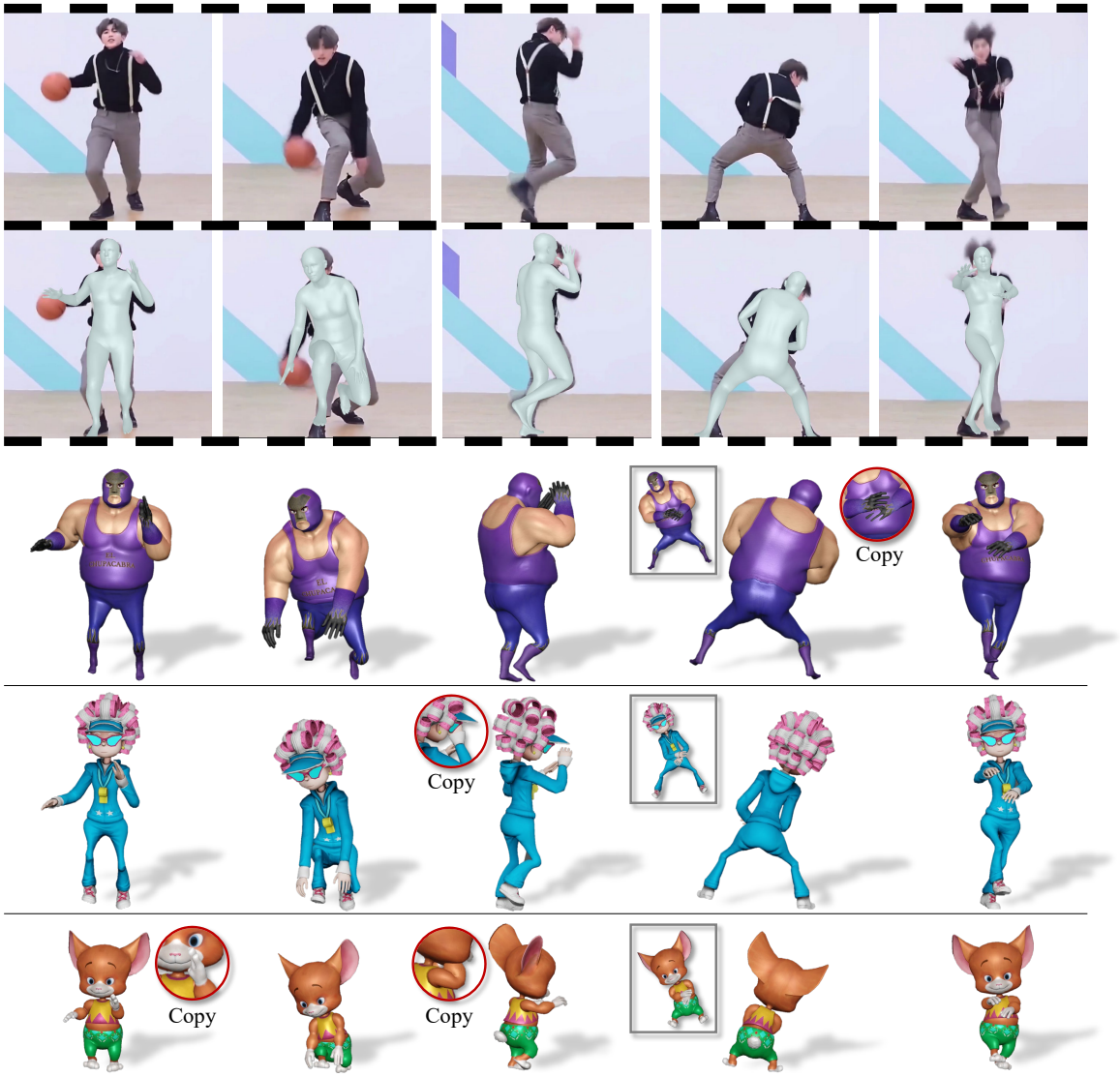


Fig. 18: Motion retargeting from the video motion capture data. We retarget motion that is estimated by [50] from a wild video into different characters.

reaches a good balance among these three quantitative metrics and obtains the best qualitative visualization results (see Figure 12).

Table 3 presents a comparison between our method and SAN on cross-structure skinned motion retargeting. The $\text{MSE}^{cano.}$ metric is evaluated on the joints of the canonical skeleton. M-R^{2}ET$_{w/oSGW}$ is the model without the semantics and geometry perception modules. The results clearly indicate that our cross-structure alignment strategy outperforms SAN, achieving a significantly lower $\text{MSE}^{cano.}$ value of 0.324 compared to 1.012 obtained by SAN. Additionally, our M-R^{2}ET system is able to effectively reduce interpenetration (5.31 vs 9.36) and preserve contact (4.57 vs 6.44) on cross-structure motion retargeting.

The Contact-aware Model (CAR) [7] focuses on skinned motion retargeting, which can also effectively reduce interpenetration and preserve self-contacts as Table 2 shows. However, unlike our M-R^{2}ET, CAR adopts a post-processing method to optimize the latent space of motion feature, which may not generate plausible results in a single inference pass, and it may result in unstable real-time inference. Therefore, our method is more efficient and easier to use. In addition, CAR heavily relies on high-quality source motion. Unfortunately, the majority of characters in the Mixamo dataset lack a clean ground-truth motion, resulting in inaccuracies in contact and interpenetration. This misguides CAR's optimization objective. Consequently, our M-R^{2}ET demonstrates greater generalizability, and our results outperform CAR in both motion precision and geometry quality, as detailed in Table 2.

Figure 15 shows the change of the end-effector's height of a retargeted motion on time domain. Compared to the NKN and SAN that are based on the full-motion mapping structure, our M-R^{2}ET with the residual structure can obtain smooth and stable retargeted motion in time series.

Ablation study. Figure 16 shows the comparisons of the penetration rates of the character's four limbs between the motion copy and our models. The interpenetration issues mainly occur between the arms and the body of the character, and our models can significantly reduce their

penetration rates.

The Mixamo dataset does not provide perfect ground truth: many of the motion sequences may have interpenetration or contact-missing issues. Thus, our M-R^{2}ET will cause an increase in the MSE of the joint positions while alleviating the interpenetration problem. Figure 17 shows the curves of MSE and penetration rate of our results as the change of the balancing weight $\boldsymbol{w}$.

6.4 User Study

We conduct a user study to evaluate the performance of our M-R^{2}ET against the relevant methods NKN, PMnet, and motion copy. We invited 100 users and gave them six skeletal action videos and seven shape action videos in total to evaluate. Each video includes one source motion and four anonymous results. We ask users to rank the four results in three aspects: overall quality (Q), semantics preservation (S), and motion details (D). After that, we exclude the questionnaires whose verification questions are incorrectly answered or are completed in less than 10 minutes. In the end, 80 questionnaires are retained, which contained 3120 ranking comparison results, and the average rank of the methods is summarized in Table 4. For skeletal motion, our method ranks 1.68 on average. For skinned motion, our method ranks 1.64 on average. In general, more than 71.2% of users prefer the retargeting results of our method.

6.5 Additional Applications

We have also implemented an additional application of our M-R^{2}ET system, as exemplified in Figure 18, where we demonstrate its effectiveness in retargeting video motion capture data. In this application, we utilize [50] to estimate the SMPL model from an in-the-wild video, which has a different skeleton structure compared to Mixamo characters. The results of this application show that our M-R^{2}ET performs well on video motion capture data. It successfully handles cross-structure motion retargeting, ensuring seamless transfer of motion between diverse skeleton structures, while preserving the motion semantics and avoiding issues related to interpenetration.

7 Conclusion

This work proposed a novel neural motion retargeting system called M-R^{2}ET, featuring a modular structure, to achieve a complete retargeting process in a single inference. In M-R^{2}ET, a cross-structure alignment module is devised to standardize various skeleton typologies into a canonical skeleton space. Meanwhile, two motion modification modules are designed to generate plausible target motion. Specifically, the skeleton-aware module adjusts the input motion to retain the source motion semantics in the target character, ensuring the preservation of the original motion's essence. Simultaneously, the shape-aware module evaluates the compatibility between the target shape and the semantics-preserved pose, effectively avoiding issues such as interpenetration and contact-missing. Moreover, a balancing gate is exploited to achieve a harmonious trade-off between the skeleton-level and geometry-level modifications by learning an adjusting weight. This allows for fine-tuning the balance between preserving motion semantics and minimizing interpenetration and contact-missing. With the aid of the two distance-based measurements, M-R^{2}ET is trained in a self-supervised fashion. Extensive experiments on the Mixamo dataset show its state-of-the-art motion retargeting performance. It excels in enabling cross-structure motion retargeting and striking an ideal balance between preserving motion semantics and geometries, all without the need for post-processing.

Limitations. One potential drawback of the proposed M-R^{2}ET lies in the presence of noisy motion data. For future work, we are committed to reducing noise interference and enhancing the robustness of the system. While foot contact is not the primary focus of our current research, it can be addressed by utilizing the method presented in [6]. Regarding quadrupeds, their distinct movement patterns significantly differ from those of the humanoid characters, making it impractical to learn them solely from humanoid motion data. Therefore, our M-R^{2}ET system currently does not support motion retargeting among quadrupeds. However, exploring motion retargeting techniques specifically tailored for quadrupeds could be a valuable avenue for future research.

Acknowledgment

This work was supported by the Natural Science Fund for Distinguished Young Scholars of Hubei Province under Grant 2022CFA075, and the National Natural Science Foundation of China (NSFC) under Grant 62106177. The numerical calculation was supported by the super-computing system in the Super-computing Center of Wuhan University.

References

[1] R. Villegas, J. Yang, D. Ceylan, and H. Lee, "Neural kinematic networks for unsupervised motion retargetting," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8639–8648.

[2] C. Hecker, B. Raabe, R. W. Enslow, J. DeWeese, J. Maynard, and K. van Prooijen, "Real-time motion retargeting to highly varied user-created morphologies," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, pp. 1–11, 2008.

[3] S. Tak and H.-S. Ko, "A physically-based motion retargeting filter," *ACM Transactions on Graphics*, vol. 24, no. 1, pp. 98–117, 2005.

[4] S. Hoshyari, H. Xu, E. Knoop, S. Coros, and M. Bächer, "Vibration-minimizing motion retargeting for robotic characters," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 1–14, 2019.

[5] J. Lim, H. J. Chang, and J. Y. Choi, "Pmnet: learning of disentangled pose and movement for unsupervised motion retargeting," in *30th British Machine Vision Conference*. British Machine Vision Association, BMVA, 2019.

[6] K. Aberman, P. Li, D. Lischinski, O. Sorkine-Hornung, D. Cohen-Or, and B. Chen, "Skeleton-aware networks for deep motion retargeting," *ACM Transactions on Graphics*, vol. 39, no. 4, pp. 62–1, 2020.

[7] R. Villegas, D. Ceylan, A. Hertzmann, J. Yang, and J. Saito, "Contact-aware retargeting of skinned motion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9720–9729.

[8] S. Li, L. Wang, W. Jia, Y. Zhao, and L. Zheng, "An iterative solution for improving the generalization ability of unsupervised skeleton motion retargeting," *Computers & Graphics*, vol. 104, pp. 129–139, 2022.

[9] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5745–5753.

[10] Adobe, "Mixamo," https://www.mixamo.com/.

[11] J. Zhang, J. Weng, D. Kang, F. Zhao, S. Huang, X. Zhe, L. Bao, Y. Shan, J. Wang, and Z. Tu, "Skinned motion retargeting with residual perception of motion semantics & geometry," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 13 864–13 872.

[12] M. Gleicher, "Retargetting motion to new characters," in *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, 1998, pp. 33–42.

[13] A. Savenko and G. Clapworthy, "Using motion analysis techniques for motion retargeting," in *Proceedings Sixth International Conference on Information Visualisation*. IEEE, 2002, pp. 110–115.

[14] J. Lee and S. Y. Shin, "A hierarchical approach to interactive motion editing for human-like figures," in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 1999, pp. 39–48.

[15] B. Dariush, M. Gienger, A. Arumbakkam, C. Goerick, Y. Zhu, and K. Fujimura, "Online and markerless motion retargeting with kinematic constraints," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008, pp. 191–198.

[16] K.-J. Choi and H.-S. Ko, "Online motion retargetting," *The Journal of Visualization and Computer Animation*, vol. 11, no. 5, pp. 223–235, 2000.

[17] K. Ayusawa, M. Morisawa, and E. Yoshida, "Motion retargeting for humanoid robots based on identification to preserve and reproduce human motion features," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 2774–2779.

[18] A. Bernardin, L. Hoyet, A. Mucherino, D. Gonçalves, and F. Multon, "Normalized euclidean distance matrices for human motion retargeting," in *Proceedings of the Tenth International Conference on Motion in Games*, 2017, pp. 1–6.

[19] A. Feng, Y. Huang, Y. Xu, and A. Shapiro, "Automating the transfer of a generic set of behaviors onto a virtual character," in *International Conference on Motion in Games*. Springer, 2012, pp. 134–145.

[20] H. Jang, B. Kwon, M. Yu, S. U. Kim, and J. Kim, "A variational u-net for motion retargeting," in *SIGGRAPH Asia 2018 Posters*, 2018, pp. 1–2.

[21] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. Tenenbaum, "Deep convolutional inverse graphics network," *Advances in neural information processing systems*, vol. 28, 2015.

[22] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[23] D. Rempe, L. J. Guibas, A. Hertzmann, B. Russell, R. Villegas, and J. Yang, "Contact and human dynamics from monocular video," in *European conference on computer vision*. Springer, 2020, pp. 71–87.

[24] R. Shah, V. Srivastava, and P. Narayanan, "Geometry-aware feature matching for structure from motion applications," in *2015 IEEE Winter Conference on Applications of Computer Vision*. IEEE, 2015, pp. 278–285.

[25] L.-Y. Gui, Y.-X. Wang, X. Liang, and J. M. Moura, "Adversarial geometry-aware human motion prediction," in *Proceedings of the european conference on computer vision (ECCV)*, 2018, pp. 786–803.

[26] T. L. Gomes, R. Martins, J. Ferreira, R. Azevedo, G. Torres, and E. R. Nascimento, "A shape-aware retargeting approach to transfer human motion and appearance in monocular videos," *International Journal of Computer Vision*, vol. 129, no. 7, pp. 2057–2075, 2021.

[27] S. Peng, J. Dong, Q. Wang, S. Zhang, Q. Shuai, X. Zhou, and H. Bao, "Animatable neural radiance fields for modeling dynamic human bodies," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 314–14 323.

[28] B. Jiang, Y. Hong, H. Bao, and J. Zhang, "Selfrecon: Self reconstruction your digital avatar from monocular video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5605–5615.

[29] T. Jin, M. Kim, and S.-H. Lee, "Aura mesh: Motion retargeting to preserve the spatial relationships between skinned characters," in *Computer Graphics Forum*, vol. 37, no. 2. Wiley Online Library, 2018, pp. 311–320.

[30] J. Basset, S. Wuhrer, E. Boyer, and F. Multon, "Contact preserving shape transfer: Retargeting motion from one shape to another," *Computers & Graphics*, vol. 89, pp. 11–23, 2020.

[31] Y. Peng, Y. Yan, S. Liu, Y. Cheng, S. Guan, B. Pan, G. Zhai, and X. Yang, "Cagenerf: Cage-based neural radiance field for generalized 3d deformation and animation," *Advances in Neural Information Processing Systems*, vol. 35, pp. 31 402–31 415, 2022.

[32] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM transactions on graphics*, vol. 34, no. 6, pp. 1–16, 2015.

[33] Q. Tan, N. Liu, and X. Hu, "Deep representation learning for social network analysis," *Frontiers in big Data*, vol. 2, p. 2, 2019.

[34] K. Z. Khanam, G. Srivastava, and V. Mago, "The homophily principle in social network analysis: A survey," *Multimedia Tools and Applications*, vol. 82, no. 6, pp. 8811–8854, 2023.

[35] J. Zhang, X. Shi, S. Zhao, and I. King, "Star-gcn: Stacked and reconstructed graph convolutional networks for recommender systems," *arXiv preprint arXiv:1905.13129*, 2019.

[36] H. Tang, G. Zhao, X. Bu, and X. Qian, "Dynamic evolution of multi-graph based collaborative filtering for recommendation systems," *Knowledge-Based Systems*, vol. 228, p. 107251, 2021.

[37] Y. Xie, Z. Xu, J. Zhang, Z. Wang, and S. Ji, "Self-supervised learning of graph neural networks: A unified review," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 2, pp. 2412–2429, 2022.

[38] Z. Tu, J. Zhang, H. Li, Y. Chen, and J. Yuan, "Joint-bone fusion graph convolutional network for semi-supervised skeleton action recognition," *IEEE Transactions on Multimedia*, 2022.

[39] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[40] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *International Conference on Learning Representations*, 2018.

[41] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, "Graph transformer networks," *Advances in neural information processing systems*, vol. 32, 2019.

[42] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, vol. 30, 2017.

[43] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," *Advances in neural information processing systems*, vol. 29, 2016.

[44] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" in *International Conference on Learning Representations*, 2018.

[45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[46] S. Xiao, S. Wang, Y. Dai, and W. Guo, "Graph neural networks in node classification: survey and evaluation," *Machine Vision and Applications*, vol. 33, pp. 1–19, 2022.

[47] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[48] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *The International Conference on Learning Representations*, 2015.

[50] Y. Sun, W. Liu, Q. Bao, Y. Fu, T. Mei, and M. J. Black, "Putting people in their place: Monocular regression of 3d people in depth," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Jiaxu Zhang received his B.S. degree from Southeast University, Nanjing, China, in 2020. He is currently working toward a Ph.D. degree at the LIESMARS (State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing), Wuhan University, China. His research interests include computer vision, computer graphics, and motion synthesis.

Zhigang Tu (Member, IEEE) received the Ph.D. degree from Wuhan University, China, in 2013, and the Ph.D. degree from Utrecht University, Netherlands, in 2015. From 2015 to 2016, he was a Postdoctoral Researcher with Arizona State University, USA. From 2016 to 2018, he was a Research Fellow with Nanyang Technological University, Singapore.

He is currently a Professor with Wuhan University. He has co-/authored more than 70 papers in international SCI-indexed journals and conferences. His current research interests include computer vision, image processing, video analytics, machine learning, motion estimation, human action and gesture recognition, and anomaly event detection. He is the first organizer of the ACCV2020 Workshop on MMHAU, Japan. He received the Best Student Paper Award at the 4^{th} Asian Conference on Artificial Intelligence Technology and one of the three best reviewers awards for *IEEE Transactions on Circuits and Systems for Video Technology (IEEE T-CSVT)* in 2022. He is the Area Chair of AAAI2023/2024 and VCIP2022. He is an Associate Editor of the SCI-indexed journal *The Visual Computer* (IF=3.5) and a Guest Editor of *Journal of Visual Communications and Image Representation* (IF=2.6).

Junwu Weng obtained his Ph.D. degree from Nanyang Technological University (NTU), Singapore, in 2020. Before that, he received both his M.Eng. degree and B.Eng degree from South China Univeristy of Technology (SCUT), Guangdong, in 2015 and 2012 respectively. He is now a senior researcher in Tencent AI Lab. His research interests include Video Understanding, Cross-modality Learning, Noisy Label Learning, Motion Retargeting.

Junsong Yuan (Fellow, IEEE) is Professor and Director of Visual Computing Lab at Department of Computer Science and Engineering, State University of New York at Buffalo (UB), USA. Before that he was Associate Professor (2015-2018) and Nanyang Assistant Professor (2009-2015) at Nanyang Technological University (NTU), Singapore. He obtained his Ph.D. from Northwestern University in 2009, M. Eng. from National University of Singapore in 2005, and B. Eng. from Huazhong University of Science Technology in 2002. His research interests include computer vision, pattern recognition, video analytics, large-scale visual search and mining. He received Best Paper Award from IEEE Trans. on Multimedia, Nanyang Assistant Professorship from NTU, and Outstanding EECS Ph.D. Thesis award from Northwestern University.

He served as Associate Editor of IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI), IEEE Trans. on Image Process. (TIP), IEEE Trans. on Circuits and Systems for Video Tech. (TCSVT), and Senior Area Editor of Journal of Visual Communications and Image Representation. He was Program Co-Chair of IEEE Conf. on Multimedia Expo (ICME'18/2022/2024), and Area Chair for CVPR, ICCV, ECCV, and ACM MM. He was elected senator at both NTU and UB. He is a Fellow of IEEE and IAPR.

Bo Du (Senior Member, IEEE) received the Ph.D. degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China, in 2010. He is a Professor with the School of Computer Science, Wuhan University. He has over 80 research articles published in the journals of IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), IEEE Transactions on Image Processing (TIP), IEEE Transactions on Geoscience and Remote Sensing (TGRS), ISPRS Journal of Photogrammetry and Remote Sensing, etc. More than 30 of them are ESI hot articles or highly cited articles. His major research interests include pattern recognition, hyperspectral image processing, machine learning, and signal processing.