Modern Dance Retargeting using Ribbons as Lines of Action

M. Vialle[1], R. Ronfard[1] and M. Skouras[1]

[1]Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK

Abstract

We present a method for retargetting dancing characters represented as articulated skeletons with possibly different morphologies and topologies. Our approach relies on the use of flexible ribbons that can bend and twist as an intermediate representation, and that can be seen as animated lines of action. *These ribbons allow us to abstract away the specific morphology of the bodies and to well transmit the fluidity of modern dance movement from one character to another.*

CCS Concepts

• ***Computing methodologies*** → ***Animation;***

1. Introduction

Transferring an animation from one skeleton to another is a well-known problem in computer graphics called motion retargeting. In motion retargeting, the goal is to animate a skeleton, which we will call the target skeleton, from motion contained in another animated skeleton, which we will call the source skeleton. Because there is no guarantee that the source skeleton and the target skeleton have the same topology or morphology, the question is, how to transfer that animation from the source skeleton to the target skeleton?

In this work, we are interested in transferring *dance* movement, and more specifically, that of modern dance. Vialle et al. [VAS*22] recently proposed a new representation of modern dancers' bodies in the form of flexible ribbons joined at the solar plexus, in the context of the visualization of the movement qualities of Isadora Duncan (1877-1927), a pioneer of modern dance. They showed that this abstract representation was very successful at conveying the fluidity of movement that characterizes Duncan's choreography.

Interestingly, the ribbons of Vialle et al. are reminiscent of the *lines of action*, 2D hand-drawn curves used by cartoonists to help draw the first shape of expressive motion. Inspired by the work by Guay et al. [GCR13] who use lines of action to pose 3D characters, possibly through time [GRGC15, GGC15], we propose to use Vialle's ribbons as an intermediate representation to transfer dance animation from one character to another. Indeed, contrary to the articulated chains of the traditional skeletons used in animation, ribbons have a smooth shape that blurs the position of the joints. A same arrangement of ribbons could therefore be used for animating bodies with different morphologies.

Ribbons extend lines of action to 3D and allow for a richer set of deformation, including twisting. We use them in the same fashion as a spline IK [Aut09] that provides the ability to manipulate a chain of joints using a spline curve, with all the joints influenced by the spline IK handle remaining attached to the curve. We generate ribbons from the input motion at each time frame, which can be considered as lines of action, and then use them to animate the target skeleton.

2. State of the art

Motion retargeting has been investigated for almost three decades. While at first sight this may simply appear as a matter of transferring joint angles from one skeleton to another, this naive approach rarely performs well as source and target skeletons may have different proportions or even different topologies.

A first attempt to solve this problem comes from [Gle98] who introduced a space-time optimization framework with kinematic constraints. Optimization is performed through the entire motion sequence. A year later, [LS99] took a different approach, applying inverse kinematics on the motion to satisfy constraints and then applying a smoothing filter to the motion through multi-level B-spline curve fitting. [CK00] introduced an online retargeting method that computed changes in joint angles corresponding to end-effector position adjustments, preserving high-frequency details of the original motion. [MBBT00] introduced a new method using an intermediate skeleton that enables retargeting to skeletons with different morphologies. The intermediate skeleton has the same number of nodes and local axis systems orientation as the target skeleton, but the bones are oriented as the input skeleton nodes. Then, they apply inverse kinematics on the obtained skeleton. In our case, we use a similar metaphor of an intermediate/primal skeleton. However, our intermediate "skeleton" is the set of animated ribbons and has a much higher number of "joints", leading to the limbs' bending to spread more evenly along the articulated chain. [TK05] introduced a physically-based motion retargeting filter, leveraging dynamic constraints for physically plausible motions. [FHXS12]

DOI: 10.2312/egs.20241019

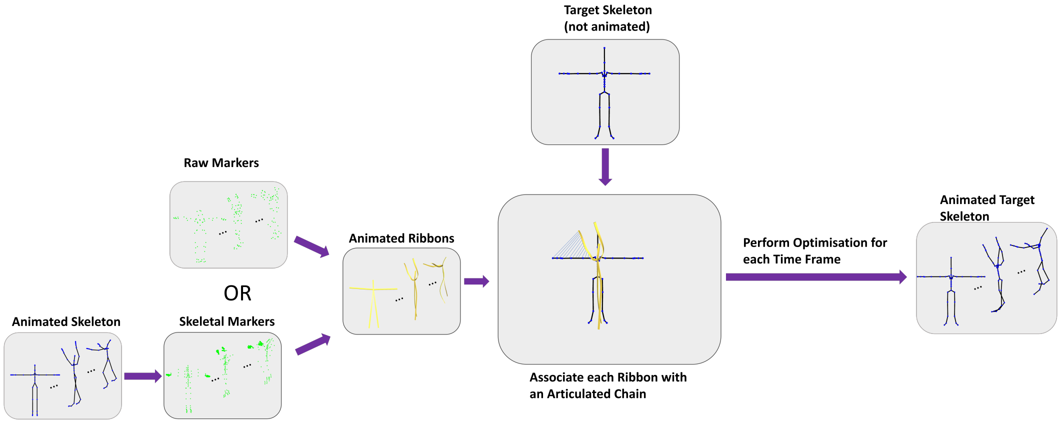


Figure 1: *Overview of our proposed method for dance motion retargeting.*

proposed heuristics for mapping arbitrary joints to canonical ones and outlined an algorithm to instill behaviors onto an arbitrary humanoid skeleton.

Classical motion retargeting approaches, like those mentioned above, primarily relied on optimization with hand-crafted kinematic constraints tailored to specific motions, often involving simplifying assumptions. With the increased availability of motion data, data-driven approaches have gained prominence. [DERC17] described a method for transferring learned latent representations of motions from one robot to another. [JKY*18] employed a deep autoencoder to optimize the latent space for desired changes in bone lengths, albeit requiring paired training data.

Several studies have explored retargeting human motion data to non-humanoid characters [SOL13, YAH10], where source and target skeletons may significantly differ. However, these approaches require captured motions of a human subject emulating the target character's style. They also involve selecting key poses from captured motion sequences and matching them to the corresponding character poses or pairing corresponding motions. [AMYB17] proposed representing a character's motion style through groups of body parts (GBPs) and suggested motion style retargeting across skeleton structures by establishing correspondence between GBPs. This process necessitates defining GBPs and their correspondence for each character pair. In [ALL*20], the skeleton is simplified in the latent space as a primal skeleton, and motion retargeting is performed by decoding the motion in the latent space through the target skeleton.

When looking at retargeting dance motion specifically, researchers focused on retargeting motion from one video to another using machine learning methods [CGZE19, RCW*21]. However, motion from video lacks information about the depth and displacements in space, which are important features in modern dance choreography.

3. Overview

An overview of our proposed method is described in Figure 1. The first step of our method is to generate the star-shaped ribbon representing the source dancer's body as described by Vialle et al. [VSR21, VAS*22]. The ribbons can be generated from the raw positions of 3D markers located on the surface of the dancer's body or from the positions of the joints of the animated skeleton. These markers are attached to originally straight ribbons through springs. As the markers move over time, they drag the ribbons along.

Then, we manually associate each ribbon with an articulated chain of the target skeleton. An *articulated chain* is defined as a set of connected bones, each bone being connected to one or two bones. Using springs, we then link each vertex of the ribbon to a corresponding point on the corresponding articulated chain. The springs' rest lengths are defined when the ribbons and the target skeleton are in a T-Pose.

Then, we use an optimization process at each time frame to minimize the summed energy of all springs connecting ribbons' vertices and their corresponding points on the articulated chain. This optimization process yields the corresponding pose of the target skeleton at each time frame. The output is then the resulting animated target skeleton. In the optimization process, the variables are the 3D positions of the skeleton's joints.

Algorithm 1 describes an overview of the process of generating the animated ribbons from motion-captured data (refer to [VSR21] for a detailed description of the process). In the following, we describe the process of animating a skeleton from our animated ribbons.

```
input  : Set of markers  output : Animated Ribbon
Create initial ribbon using PCA on the set of markers;
Generate rest state values by projecting markers on
  centerline;
for each time frame do
    Find optimum centerline position;
    Set centerline;
    Set Bishop frames;
    Find optimal material frame orientation;
    Set material frames;
end
```

Algorithm 1: Simulation steps to generate one ribbon

4. Animating a skeleton from a ribbon

The process of animating a skeleton from a ribbon is illustrated in Figure 2 and starts once all the ribbons have been generated using the source motion (Figure **??**). These ribbons are discretized using discrete elastic rods [BWR*08]. Each ribbon is defined as a rod made of a set of vertices $(x_i)_{i\in[\![0,n+1]\!]}$ and a set of frames $M^i = \{t^i, m_1^i, m_2^i\}$, adapted to the centerline of the rod, defined with respect to Bishop frames $B^i = \{t^i, u^i, v^i\}$.

We first manually select what part of the skeleton is affected by each animated ribbon (Figure **??**). Each articulated chain starts at the sternum and ends at an extremity (hands, feet and head), using the shortest path in the skeleton. A change in the choice of the articulated chain will lead to a change in the retargeted skeleton.

4.1. Preprocessing the ribbons

The animated ribbon is not guaranteed to be the same size as the chosen articulated chain. Therefore, we uniformly resize the ribbon to the length of the articulated chain at rest, i.e. we keep the ratios between edge lengths unchanged.When evaluating the rest length of the springs connecting the ribbons' vertices to their corresponding point on the articulated chain, there is also no guarantee that the orientation and position of the ribbons are close to the desired articulated chain. To guarantee closeness, we first generate another ribbon, which we will call *articulated chain ribbon* from the selected articulated chain during the T-pose, using the same algorithm as the one used to generate the animated ribbon, and using the joints of the articulated chain as marker input, as well as offsetted joints as additional virtual markers to properly orient the ribbons.

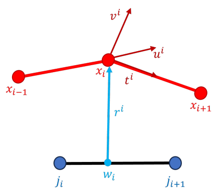
We then evaluate the translation between the *articulated chain ribbon* and the animation ribbon by evaluating the translation between their solar plexus. After the translation is performed, the rotation between the ribbons is defined as the rotation of minimal angle that aligns the animation ribbon to the *articulated chain ribbon*. As the ribbons are in a rest state and are straight 3D segments, this rotation can be seen as the parallel transport of the animation ribbon to the *articulated chain ribbon*. We then store this initial rotation.

Scaling, translation, and initial rotation are applied to the animated ribbons across all time frames. The translation vector is evaluated at each time frame, whereas scaling and initial rotation are evaluated during the T-pose.

4.2. Cost function

4.2.1. Shape energy

To ensure that the shape of the articulated chain will be as close as possible to the shape of the ribbon, we associate each vertex of the ribbon to the point on the articulated chain with the corresponding arc length position using springs. Let us denote the point on the articulated chain w_i (see inset figure). This point belongs to a bone b_i with extremities j_i, j_{i+1}. We can express w_i with respect to the joints using barycentric coordinates:



$$w_i = y j_i + (1-y) j_{i+1}, \quad y \in [0,1]. \tag{1}$$

We associate the vertex to the associated point using the vector

$$r^i = x_i - w_i. \tag{2}$$

In order to keep the natural angles (i.e., angles when in a T-Pose) of the target skeleton through time, we lock the vector r^i in the Bishop frame B^i associated with the edge e^i. We do so by asking the local coordinates of r^i in the Bishop frame basis to change as little as possible via the energy

$$\begin{aligned} E^i_{springs} =& ((\overline{r^i} \cdot \overline{t^i}) - (r^i \cdot t^i))^2 + ((\overline{r^i} \cdot \overline{u^i}) - (r^i \cdot u^i))^2 \\ &+ ((\overline{r^i} \cdot \overline{v^i}) - (r^i \cdot v^i))^2. \end{aligned} \tag{3}$$

To obtain the total spring energy, we sum this energy over all vertices of all ribbons.

4.2.2. Enforcing the bones constraint

In a rig-based animation, the length of the skeleton's bones remains constant throughout the animation. This inextensibility constraint can be enforced using a stretch energy of the form:

$$E^k_{stretch} = \alpha(||\overline{j_k} - \overline{j_{k+1}}|| - ||j_k - j_{k+1}||)^2, \tag{4}$$

for each bone k connecting joints j_k and j_{k+1}, where α is a strong penalty coefficient. The total energy to enforce the inextensibility of the bones is then obtained by summing over all bones in the skeleton. The final minimization problem is then:

$$\min_{j_i \in skeleton} \sum_{ribbons} \sum_i E^i_{spring} + \sum_k E^k_{stretch} \tag{5}$$

We minimize this cost function at each time frame to find the 3D position of each joint. As opposed to the generation of the ribbons, where each ribbon is generated independently from one another, we create *one* optimization problem that takes into account the total energy over all ribbons associated with the skeleton's articulated chains. This optimization process enables us to find the optimal position of the skeleton at each time frame.

5. Results

Figure 3 shows the input ribbons and two skeletons with different topologies retargeted using our algorithm in different poses of the *Prelude* dance, captured by Vialle et al. [VAS*22]. See the accompanying video for the corresponding animations. This first result shows that the proposed method is able to retarget the animation and that the shape of the input ribbon is well transferred to the target skeleton. However when looking at the animations, we note some instabilities around the feet area. This is due to the fact that the generated ribbons do not take the feet markers into consideration while we used the entire target skeleton (including the feet) when retargeting the dance.

6. Conclusion & Future work

We proposed a method for retargeting dance motion based on a flexible ribbon as intermediate representation. Initial results show that our approach is effective at transferring the movement to skeletons with different shapes while preserving the fluidity of the dance. Since no ground truth to the problem exists, running a user study to

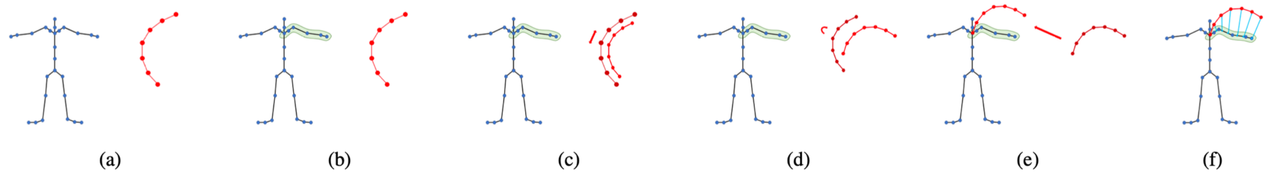


Figure 2: *Steps for pre-processing the ribbon: (a) Generate ribbon, (b) Manually select articulated chain (in green), (c) Scale the ribbon, (d) Rotate the ribbon, (e) Translate the ribbon, (f) Associate each vertex of the ribbon to its corresponding point on the articulated chain. Steps (c), (d) and (e) are run for each time frame.*

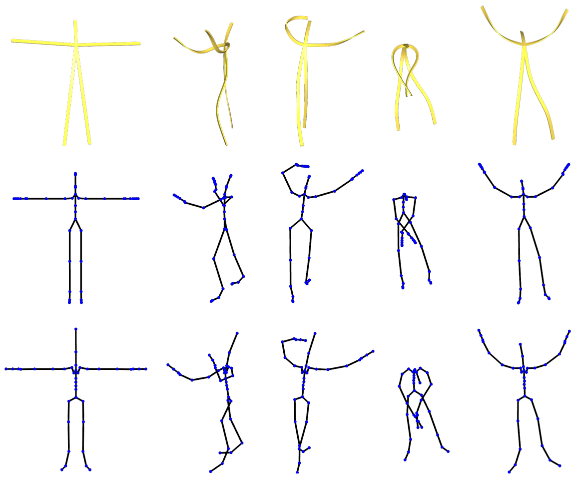


Figure 3: *Retargeting results. From top to bottom: input ribbon, retargeted humanoid skeleton with the same morphology as the input data, retargeted skeleton with added bones and different T-Pose shape.*

evaluate the "naturalness" of the resulting motion would be a useful validation of this work.

As future work, we would also like to adapt our algorithm to ribbons that have been generated with different topologies than the star shape used in this work. This could help retarget the motion to skeletons with more diverse topologies. We would also be interested in testing our algorithm with other types of dance to evaluate the generality of our method. Lastly, since the ribbons capture the motion qualities of Isadora Duncan, we could investigate if they could be used for the problem of motion style transfer.

References

[ALL*20] Aberman K., Li P., Lischinski D., Sorkine-Hornung O., Cohen-Or D., Chen B.: Skeleton-aware networks for deep motion retargeting. *ACM Trans. Graph. 39*, 4 (2020). 2

[AMYB17] Abdul-Massih M., Yoo I., Benes B.: Motion style retargeting to characters with different morphologies. *Computer Graphics Forum 36*, 6 (2017). 2

[Aut09] Autodesk: Maya, 2009. URL: http://www.autodesk.com/maya. 1

[BWR*08] Bergou M., Wardetzky M., Robinson S., Audoly B., Grinspun E.: Discrete Elastic Rods. *ACM Trans. Graph. 27*, 3 (2008). 3

[CGZE19] Chan C., Ginosar S., Zhou T., Efros A. A.: Everybody dance now. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2019). 2

[CK00] Choi K.-J., Ko H.-S.: Online motion retargetting. *The Journal of Visualization and Computer Animation 11*, 5 (2000), 223–235. 1

[DERC17] Delhaisse B., Esteban D., Rozo L., Caldwell D.: Transfer learning of shared latent spaces between robots with similar kinematic structure. In *Int. Joint Conf. on Neural Networks (IJCNN)* (2017). 2

[FHXS12] Feng A., Huang Y., Xu Y., Shapiro A.: Automating the transfer of a generic set of behaviors onto a virtual character. In *Motion in Games* (2012), Kallmann M., Bekris K., (Eds.). 1

[GCR13] Guay M., Cani M.-P., Ronfard R.: The Line of Action: an Intuitive Interface for Expressive Character Posing. *ACM Trans. Graph. 32*, 6 (2013). 1

[GGC15] Guay M., Gleicher R. R. M., Cani M.: Adding dynamics to sketch-based character animations. In *Proceedings of the workshop on Sketch-Based Interfaces and Modeling, SBIM'15* (2015). 1

[Gle98] Gleicher M.: Retargetting motion to new characters. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques* (1998), SIGGRAPH '98. 1

[GRGC15] Guay M., Ronfard R., Gleicher M., Cani M.-P.: Space-time sketching of character animation. *ACM Trans. Graph. 34*, 4 (2015). 1

[JKY*18] Jang H., Kwon B., Yu M., Kim S. U., Kim J.: A variational u-net for motion retargeting. In *SIGGRAPH Asia 2018 Posters* (2018), SA '18. 2

[LS99] Lee J., Shin S. Y.: 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), SIGGRAPH '99. 1

[MBBT00] Monzani J.-S., Baerlocher P., Boulic R., Thalmann D.: Using an intermediate skeleton and inverse kinematics for motion retargeting. *Computer Graphics Forum 19*, 3 (2000). 1

[RCW*21] Ren J., Chai M., Woodford O. J., Olszewski K., Tulyakov S.: Flow guided transformable bottleneck networks for motion retargeting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021). 2

[SOL13] Seol Y., O'Sullivan C., Lee J.: Creature Features: Online motion puppetry for non-human characters. In *Eurographics/ ACM SIGGRAPH Symposium on Computer Animation* (2013). 2

[TK05] Tak S., Ko H.-S.: A physically-based motion retargeting filter. *ACM Trans. Graph. 24*, 1 (2005). 1

[VAS*22] Vialle M., Alaoui S. F., Skouras M., Vilvanathan V., Schwartz E., Ronfard R.: Visualizing Isadora Duncan's Movements Qualities. In *Proceedings of the 14th Conference on Creativity and Cognition* (2022), C&C '22. 1, 2, 3

[VSR21] Vialle M., Skouras M., Ronfard R.: Elastic Rods for Fluid Dance Movements Modeling. In *AFIG 2021* (2021). URL: https://inria.hal.science/hal-04136667. 2

[YAH10] Yamane K., Ariki Y., Hodgins J.: Animating Non-Humanoid Characters with Human Motion Data. In *Eurographics/ ACM SIGGRAPH Symposium on Computer Animation* (2010). 2