Vis Comput (2015) 31:1569–1586
DOI 10.1007/s00371-014-1034-2

ORIGINAL ARTICLE

Correlation-optimized time warping for motion

S. Ali Etemad · Ali Arya

Published online: 11 October 2014

Abstract Retrieval and comparative editing/modeling of motion data require temporal alignment. In other words, for such processes to perform accurately, critical features of motion sequences need to occur simultaneously. In this paper, we propose correlation-optimized time warping (CoTW) for aligning motion data. CoTW utilizes a correlation-based objective function for characterizing alignment. The method solves an optimization problem to determine the optimum warping degree for different segments of the sequence. Using segment-wise interpolated warping, smooth motion trajectories are achieved that can be readily used for animation. Our method allows for manual tuning of the parameters, resulting in high customizability with respect to the number of actions in a single sequence as well as spatial regions of interest within the character model. Moreover, measures are taken to reduce distortion caused by over-warping. The framework also allows for automatic selection of an optimum reference when multiple sequences are available. Experimental results demonstrate the very accurate performance of CoTW compared to other techniques such as dynamic time warping, derivative dynamic time warping and canonical time warping. The mentioned customization capabilities are also illustrated.

Keywords Motion analysis · Time warping · Temporal alignment · Correlation · Optimization

Electronic supplementary material The online version of this article (doi:10.1007/s00371-014-1034-2) contains supplementary material, which is available to authorized users.

S. A. Etemad (✉) · A. Arya
School of Information Technology,
Carleton University, Ottawa, ON, Canada
e-mail: ali.etemad@carleton.ca

A. Arya
e-mail: arya@carleton.ca

1 Introduction

As a direct result of recent advancements in computer hardware and software, recorded or live streams of human motion data are being extensively exploited. From interactive gaming systems such as Kinect for Xbox (http://www.xbox.com/en-US/Kinect) to 3D virtual environments and animated movies, all deal with this type of data. Subsequently, retrieval and modeling sequences of human motion have attracted many researchers [1].

Generally, humans perform actions differently with respect to one another. It has been previously shown that factors such as gender, age, energy, health, ethnicity, and affect, influence the way we carry out actions [2]. As a result, even when motion data contain similar content, they vary in style, and these variations are manifested as spatiotemporal misalignments. On the other hand, in many cases, processing human motion requires motion data to be perfectly aligned. For example, it has been demonstrated that altering [3], blending [4], extraction [5], and translation [6] of style features require aligned motion data.

To tackle the issue of misalignment in time series, various techniques have been proposed based on application and context. For example, dynamic time warping (DTW) was proposed to align speech signals [7], while canonical time warping (CTW) [8] and iterative motion warping (IMW) [9] were proposed for motion data. In this paper, we propose correlation-optimized time warping (CoTW) for aligning motion sequences. The method proposed in this paper is inspired by correlation-optimized warping, which was initially developed for aligning chemometric data [10]. We briefly introduced the concept in [11] and utilized it in [5,6,12]. In this paper, we further develop the technique, study the parameters involved, and take a deep look into its performance, especially with respect to existing time-

warping methods. In addition to robust alignment, CoTW has several advantages over most other time-warping techniques: (a) it uses a more effective objective function that is based on correlation; (b) it allows for alignment to be customized both temporally and with respect to spatial regions of interest within the character model; (c) it reduces artifacts such as distortion and does not employ still frames that appear in existing methods for timing adjustments; (d) optimal reference is automatically selected when multiple sequences are being warped. While some of the previous works partially address these issues, to the best of our knowledge there is no warping technique that attends to them all. In the following sections, we study the related work, present the proposed method, study the parameters and the details of the method and perform rigorous experiments showing the robustness of the method.

2 Related work

Uniform scaling [13], or uniform time warping (UTW), is naturally the most simple and possibly the most naïve method for aligning sequences. This technique is useful for length matching of two or more time series and is not capable of aligning particular features in the process. However, this operation is used as one of the major components of our proposed method.

DTW [7] is one of the first non-linear warping techniques, which finds the similarity between two time series. This is done through calculating the corresponding instances in the series by minimizing a distance objective function. Multiple variations and extensions of this method [14–17] such as derivative dynamic time warping (DDTW) [14] have also been proposed. Generalized time warping (GTW) [18] has been proposed by Zhou and De La Torre as an effective extension to DTW. The extensions involve the capability of working with multiple modalities, for example motion capture data as well as recorded videos, more warping flexibility, and reduction in computational complexity.

Component-based approaches have also been explored for alignment. In this category, we can name methods that utilize independent component analysis (ICA) [19,20] and weighted principal component analysis (wPCA) [21]. Probabilistic methods [22] and Markov models [23–25] have also been utilized for aligning time series. In such methods, alignment is usually carried out in a testing phase through utilization of previously learned dynamic models.

The animation community has shown great interest in time warping for aligning motion capture data. For example, Bruderlin and Lance Williams [26] used DTW for interpolating between sequences. The motion warping method by Witkin and Popovic [27] is a variation of Bruderlin and Williams', and is intended to add small yet smooth changes to motion. The method proposed by Rose et al. [4] manually selects key frames that need to be aligned, according to which in-between frames are aligned. Gleicher [28] used spatial constraints and inverse kinematics to preserve quality, while retargeting motion. The methods used by Kovar and Gleicher [29,30] find correspondences between frames using distance minimization. Müller et al. [31] used DTW toward their retrieval method. This was done successive to segment-wise alignment using a proposed index. Müller and Röder [32] employed DTW to derive motion templates used in the classification and retrieval of motion capture data. Zhou et al. [33] used DTW in their proposed aligned cluster analysis that segmented motion capture data. Kim et al. [34] employed Laplacian curve manipulation to perform warping based on user-defined constraints. Raptis et al. [35] utilized DTW in their gesture classification algorithm. Cimen et al. [36] used DTW prior to extracting affective descriptors from motion. To address the problem of style translation, Hsu et al. [9], proposed iterative motion warping (IMW). Style translation is the process of transferring the style of one particular motion sequence onto another. This process requires accurately aligned sequences. IMW is composed of space and time-warping procedures (based on DTW) to address the problem. Finally, Hsu et al. [37] utilized pose-, velocity-, and acceleration-based feature vectors in their time warp objective function. Based on the above, most of the warping methods used in field of animation are built on UTW or DTW techniques.

As one of the more recent techniques, Zhou and De La Torre also proposed CTW and local canonical time warping (LCTW) [8] based on canonical correlation analysis (CCA) and DTW. Evaluated with synthetic, facial expression videos, and motion capture data, the method was shown to outperform other DTW-based techniques including IMW, DTW, and DDTW.

Finally, different approaches have been developed for alignment in the computer vision community. Homography computations are popular means in this regard [38–40]. Similar to techniques developed for motion capture data, most computer vision methods of alignment rely on DTW. For example, Junejo et al. [41] utilized self-similarity matrix and proposed a video alignment technique based on DTW. As another example, Lu and Mandal [42] proposed a method based on computation of the trajectory of object of interest and correspondence calculation using DTW.

Our proposed method, based on [10], has previously illustrated good performance in different aspects such as peak shape and area preservation [43], and has been widely explored and developed in the fields of chemistry [44–46]. The method has been used in image processing and biomedical image analysis [47] in addition to the initial application of chemometrics for which it was proposed. To the best of our knowledge, this method has not been the basis of any warping

techniques for human motion data. We believe further developing and tailoring this technique for human motion data can provide a new and robust method for aligning multiple motion sequences while addressing some of the shortcomings in currently available techniques.

3 Methodology

3.1 Algorithm

Temporal modifications can be carried out linearly. For example, a motion trajectory with temporal length m (i.e. m frames) can be linearly compressed or stretched to new length m'. This correction may or may not align the critical features that are of importance. For instance, a fatigued walk, which is slower than a normal walk, can be compressed to take the same temporal length as the regular walk. The strides, however, will not necessarily be aligned. This process, also referred to as uniform time warping (UTW), will be useful to simply length match motion sequences. However, UTW can be used as a major building block in piece-wise or non-linear warping methods.

Given a motion matrix $\mathcal{D} = [\boldsymbol{\theta}_1\, \boldsymbol{\theta}_2 \ldots \boldsymbol{\theta}_n]$ with n degrees of freedom (DOFs), the ith joint angle trajectory $\boldsymbol{\theta}_i$ with m frames is defined by $\boldsymbol{\theta}_i = \left\{\theta_i^{(t)} : t = 1, \ldots, m \in \mathbb{N}\right\}$. Accordingly for $\boldsymbol{\theta}_i$, the uniformly warped trajectory $\boldsymbol{\theta}_{UTW,i}$ is calculated using $\boldsymbol{\theta}_{UTW,i} = \boldsymbol{\theta}_i^T \mathbf{W}$, where $\boldsymbol{\theta}_{UTW,i} = \left\{\theta_{UTW,i}^{(t)} : t = 1, \ldots, m' \in \mathbb{N}\right\}$ and $\mathbf{W}$ is the $m \times m'$ warping matrix populated using linear interpolation factors required to warp $\boldsymbol{\theta}_i$ to achieve temporal length m'. If $m' > m$, the trajectory is stretched, and where $m' < m$ the trajectory is compressed. In addition to linear interpolation, non-linear methods can also be used for calculating $\mathbf{W}$. A sample compressing and stretching is illustrated in Fig. 1 where linear interpolation is used.

Our proposed method uses UTW in two different stages. First, CoTW linearly warps the input trajectory using UTW to length match the trajectory with respect to the reference. The input is then divided into a number of equal segments. Accordingly, $\boldsymbol{\theta}_i$ is rearranged as $\boldsymbol{\theta}_i = \left[\theta_i^{(1:\lambda)}\, \theta_i^{(\lambda:1+2\lambda)} \ldots \theta_i^{(c\lambda+1:m)}\right]^T$ where we have c segments, each with a length of $\lambda \in \mathbb{N}$. Given $\boldsymbol{\theta}_i$ with a length of m, the number of segments is calculated using $c = m/\lambda$, $c \in \mathbb{N}$.

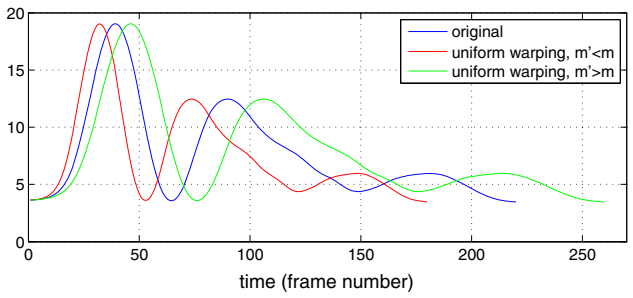

Fig. 1 Linear stretching and compressing of a motion trajectory

In addition to the segment size, a different parameter called slack size is introduced. We denote slack size by $\delta \in \mathbb{N}$. This parameter determines how much each segment is permitted to warp in either direction. In other words, each segment of $\boldsymbol{\theta}_i$ will have a temporal length in the range of $[\lambda - \delta, \lambda + \delta]$ after warping. Specifically, segment i is warped by the number of time instances η_i, where $|\eta_i| < \delta$. Accordingly, for assigned λ and $\eta_{1:c}$, the input $\boldsymbol{\theta}_i$ is warped using:

$$\boldsymbol{\theta}_{CoTW,i} = \left[\theta_i^{(1:\lambda)}\mathbf{W}_1\, \theta_i^{(\lambda:1+2\lambda)}\mathbf{W}_2 \ldots \theta_i^{(c\lambda+1:m)}\mathbf{W}_c\right]^T, \quad (1)$$

where $\mathbf{W}_1$ to $\mathbf{W}_c$ are warping matrices with dimensions $\lambda \times (\lambda + \eta_i)$. The entries of $\mathbf{W}_i$ are populated with values required to linearly warp the designated segment to match the required length. We combine the segment-wise warping matrices to create the $m \times m$ global warping matrix:

$$\mathbf{W}_g = \begin{bmatrix} \mathbf{W}_1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \mathbf{W}_c \end{bmatrix}, \quad (2)$$

where $\boldsymbol{\theta}_i^T \mathbf{W}_g$ warps the entire input $\boldsymbol{\theta}_i$ in segment-wise fashion. Since the input has already been length matched with the reference, CoTW only allows combinations of warping degrees that result in the output having the length of m, in other words $\sum_{i=1}^{c} \eta_i = 0$.

Figure 2 illustrates the process where the original input is first uniformly warped (UTW) and length matched with the reference. The input is then divided into a number of segments. Each segment is warped by $\eta_i < \delta$. The objective function, which is described in the following sections, utilizes segments of the input and corresponding segments of the reference, and optimizes the set of η_i. While the optimization process is carried out linearly, as each sequence is sliced into a number of segments, and each segment is warped separately and differently, a global non-linear warp is achieved. The amount of non-linearity can simply be increased if the length of segments (λ) is decreased and different warps (η) are utilized. A schematic of the entire warping system and its different components is illustrated in Fig. 3. Other components of the system are described into the following sections.

3.2 Objective function

Most existing warping techniques such as DTW and CTW utilize distance-based objective functions. While distance is often used to quantify similarity between sequences, additional steps are sometimes taken to ensure such measures correctly represent similarity in motion data [30,48]. In fact,

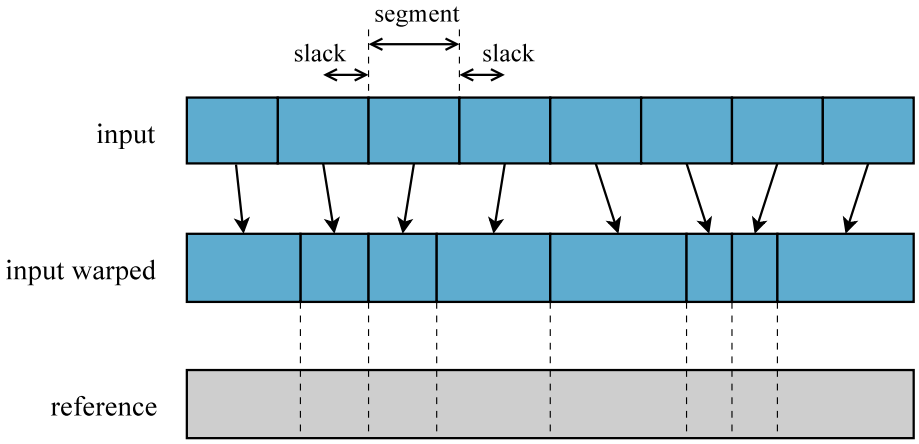

Fig. 2 An input trajectory is divided into a number of segments. Each segment is allowed to warp, using UTW, by a bounding slack parameter. Warping is carried out with the aim of achieving maximized correlation with respect to the corresponding segments of reference

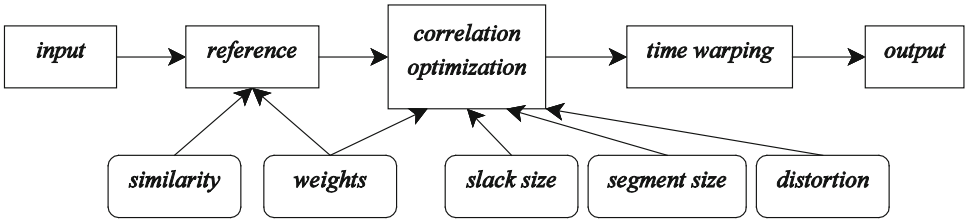

Fig. 3 The overall schematic of the system is presented

it has been previously observed that distance-based measurements alone are not necessarily the best indicators of perceptually similar motion sequences [49]. Such objective functions analyze the proximity of entries of two trajectories at corresponding time instances (frames). Nevertheless, humans are known to look for shapes, forms, and patterns rather than investigate individual entries [50,51]. Accordingly, this property needs to be taken into account when aiming to achieve perceptually sound processed motion trajectories.

From a computational perspective, let us assume two hypothetical joint angle trajectories $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ with lengths m. Using two typical distance-based objective functions such as $\|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|_2$ or $\sum_{i=1}^{m} \left|\theta_1^{(i)} - \theta_2^{(i)}\right|$, two identical trajectories with only one or few relatively distant entries can result in an average distance identical to two trajectories where all of their entries are different (different shapes). However, semantically and perceptually, the first two trajectories are more similar than the latter. Fig. 4a shows an example where the pair shown in the bottom is more similar than the pair at the top, while the distances of the two pairs are similar. To illustrate another computational shortcoming, let us assume two identical trajectories, to one of which noise is introduced. If the magnitude of this noise is relatively small with respect to the magnitude of the trajectory, the overall shape of the trajectory will not be affected and the two trajectories remain relatively similar. Meanwhile, a distance-based analysis will point to the two being quite dissimilar. Fig. 4b illustrates the situation where a calculated distance is relatively large, but the two trajectories are perceptually similar. This can be seen as an extension to the previous case. Finally, given two identical trajectories, one of which is spatially shifted, a distance-based measurement will indicate dissimilarity (based on the magnitude of the offset). Nevertheless, the two trajectories, especially from a motion perspective, as well as from a perceptual standpoint, are identical. Fig. 4c illustrates this situation where a relatively large distance is computed, whereas the trajectories are identical in shape and form.

Different approaches have often been proposed to overcome the limitations of Euclidean distance. For example, towards human motion tracking and segmentation, Wang et al. [52] utilized orientation information alongside distance for path optimizing in spatiotemporal domain. Based on earlier arguments, and similar to [10], we suggest and utilize Pearson's linear correlation coefficient (PCC), which is a numerical determinant of dependence of two variables or how similar the shapes of two trajectories are. We derive an objective function based on PCC as the means of quantifying alignment. Specifically, for two motion trajectories $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ with m frames, the correlation coefficient is calculated as $\rho(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = cov(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)/\sqrt{(var(\boldsymbol{\theta}_1)var(\boldsymbol{\theta}_2))}$. The objective function is based on ρ as given by:

$$\rho(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \frac{\sum_{t=1}^{m}(\boldsymbol{\theta}_1^{(t)} - \mu_{\boldsymbol{\theta}_1})(\boldsymbol{\theta}_2^{(t)} - \mu_{\boldsymbol{\theta}_2})}{\sqrt{\sum_{t=1}^{m}(\boldsymbol{\theta}_1^{(t)} - \mu_{\boldsymbol{\theta}_1})^2 \sum_{t=1}^{m}(\boldsymbol{\theta}_2^{(t)} - \mu_{\boldsymbol{\theta}_2})^2}}, \quad (3)$$

where μ represents the mean.

Revisiting the three cases where distance failed to represent similarity, we observe that in Fig. 4a PCC calculates $\rho = -0.0587$ for the top pair, which indicates dissimilar

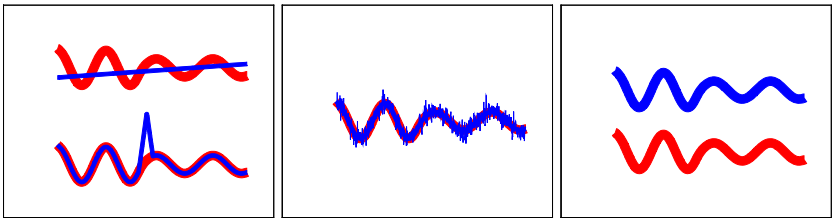

Fig. 4 Three hypothetical situations where the distance between the two trajectories fails to indicate similarity. In **a**, from a shape and form standpoint, the pair at the bottom is more similar than the pair at the *top*. However, the norm-2 distances between trajectories in each pair are equal. In **b** relatively large distance is calculated for the pair, while they are quite similar in shape. The *blue* trajectory is the noisy version of the *red* one. Finally, in **c** the distance between the pair is quite large given an added spatial offset. The two trajectories, however, are identical

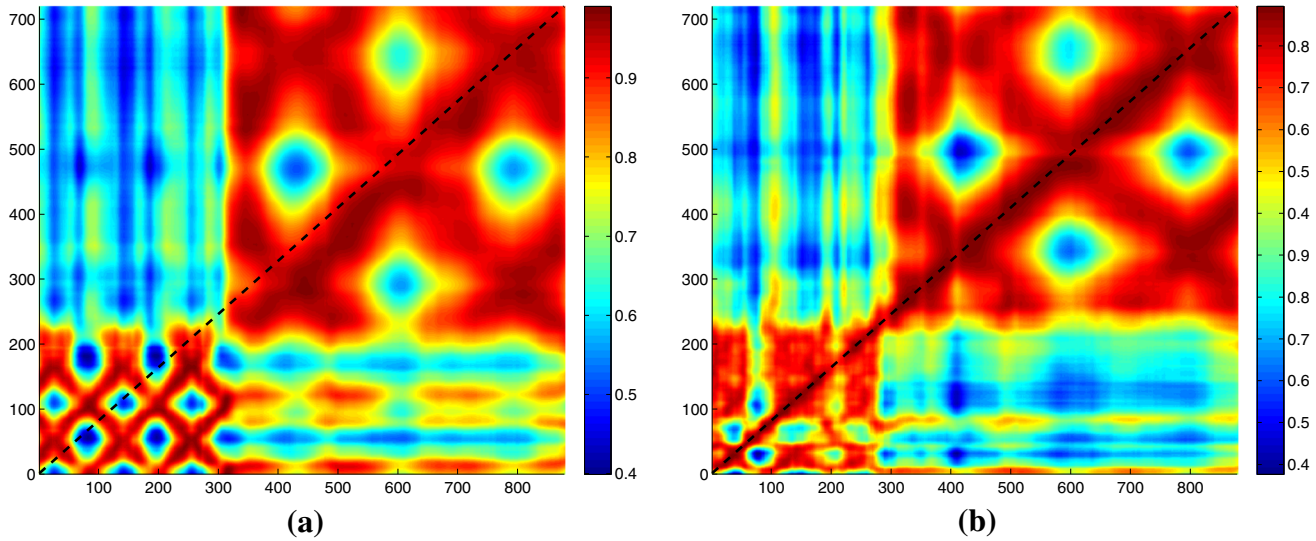

Fig. 5 Correlation matrices between two sequences of motion before **a** and after **b** warping. Each entry is the correlation value between postures of the two sequences at a given frame. Increased correlation entries on the *diagonal line* indicate that alignment increases correlation values between corresponding frames

trajectories, while for the bottom pair $\rho = 0.8231$, indicating that the two are relatively similar. For Fig. 4b, $\rho = 0.8231$, again pointing to the two trajectories being similar, while distance measurements will calculate large values. Finally in Fig. 4c, the calculated distance is very large while PCC calculates $\rho = 1.0$, pointing to identical shapes.

Another significant advantage of using a correlation-based function is that PCC is a normalized value (maximum = 1). Therefore, the calculated results for different motion representations such as Euclidean displacement vectors and joint angle trajectories will be comparable and can fit the same framework. This cannot be said about warping methods that utilize distance-based objective functions.

Figure 5 illustrates the correlation matrices between two sequences of motion before (a) and after (b) alignment. In these matrices, each entry is the correlation between postures of the two sequences at given time instances. The diagonal line denotes a one-to-one correspondence between postures of the two sequences at the corresponding frames. Evidently, the sum of all correlation values between corresponding postures significantly increased after alignment. This is a testament that a correlation-based objective function can accurately represent motion alignment.

Our objective function calculates PCC for each segment of the length-matched input trajectory with respect to its corresponding section from the reference. The goal is to calculate the set of $\boldsymbol{\eta}$ that maximizes this function. For warping multi-dimensional data, one approach is to warp each DOF separately. Using CoTW, despite being bound by δ, each segment will be warped by a different warping degree. As a result, synchronization between different DOFs of the motion sequence will be lost. To prevent this from happening, we calculate and combine the objective function for each of the n DOFs. Accordingly, for length-matched input and reference motion data, $\boldsymbol{\mathcal{D}}_{in}$ and $\boldsymbol{\mathcal{D}}_{ref}$, maximizing objective function,

Table 1 The dynamic programming optimization used in CoTW

```
F ← −∞ × 1_{(c+1)×(m′+1)}
F(c + 1,1) ← 0
for i = c to 1
    a ← max(i × (λ − δ), (c − i) × (λ + δ))
    b ← min(i × (λ + δ), (c − i) × (λ − δ))
    for j = a to b
        for η = − δ to + δ
            v ← F(i + 1, j + λ + η) + J(D_input, D_reference|s_i)
            if v > F(i, j)
                F(i, j) ← v
                U(i, j) ← η
            end
        end
    end
end
Y(1) ← 1
for i = 1 to c
    Y(i + 1) ← Y(i) + λ + U(i, Y(i))
end
```

$$J_u = \sum_{i=1}^{n} \rho\left(\mathcal{D}_{in,i}, \mathcal{D}_{ref,i}|s_k\right), \quad \text{for } k = 1:c, \tag{4}$$

results in a uniform warping of all DOFs, where i represents the DOF, s the segment, and k the segment number.

Using Eq. (4), all DOFs maintain uniform and equal significance in the overall warping of the sequence. Incorporating a weight parameter in the objective function will result in alignment of motion sequences with more emphasis on particular DOFs. For example, warping with the aim of only aligning the arms/hands (and not the head or the feet) could be carried out using a weighted sum of ρ, where the weights of joints other than arms/hands are set to zero. Accordingly, Eq. (4) will be updated as:

$$J = \sum_{i=1}^{n} w_i \cdot \rho(\mathcal{D}_{in,i}, \mathcal{D}_{ref,i}|s_k), \quad \text{for } k = 1:c, \tag{5}$$

where w_i is the weight associated with the ith DOF, and $\arg\max_{\boldsymbol{\eta}} J$ calculates the set of segment-wise warping degrees $\boldsymbol{\eta}$.

Ad hoc means can be used to determine the weights, w_i. In gaits, for example, the importance of fingers and toes is significantly less than that of other limbs and joints. In sign language, on the other hand, the fingers are of critical importance. These considerations, among others, can be integrated into the process using the weight parameter. In the Results section, we suggest and test weights for aligning different regions of interest. Previous studies such as [53] have investigated the relative importance of different joints in animation of human motion, which can provide suitable guidelines for tuning w_i.

3.3 Optimization

Given length-matched input and reference motions $\mathcal{D}_{in}$ and $\mathcal{D}_{ref}$, and a set of assigned warping parameters (λ, δ), the optimal warped output sequence needs to be computed. Accordingly, the optimal warping degree (η) for each segment needs to be calculated and utilized. In other words, all possibilities of η within $-\delta \leq \eta \leq \delta$ must be investigated for each segment. Dynamic programming is often used to solve problems that would require a very large number of iterations if it were to be solved exhaustively. We use a 2-step backward–forward dynamic programming algorithm. The algorithm is derived from [10] and necessary modifications have been made based on descriptions provided earlier in this paper. The pseudo code of the algorithm is presented in Table 1. In this algorithm, c is the number of segments, m' is the new segment length successive to the initial UTW of the input which results in the length-matched version, F is a matrix which is populated using cost function values, v is the sum of objective function values, U is the lookup matrix containing the parameter values, and Y is the solution matrix. In this algorithm, all possible positions of a given segment are first inspected based on possible orientations of previous segments and the optimum alignment is calculated. Iteratively, permutations that result in suboptimal alignment are ruled out. As a result, the process always finds the set of warping degrees that best align the trajectory with respect to the reference.

4 Parameters and distortion

In this section, the impact of segment and slack sizes on the warped outputs are investigated. Given m' and λ, after dividing the trajectory into c segments, an extra segment may remain with the length of $m' - m'/\lambda \times \lambda$. This situation occurs

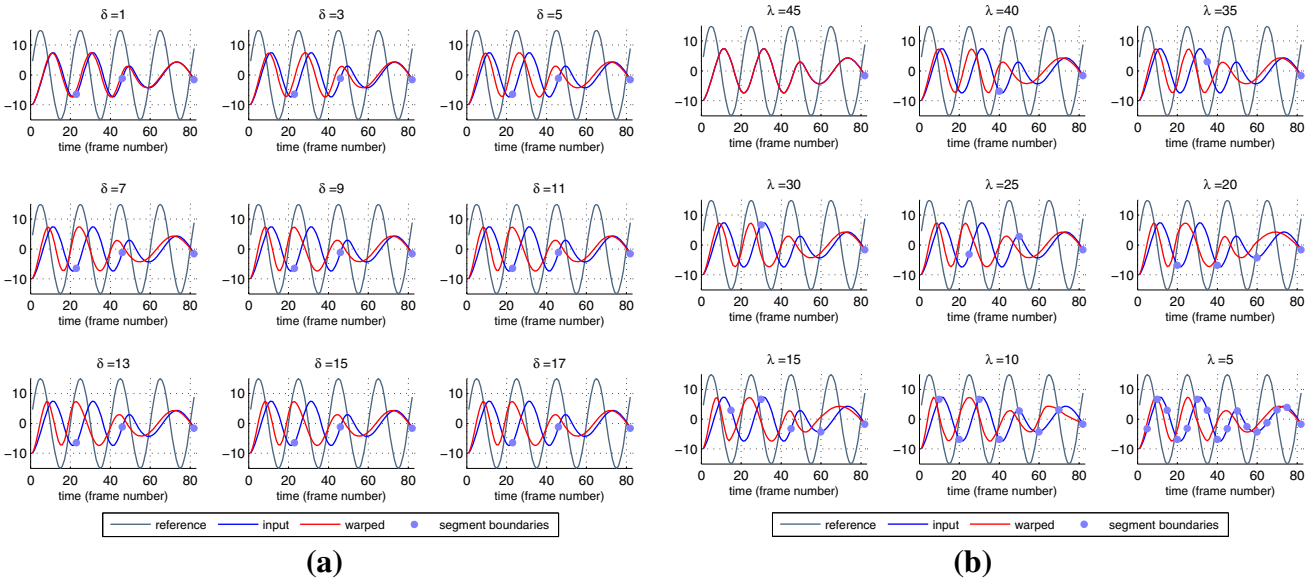

Fig. 6 CoTW outputs for **a** $\lambda = 23$ and different values of δ; **b** different values of λ and $\delta = \lambda - 4$

when $m/\lambda \neq m/\lambda$. There are two possible approaches for dealing with this residual segment: (a) counting it as a separate segment or (b) adding it to the last segment, making the cth segment a bit longer. Using the first approach, the length of the residual segment may become considerably smaller than other segments or the slack for that matter. In this case, warping it by larger values of η may cause significant distortion. We therefore use the second approach.

Figure 6a illustrates warping of a trajectory with $\lambda = 23$ and different δ sizes. The reference is a sinusoid and the input is a sum of sinusoids. The boundaries of segments are displayed. Misalignment is seen as the peaks and valleys of reference and input occurs at different time instances. It is evident that for $\delta = 1$, warping is minimal. As δ increases to $\delta = 7$, the alignment improves. Beyond this value, however, there is no significant change in alignment. This is because, in this particular case, optimum warping occurs at $\eta = 7$; therefore, alignment remains unchanged for $\delta > 7$. Thus, we suggest that when manually tuning the warping parameters, it would always be safe to set δ to the maximum possible length. The method's boundary conditions do not allow $\delta > \lambda - 4$ and, so, the maximum possible length of slack is $\delta = \lambda - 4$. Using the maximum δ, however, despite resulting in optimum warping, may not always be suitable as it can cause distortion. The notion of distortion is described later in this section.

With regard to the segment length, let us initially assume that the entire trajectory is one segment, meaning $\lambda = m'$. This yields that the trajectory is not permitted to warp, since the post-warp length must remain unchanged. Moreover, for $m'/2$, as discussed earlier, the residual segment will be added to its previous segment. Therefore, we conclude that only segment lengths of $\lambda \leq m'/2$ will result in practical warping. As λ decreases, the process will have more segments to warp to achieve greater correlation. Figure 6b illustrates the effect of segment size where for $\lambda = 45 > m/2$, no warping occurs. As the number of segments increases (λ decreases), alignment is improved. However, since decreasing the segment length results in constraining δ and hence η, it does not always result in better alignment. For instance, in this example, the best alignment is achieved for $\lambda = 10$ where the fourth local maximum of the input is aligned with that of the reference.

To select λ and δ, we suggest an iterative exhaustive optimization problem, where for all combinations of λ and δ, the correlation of the output is calculated with respect to the reference. To further integrate the notion of joint customization, a weighted sum of DOF correlations, similar to Eq. (5), can be utilized. This step results in a correlation matrix $\boldsymbol{\rho}$ as illustrated in Fig. 7. The figure depicts the correlations of an actual joint rotation trajectory from a tired walk aligned with respect to an energetic walk. Here, the parameters resulting in the highest peak, $\lambda = 16$ and $\delta = 12$, achieve the best alignment.

Although maximum correlation can be achieved using the optimized parameters, the output motion could be distorted due to excessive warping. The algorithm by design employs two constraints with regard to the amount of warping applied to segments: (a) the sum of warping degrees of all segments is equal to zero, since the length of the motion will remain unchanged due to the initial length matching ($\sum_{i=1}^{c} \eta_i = 0$); (b) the slack parameter bounds the warping degree of each segment ($\eta \leq \delta$). Nevertheless, should δ be relatively large, it is possible for a segment to be warped excessively. This can cause the motion to seem unnaturally slow or fast in the dura-

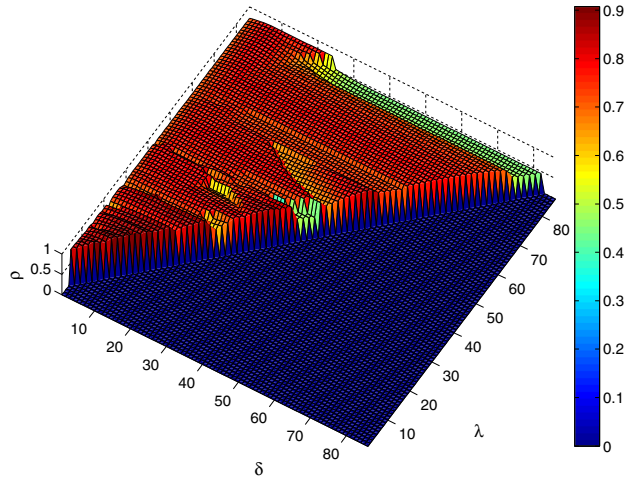

Fig. 7 Correlation matrix for permutations of λ and δ

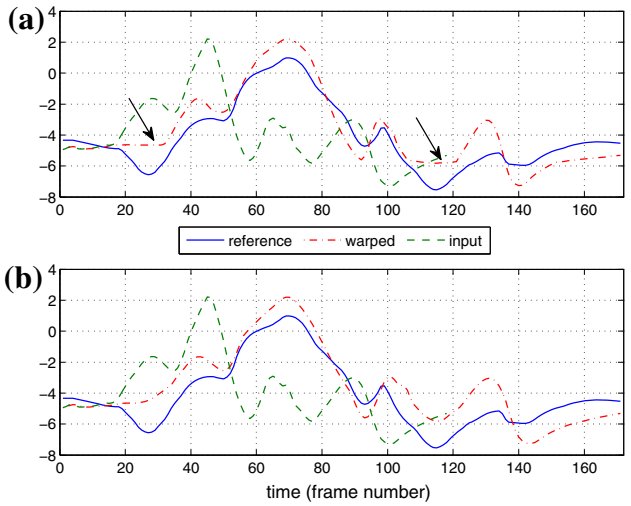

Fig. 8 A motion trajectory warped without **a** and with **b** taking the distortion factor into account

tion of that segment, as the slope of the trajectories would be affected significantly. An example of this is illustrated in Fig. 8 (top) where the arrows point to two such instances. These and other artifacts have been points of concern in the literature as well. For example, in video warping and retargeting, to avoid issues such as waving and squeezing artifacts, Wang et al. [54] performed optimized cropping and warping using motion information and distortion minimization.

To measure this distortion, the inverse of the signal to noise ratio (SNR^{-1}) can be utilized as $\text{SNR}^{-1} = \frac{P_{\text{noise}}}{P_{\text{signal}}}$, where P represents the classical definition of power. In the context of this study, for input sequence $\boldsymbol{\mathcal{D}}_{in}$ which is length matched with the reference, this definition will translate to:

$$\text{SNR}^{-1} = \frac{\|\boldsymbol{\mathcal{D}}_{in,CoTW} - \boldsymbol{\mathcal{D}}_{in}\|}{\|\boldsymbol{\mathcal{D}}_{in}\|}, \tag{6}$$

where $\boldsymbol{\mathcal{D}}_{in,CoTW}$ is the same sequence after CoTW warping. Note that we use the length-matched version of the input sequence for computing the noise. This is to calculate the distortion caused by the non-uniform changes in the sequence rather than the changes that occur as a result of the initial length matching. Eq. (6) indicates that to acquire minimum SNR^{-1}, $\boldsymbol{\mathcal{D}}_{in,CoTW}$ must approach $\boldsymbol{\mathcal{D}}_{in}$. As a result, minimizing this measurement during the warping process tends to cancel out the attempts made to carry out the non-uniform segment-wise warps. Moreover, our experiments showed that when SNR^{-1} with the described definition is minimized during the warping process, almost no warping occurs. As a result, we investigated with several variations of Eq. (6) to find a practical substitute for measuring distortion. For example, we used first and second derivatives of the components in Eq. (6). We also experimented with using the difference of the norms instead of the norm of differences. Eventually, we concluded that the following, which we denote by h, is the most suitable means of quantifying distortion without countering the warping process:

$$h = \left(\left\| \frac{\Delta \boldsymbol{\mathcal{D}}_{in,CoTW}}{\Delta t} \right\| - \left\| \frac{\Delta \boldsymbol{\mathcal{D}}_{in}}{\Delta t} \right\| \right) . \left\| \frac{\Delta \boldsymbol{\mathcal{D}}_{in}}{\Delta t} \right\|^{-1} . \tag{7}$$

This measurement entails that distortion is minimized when the norm of the slopes of $\boldsymbol{\mathcal{D}}_{in,CoTW}$ and $\boldsymbol{\mathcal{D}}_{in}$ converge. In other words, minimizing $\boldsymbol{h}$ prevents $\boldsymbol{\mathcal{D}}_{in,CoTW}$ from having drastic changes in slope, and hence less distortion. Similar to the $\boldsymbol{\rho}$ matrix, the distortion caused by different combinations of λ and δ can populate a matrix, which we denote by $\boldsymbol{h}$. A weighted sum of the distortions of the DOFs can be used. Accordingly,

$$\arg\max_{\lambda,\delta} \boldsymbol{\rho} - \alpha \cdot |\boldsymbol{h}| \tag{8}$$

calculates the optimum λ and δ for warping the input where α is a weight factor for distortion. For applications where reducing the distortion is more central, larger values of α are used. As the trade-off for minimization of distortion, suboptimal parameters are used, and the best possible alignment may not be achieved. Nevertheless, the warped motion appears more natural. Figure 8 (bottom) presents CoTW warping, taking $\boldsymbol{h}$ into account with $\alpha = 0.5$. We observe that the two instances of distortion are prevented. Yet on the other hand, suboptimal alignment is achieved. For example, distortion minimization counters the alignment of the local minima at the 100th frame, where they were previously aligned with $\alpha = 0$. Figure 9a shows the distortion matrix ($\boldsymbol{h}$) for the same trajectories used in Figs. 7 and 8. Figure 9b illustrates the correlation matrix with minimized distortion ($\boldsymbol{\rho} - \boldsymbol{h}$).

In addition to calculation of the optimum parameters, λ and δ can be manually tuned. In many cases, the input sequence is composed of multiple actions and the goal of warping is to align each action with the corresponding action

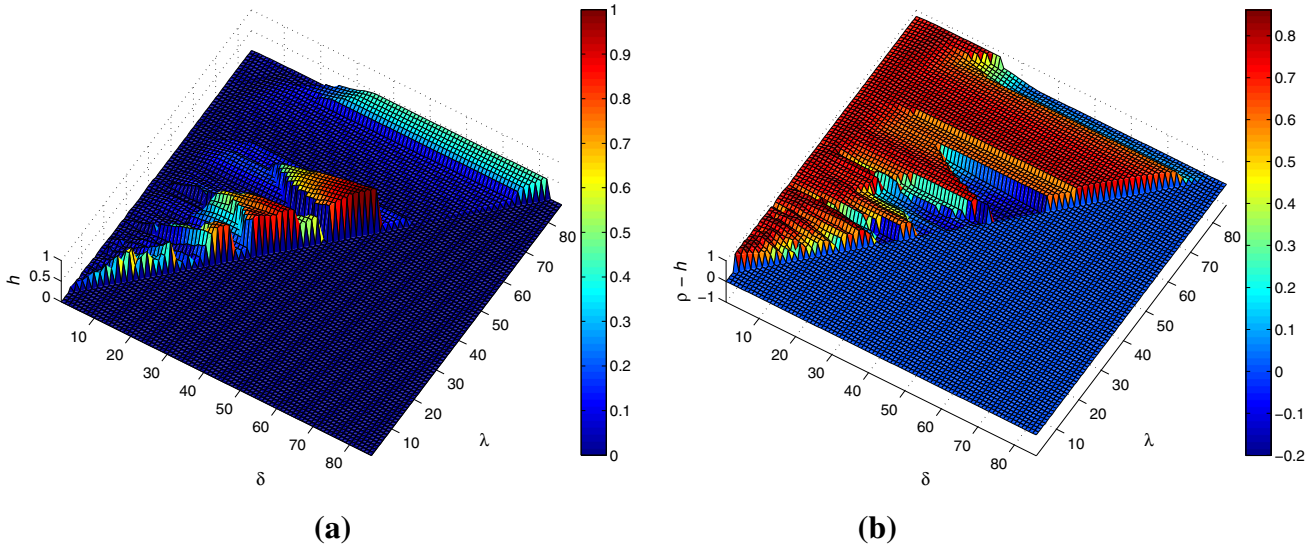

Fig. 9 Distortion (**a**) and subtraction of distortion from correlation (**b**) matrices for permutations of λ and δ

from the reference. In such cases, the number of actions in the input sequence can be a good determinant of the number of segments. For example, assuming the sequence is composed of two jumps and a kick, the number of segments can be set to three. In other cases, the goal of warping may be to align sub-actions (sub-components of actions). In this case, the number of segments can be determined by the number of these components. For example, for a sequence consisting of a single jump, the number of segments can be set to two. Here, segment 1 will correspond to the first half of the jump where the actor leaves the ground and reaches the maximum height of the jump. Segment 2 will then correspond to the second part of the jump during which the actor lands. This approach can significantly increase runtime due to leaving out the time-consuming exhaustive search. Moreover, the user can exercise this option to customize the warping process based on particular applications.

5 Automatic reference selection

When warping two sequences, the selection of the reference depends on the application. In the event that multiple sequences need to be aligned, for example when training a classifier, selection of the reference trajectory can be a difficult and influential issue. Furthermore, in multi-dimensional data, warping each DOF individually and with respect to a separate reference is one possible approach. Using DOFs of different sequences as references will cause the warped sequence to lose its synchronization among joints. In such cases, each DOF of the warped sequence will seem to be moving independent of others. To avoid this artifact, we combine the information regarding the different DOFs of all sequences to select a single sequence as reference. In other words, we select a single sequence and use all its DOFs as references for the respective DOFs of all other sequences.

To select a reference sequence, several approaches are possible. In addition to manual selection, which can be a valid approach based on the application, other techniques can be used. Skov et al. [46] mention several methods for this purpose. We utilize a similarity index. For a set of sequences, $\mathcal{F}$, where $\boldsymbol{\mathcal{D}}_{i,j}$ is the ith DOF of the jth length matched sequence, and w_i is the weight set described earlier, we calculate the similarity index through:

$$\mathcal{P} = p \sum_{i=1}^{n} \left(w_i \prod_{j \in \{\mathcal{F}\}-r} \left| \rho(\boldsymbol{\mathcal{D}}_{i,r}, \boldsymbol{\mathcal{D}}_{i,j}) \right| \right). \tag{9}$$

Accordingly, $\arg\max_r \mathcal{P}$ determines the sequence that is most similar to all other sequences. Using this sequence will decrease the need for warping, potentially decreasing the imposed distortion. Note that we blended the similarity indexes of different DOFs using the same weight set described in previous sections to take into account the significance of each DOF.

We investigate the effectiveness of this index by comparing it with other techniques for selecting a reference. These methods include using the average trajectory, median, maximum of all, and minimum of all. Figure 10 presents motion trajectories from the xth axis of the right foot from 15 male walkers. To add differently shaped trajectories, we included heavily smoothed versions of three of these trajectories as well. Smoothing was carried out using low-pass filtering. The trajectories computed as reference are shown in red for each method. It is observed that the mean, median, and similarity index select the smoothest references, while min-

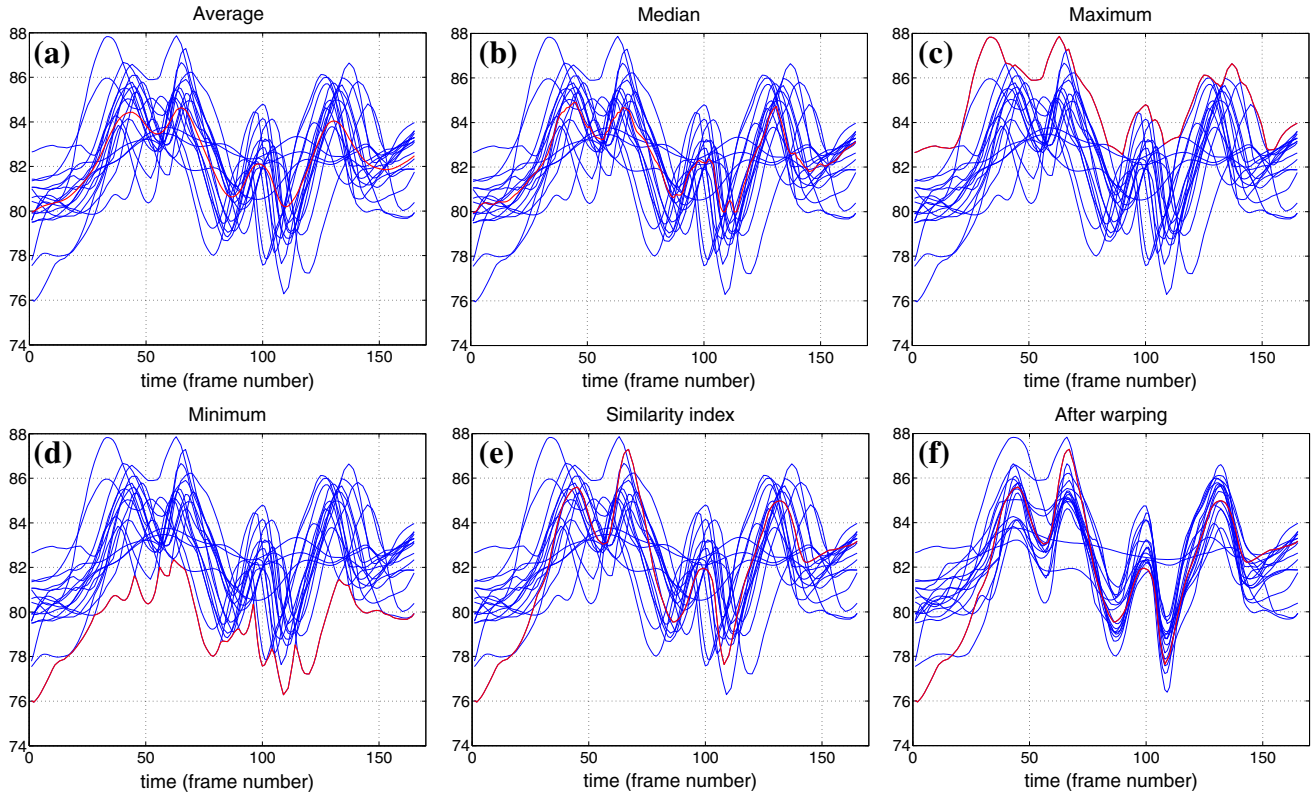

Fig. 10 Different methods of determining a reference when multiple trajectories are available. The first five figures show the different possible methods, one of which is the proposed similarity index. In each case, the calculated reference is shown in *red*. The last figure presents the trajectories after warping (CoTW) with respect to the reference selected using the proposed similarity index

imum and maximum methods select trajectories with many local extrema. It is generally preferred that the reference be smooth, as well as being one of the *original* trajectories rather than a calculated trajectory based on others (mean or median). Figure 10 also illustrates all the trajectories warped with respect to the reference selected using the proposed similarity index.

6 Results and discussions

6.1 Performance and alignment

The proposed algorithm is implemented in MATLAB. To subjectively and quantitatively evaluate the results, other popular warping techniques, namely UTW, DTW, DDTW, and CTW are used. We use motion capture data from the Carnegie Mellon University motion dataset (http://mocap.cs.cmu.edu/). Six different action classes are used. Complex actions such as climbing a ladder, dribbling a basketball, side twists, and kicking a ball as well as more simple actions such as jumping and walking are utilized. In most cases, the number of steps or actions in the input and reference vary, making it a relatively difficult alignment problem. The parameters have been optimized as discussed earlier. The mean λ is 25 frames and the standard deviation is 13. δ is always set to $\lambda - 4$.

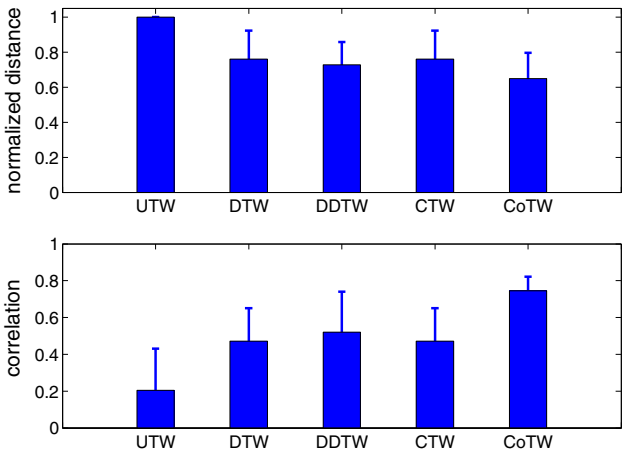

Fig. 11 Means and standard deviations of normalized distance and correlations for different actions warped using UTW, DTW, DDTW, CTW, and CoTW

Figure 11 illustrates how the absolute distances per frame per DOF compare for different warping methods and actions. Since for different sequences, the range of distances can significantly vary, they are normalized with respect to the

maximum distance for each action class. In all cases, the maximum distance was observed for UTW; hence, all UTW-normalized distances are equal to one. Generally, DTW, DDTW, and CTW are not too different in terms of distance, while CoTW shows reduced distance. Similarly, for correlation, CoTW outperforms the others, followed by DDTW, CTW, and DTW. To further investigate the results, we performed post hoc analysis using t test between the correlations and normalized distances achieved using CoTW vs the other methods. For normalized distance, CoTW (M_{CoTW} = 0.65, SD_{CoTW} = 0.15) is significantly superior to UTW (M_{UTW} = 1, SD_{UTW} = 0), at the $p < 0.001$ level with $t(10)$ = 5.86. While CoTW is also superior to DTW (M_{DTW} = 0.76, SD_{DTW} = 0.16), DDTW (M_{DDTW} = 0.73, SD_{DDTW} = 0.13), and CTW (M_{CTW} = 0.76, SD_{CTW} = 0.16), the reduced distance is not statistically significant at the $p < 0.05$ level. For increased correlation, however, CoTW (M_{CoTW} = 0.75, SD_{CoTW} = 0.08) is significantly superior to UTW (M_{UTW} = 0.20, SD_{UTW} = 0.23) at $p < 0.001$ with $t(10) = 5.53$, DTW (M_{DTW} = 0.47, SD_{DTW} = 0.17) at $p < 0.01$ with $t(10)$ = 3.46, DDTW (M_{DDTW} = 0.52, SD_{DDTW} = 0.22) at $p < 0.05$ with $t(10) = 2.37$, and CTW (M_{CTW} = 0.48, SD_{CTW} = 0.18) at $p < 0.01$ with $t(10) = 3.45$.

In addition to DTW, DDTW, and CTW, as mentioned in Sect. 2, GTW is another relatively recent technique for warping. However, the method is still under development, and unresolved converging and local minima issues persist. Most importantly, the number and types of basis functions (monotonically increasing base sets), using which warping is performed, need to be selected manually. The basis functions significantly impact the warping output, and as a result a comparison between our method and GTW would not provide an accurate account of the relative performance of CoTW vs GTW. We tried several combinations of basis functions for GTW. Accordingly, the best performance resulted in a post-warping normalized distance of approximately 0.8 and correlation of approximately 0.4, meaning that CoTW has significantly better performance. Nonetheless, automated learning of the basis functions may result in more accurate results for the GTW.

Video clip 1 accompanying this manuscript shows the actual warping of the sequences using different methods. In general, DTW, DDTW, and CTW work well for aligning sequences. CoTW, however, shows more accurate alignment. In Fig. 12, we illustrate the performance of CoTW, where accurate alignment is achieved. In the climb ladder sequences, the input contains two extra steps. Interestingly, in the warped output, the character first climbs in an aligned fashion with respect to the reference, and then to make up for the extra steps, intensely slows down, almost to the point of stopping. The stop goes on for the duration of the excess steps in the reference. The climb then resumes after this correction. While this may seem abnormal at first glance, it is in fact the best possible solution for such cases where the number of actions differs. The alternative solution is for the excess steps to be spread out through the entire sequence. This would result in misalignment in all sub-actions. As seen in the video, the warped output behaves very naturally and accurate alignment with respect to the reference is achieved. The dribble sequences are composed of two different main actions, namely walking and dribbling. We observe that in the output, correction is mostly applied to the arms while the legs are only partially aligned. This is natural as uniform weights have been utilized for alignment and, as a result, the maximized correlation enforces the warping process, regardless of the spatial regions. Had the weights been selected to highlight only specific regions of the body, warping would have been focused on specific joints. We test this in the following paragraphs of this section. In the sideway twist motions, the last few frames of each twist appear to be misaligned in the output. This is not, however, actual misalignment. The two characters simply twist with different degrees. As a result, they appear to be misaligned, whereas from a relative standpoint, alignment is achieved. Moreover, it should be noted that CoTW does not alter spatial variations, rather only the timing of actions. So the turning degree of the twist cannot be altered. In the walk sequences, the first and last frames of the output and reference differ and have not been aligned. While in most examples, the input and reference motions start and end with similar poses, the walking input and reference start and end during the walk and with different poses. Referring back to the algorithm and Fig. 6 as an example, we see that the first and last frames remain temporally unchanged after warping. Accordingly, these postures cannot be aligned through the proposed method. The rest of the sequence, however, is aligned. Similar to the climb ladder, rapid changes in velocity, this time in the form of increase, are observed. This is because the input contains two extra steps with respect to the references, and to compensate faster strides are necessary.

Video clip 1 illustrates the input motions warped using UTW, DTW, DDTW, and CTW along with CoTW. While DTW, DDTW, and CTW are practical, fast, and generally accurate, for presented samples, misalignments persist. Reasons for the misalignments can be the use of distance-based objective functions as well as the complexity and non-equal number of actions in the sequences. These misaligned instances are clearly seen in the video. In Fig. 13, we illustrate sample frames where DTW, DDTW, and CTW failed, while CoTW has performed well. Moreover, it should be pointed out that in sequences such as climb ladder and walk where the number of actions significantly differs, instead of decelerating/accelerating the input to compensate for the timing differences, DTW, DDTW, and CTW use still frames where particular frames are repeated. Moreover, DTW, DDTW, and CTW produce a warped version of the reference, with respect to which the input is aligned. This can be avoided, but will

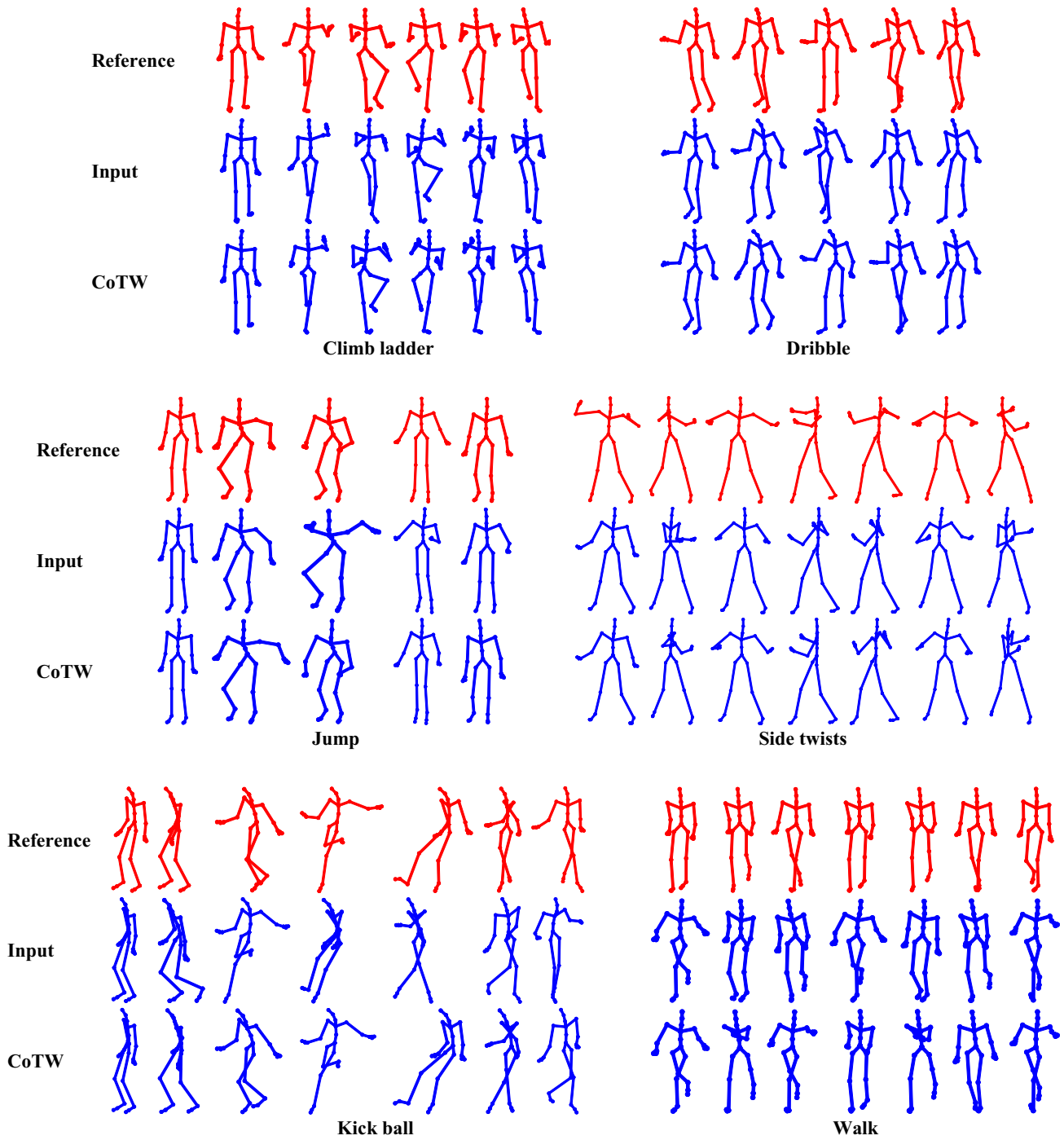

Fig. 12 Snapshots from different actions being aligned using CoTW

result in very fast jumps in the input sequence for it to catch up with the reference.

6.2 Customization

As described earlier, one of the main advantages of CoTW is its customizability. Here, we illustrate how the weight parameter can be adjusted to efficiently and accurately warp different regions of interest in the input sequence. As an example, we use dribbling, an action that is composed of two separate regions performing different actions, namely the arms dribbling and the legs walking. Ad hoc means are used to determine the weights presented in Table 2. Previous studies such as [53] have studied joints that are critical in motion animation and proposed weights for their significance, which can be utilized in our system. In

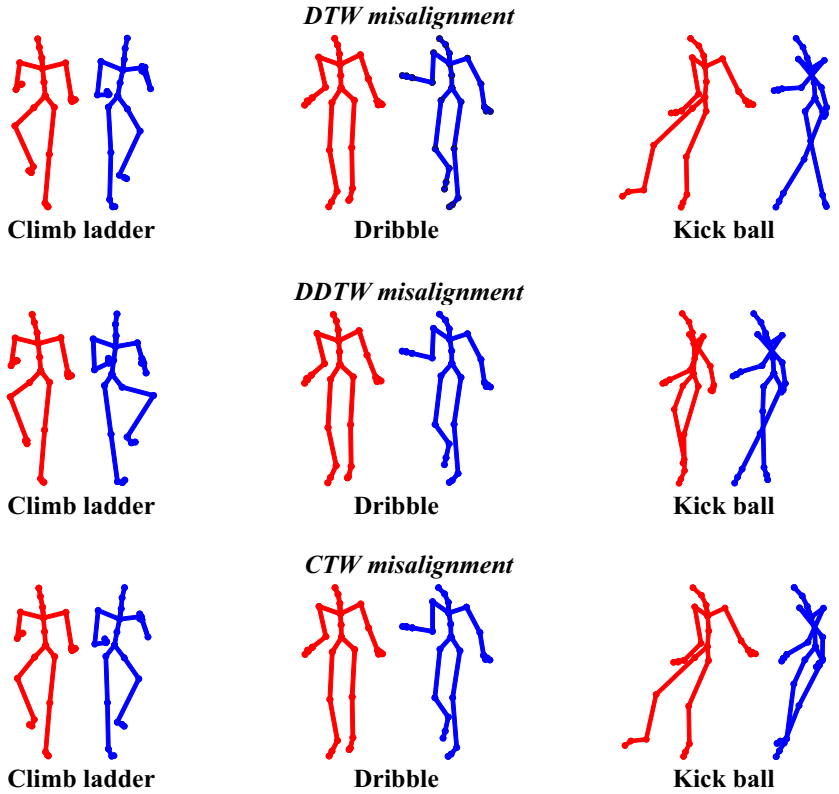

Fig. 13 Snapshots from where DTW and CTW have failed to perfectly align the sequences. As shown in Fig. 12 and in the accompanying Video clip 1, CoTW has correctly aligned these instances

Table 2 Suggested for different regions of the body

DOF	Full	Legs/feet	Arms/hands
Global displacement	1.00	1.00	0.00
Global orientation	1.00	1.00	1.00
Hips	1.00	1.00	0.00
Shoulders	1.00	0.00	1.00
Thighs	1.00	1.00	0.00
Spine	1.00	0.00	0.50
Arms	1.00	0.00	1.00
Legs	1.00	1.00	0.00
Hands	1.00	0.00	1.00
Feet	1.00	1.00	0.00
Neck	1.00	0.00	0.00
Head	1.00	0.00	0.00
Fingers	0.00	0.00	0.00
Toes	0.00	0.00	0.00

Table 2, we used the semantic names of the joints rather than the exact names used in the CMU dataset files. For example, the name "Thigh" is used instead of "UpLeg". Figure 14 and Video clip 2 illustrate that when customizing the alignment process using CoTW, different joints can be accentuated with different significance levels. When the weights are uniformly distributed, a combination of a two-region alignment is achieved, partially aligning the walk and partially aligning the dribble. When the regions of interest are the shoulder, arms, and hands, the process is performed with these regions being completely aligned while the legs are not. A similar observation is made when the regions of interest are the thighs, legs, and feet.

Another customization capacity with CoTW is manual selection of segment and slack lengths. Previously, we illustrated that when a walking sequence is warped, using the optimum λ and δ results in accurate alignment. Figure 15 and Video clip 2 illustrate that when we manually assign the segment length to that of one step, the steps are aligned with acceptable precision.

With regard to the parameter α, which determines the significance of distortion minimization, we observed that generally, values of around $\alpha = 0.5$ are suitable for maintaining a good balance between minimization of distortion and increasing alignment (correlation). Nonetheless, the assignment of α depends on the application at hand. If alignment is more critical compared to the distortion caused as a result of the warp, lower values for α should be utilized. On the other hand, if naturally appearing motion is prioritized over the accuracy of the alignment, higher α values should be used. The amount of initial misalignment between the sequences can also be a determining factor. For example, the side twist action in this paper contains significant misalignment with respect to the reference. Thus, more significant warps are needed to correct the misalignments. In cases like this, lower values of α will allow for reasonable

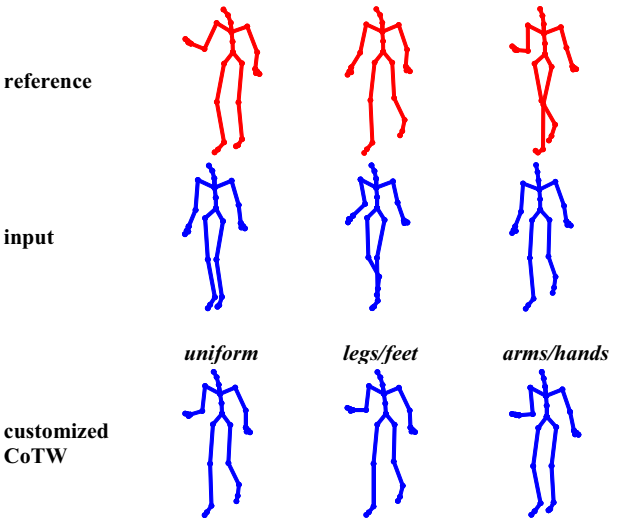

Fig. 14 Customization of CoTW is presented. The input dribbling motion is warped with uniform weights, as well as the regions of interest such as the walking or the dribbling

alignment to be achieved, while higher α values might cause the misalignments to persist.

6.3 Style translation

Style translation is the process of transferring style features from one sequence of motion onto another [2]. This procedure can often be carried out using interpolation or extrapolation [4]. However, sequences contain relative misalignments, which cause artifacts such as footskating [55] when style translation is carried out without proper alignment. As a result, style translation can be used for evaluating warping methods. We use neutral, macho, and marching style walks to perform this test. To also test the automatic reference selection, two neutral walks are employed. The similarity index selects one of the two neutral walks as the reference. The other normal walk and the macho walk are warped with respect to the selected reference. By subtracting the stylistic walks from the reference and adding the resulting style features to the other neutral walk, style is transferred from one sequence onto another. Figure 16 (from Video clip 2) illustrates frames of input (neutral) and output (macho and marching) sequences, following alignment and style translation. While it is routine to perform post-processing and cleanup successive to style translation [2,9] due to the significant footskating that is caused as a side effect of the process, our output video demonstrates very little artifacts. As a result, despite the need for cleanup, the required post-processing is not significant.

6.4 Summary of advantages

The proposed CoTW algorithm achieves robust alignment of sequences and outperforms UTW, DTW, DDTW, and CTW, especially when the number of actions in the input and reference is different. Additionally, CoTW demonstrated several advantages over existing warping techniques. While some previous works have addressed some of these issues, to the best of our knowledge, a single system that incorporates all of them does not exist. In the following, the advantages of our method are summarized:

(1) CoTW uses correlation as a measure of similarity between motion trajectories which, based on our earlier arguments, is significantly better compared to Euclidean, Manhattan, or some other distance functions that are often used. While DDTW slightly outperforms DTW and CTW due to the use of the derivative of the trajectories, our method still shows more accurate results. The reason for this is that even though differentiation will consider the trend of the trajectories rather than similarity between the trajectories themselves, the trend (differentiation) used in DDTW is instantaneous, while in CoTW, the trend (correlation) is global.
(2) CoTW can be customized both spatially and temporally. Through the former, a weight set is applied to the charac-

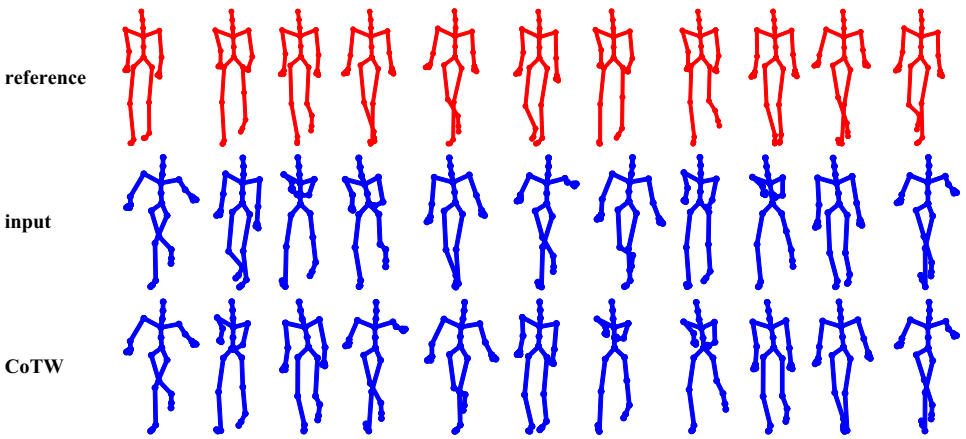

Fig. 15 Manual tuning of λ and δ results in relatively acceptable alignment. Here, λ is assigned as the approximate length of a single stride and $\delta = \lambda - 4$

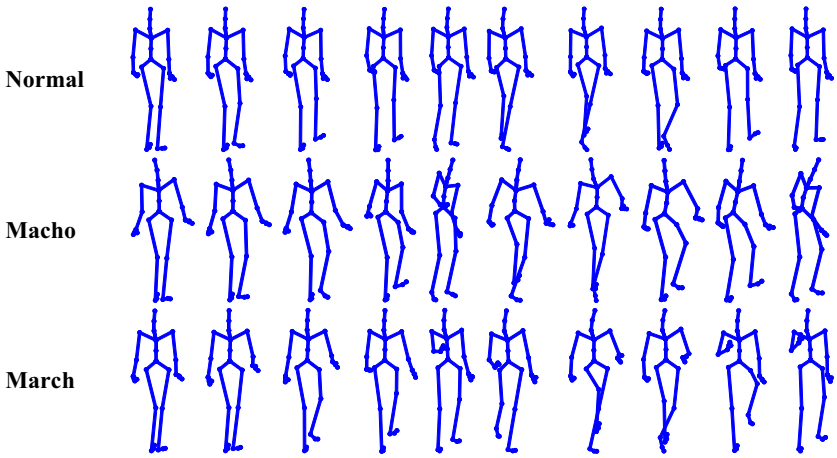

Fig. 16 Style translation where a neutral input is converted to marching and macho walks

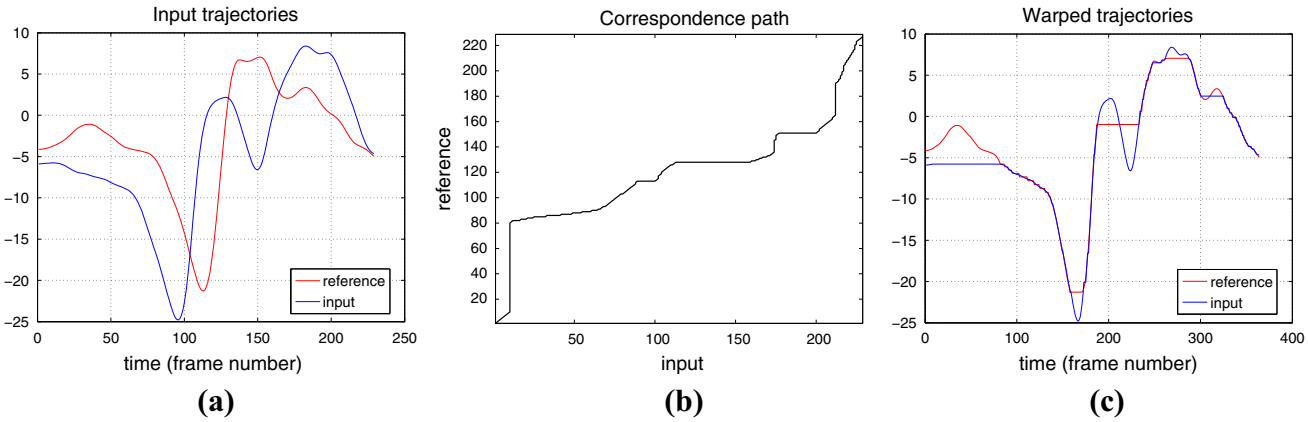

Fig. 17 Warping methods such as DTW use frame-wise correspondence and use still frames to achieve alignment. Initial trajectories are shown in **a**, the calculated alignment (correspondence) path is presented in **b**, and **c** shows the warped trajectories

ter model, resulting in more alignment in specific joints or motion DOFs. The latter allows the user to select the length of segments of the sequence that will eventually be aligned. Nevertheless, optimum segment length and slack constraints can be automatically calculated and utilized.

(3) CoTW prevents over-warping of motion that causes distortion. As a result, CoTW outputs more naturally appearing warped sequences. It should be noted, however, that other warping techniques have also enforced constraints. While these modifications have often been imposed with the purpose of increasing efficiency and speeding up the procedures, they can serve toward prevention of over-warping. For example, slope constraints [15], weight factors [7], and path constraints [7,16] have been proposed.

(4) Most techniques such as DTW, DDTW, and CTW compute an alignment path, which determines the correspondence between instances of the input and reference trajectories. Figure 17 illustrates an example. Utilizing this correspondence function for warping does not produce smooth curves. Rather, identical consecutive frames (still frames) are often used to compensate for timing where needed (see Fig. 17c). The examples of this are the vertical and horizontal paths in Fig. 17b. To remedy this, post-processing in the form of smoothing or interpolation is required. CoTW, however, produces readily smooth trajectories that can be used for animation purposes.

(5) When multiple sequences are available, the process allows for automatic selection of a reference that will most likely demand the least warping from other sequences. This process can especially be useful when aligning a dataset for different purposes, such as training a classifier. Moreover, this modality too can be customized to allow more emphasis on particular motion DOFs.

6.5 Limitations

A computational limitation of CoTW occurs when one or more segments of a motion trajectory, whether input or refer-

Table 3 Average and standard deviations of runtimes for different warping techniques. The six sequences presented earlier were used

Warping	UTW	DTW	DDTW	CTW	CoTW
Time (s)	0.02 ± 0.01	0.08 ± 0.04	0.09 ± 0.06	1.25 ± 0.59	10.01 ± 7.61

ence, have zero variance. Based on Eq. (4) and subsequently Eq. (3), the objective function cannot be calculated for zero-variance segments. This limitation does not exist in most distance-based objective functions.

Another limitation of CoTW is that based on the proposed algorithm, the first and last frame remain unchanged. To the best of our knowledge, this is the case with other existing methods such as DTW as well. Nevertheless, this property entails that should two motion sequences start or end with different poses, warping cannot correct these frames.

In Video clip 1 we see that when using CoTW, the output motion accelerates/decelerates at certain times. This is especially visible in the climb ladder and walk sequences. While such instances seem unnatural, as the number of actions (for example, number of steps in the walk) are different between reference and input, parts of the input need to be accelerated/decelerated to compensate for and correct the timing differences. Although the distortion minimization modality can reduce these changes in speed, alignment will be compromised as a trade-off.

Another limitation of the proposed method is the runtime. Since for a given segment and slack, different values of slack are evaluated, the runtime can grow rapidly, especially for large slack values. Even though dynamic programming prevents extremely long runtime, UTW, DTW, DDTW, and CTW are all faster than CoTW. Another reason for increased runtime of our method is that measuring correlation is computationally more demanding compared to measuring distance. Moreover, the exhaustive search for the optimum parameters will definitely further increase the runtime. For the six sequences presented earlier, Table 3 presents the average and standard deviations of runtimes for $\lambda = 40$ and $\delta = 20$. The results show that CoTW is the slowest among the tested approaches. The lengths of the six sequences were 310, 166, 155, 328, 228, and 102 frames.

7 Conclusion and future work

We proposed a new warping technique for aligning human motion data. The correlation-optimized time warping (CoTW) divides the input trajectories into segments, each of which is permitted to warp by a maximum of a warping degree also called slack. This warping technique uses correlation as opposed to distance for quantifying alignment. Using dynamic programming, the optimum warping degree for each segment is calculated and utilized in warping. Additionally through exhaustive means, the segment and slack parameters that maximize correlation while minimizing distortion are computed. CoTW is highly customizable with regard to regions of interest in the character model, or in other words the motion DOFs. Finally, we presented a similarity index, using which the optimum reference can be measured where multiple trajectories were available.

We illustrated the accurate and robust performance of CoTW through multiple experiments. Both distance was minimized and correlation was maximized when CoTW as compared to uniform time warping (UTW), dynamic time warping (DTW), and canonical time warping (CTW). Moreover, warped sequences were animated, visualizing the performance of CoTW. The animations showed accurate and artifact-free alignments for our test cases. Moreover, customization capabilities of CoTW were demonstrated. We performed style translation to further test the performance of CoTW. No violation of kinematic constraints, and thus no requirement for any post-processing, points to the robust performance of the proposed method.

For future work, faster computing algorithms and more efficient programming can be used. The parallel computing library in MATLAB can be utilized to reduce runtime. Lower-level programming such as MEX files in association with MATLAB, or complete C/C++ implementations, can significantly increase the speed. Finally, GPU implementations are known to speed up exhaustive search problems by more than ten times [56]. This approach can provide a practical solution for the runtime issue of CoTW, which can be explored in the near future. In addition to runtime improvement, the notion of non-equal segment lengths, which can lead to even better alignment, can be investigated in the future.

Acknowledgments This work was supported in part by the Natural Sciences and Engineering Council of Canada (NSERC) and Ontario Centers of Excellence (OCE).

References

1. Aggarwal, J.K., Ryoo, M.S.: Human activity analysis: a review. ACM Comput. Surv. **43**(3), 16 (2011)
2. Etemad, S.A., Arya, A.: Modeling and transformation of 3D human motion. In: Proceedings of 5th international conference on computer graphics theory and applications, pp. 307–315 (2010)
3. Amaya, K., Bruderlin, A., Calvert, T.: Emotion from motion. In: Proceedings of graphics interface, pp. 222–229 (1996)
4. Rose, C., Cohen, M.F., Bodenheimer, B.: Verbs and adverbs: multidimensional motion interpolation. IEEE Comput. Graph. Appl. **18**(5), 32–40 (1998)

5. Etemad, S.A., Arya, A.: Extracting movement, posture, and temporal style features from human motion. Biol. Inspir. Cognit. Archit. **7**, 15–25 (2013)
6. Etemad, S.A., Arya, A.: Classification and translation of style and affect in human motion using RBF neural networks. Neurocomputing **129**, 585–595 (2013)
7. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans. Acoust. Speech. Signal Process. **26**(1), 43–49 (1978)
8. Zhou, F., De la Torre, F.L.: Canonical time warping for alignment of human behavior. In: Proceedings NIPS, pp. 1–9 (2009)
9. Hsu, E., Pulli, K., Popovic, F.: Style translation for human motion. ACM Trans. Graph. **24**(3), 1082–1089 (2005)
10. Vest Nielsen, N.-P., Carstensen, J.M., Smedsgaard, J.: Aligning of single and multiple wavelength chromatographic profiles for chemometric data analysis using correlation optimised warping. J. Chromatogr. A **805**, 17–35 (1998)
11. Etemad, S.A., Arya, A.: A customizable time warping method for motion alignment. In: Proceedings of the 7th IEEE international conference on semantic computing, pp. 387–388 (2013)
12. Etemad, S.A., Arya, A., Parush, A.: Additivity in perception of affect from limb motion. Neurosci. Lett. **558**, 132–136 (2014)
13. Fu, A.W.-C., Keogh, E., Yung Lau, L., Ratanamahatana, C.A., Chi-Wing Wong, R.: Scaling and time warping in time series querying. Very Large Data Bases J. **17**(4), 899–921 (2008)
14. Keogh, E.J., Pazzani, M.J.: Derivative dynamic time warping. In 1st SIAM international conference on data mining. (2001)
15. Rabiner, L., Juang, B.-H.: Fundamentals of speech recognition. Prentice Hall (1993)
16. Myers, C., Rabiner, L., Rosenberg, A.: Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. IEEE Trans. Acoust. Speech Signal Process. **28**(6), 623–635 (1980)
17. Lemire, D.: Faster retrieval with a two-pass dynamic-time-warping lower bound. Pattern Recognit. **42**(9), 2169–2180 (2009)
18. Zhou, F., De la Torre, F.: Generalized time warping for multi-modal alignment of human motion. In IEEE conference on computer vision and pattern recognition, pp. 1282–1289 (2012)
19. Shapiro, A., Cao, Y., Faloutsos, P.: Style components. In graphics interface, pp. 33–39 (2006)
20. Liu, G., Pan, Z., Lin, Z.: Style subspaces for character animation. J. Vis. Comput. Anim. **19**(3–4), 199–209 (2008)
21. Heloir, A., Courty, N., Gibet, S., Multon, F.: Temporal alignment of communicative gesture sequences. J. Vis. Comput. Anim. **17**(3–4), 347–357 (2006)
22. Listgarten, J., Neal, R.M., Roweis, S.T., Emili, A.: Multiple alignment of continuous time series. In NIPS, 817–824 (2005)
23. Brand, M., Oliver, N., Pentland, A.: Coupled hidden Markov models for complex action recognition. In IEEE conference on computer vision and pattern recognition, pp. 994–999 (1997)
24. Brand, M., Hertzmann, A.: Style machines. In SIGGRAPH, pp. 183–192 (2000)
25. Taylor, G.W., Hinton, G.E., Roweis, S.T.: Modeling human motion using binary latent variables. In NIPS, pp. 1345 (2007)
26. Bruderlin, A., Williams, A.: Motion signal processing. In: Proceedings of the 22nd annual conference on computer graphics and interactive techniques, pp. 97–104 (1995)
27. Witkin, A., Popovic, Z.: Motion warping. In: Proceedings of the 22nd annual conference on computer graphics and interactive techniques, pp. 105–108 (1995)
28. Gleicher, M.: Retargetting motion to new characters. In: Proceedings of the 25th annual conference on computer graphics and interactive techniques, pp. 33–42 (1998)
29. Kovar, L., Gleicher, L.: Flexible automatic motion blending with registration curves. In: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on computer animation, pp. 214–224 (2003)
30. Kovar, L., Gleicher, M.: Automated extraction and parameterization of motions in large data sets. ACM Trans. Graph. **23**(3), 559–568 (2004)
31. Müller, M., Röder, T., Clausen, M.: Efficient content-based retrieval of motion capture data. ACM Trans. Graph. **24**(3), 677–685 (2005)
32. Müller, M., Röder, T.: Motion templates for automatic classification and retrieval of motion capture data. In: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on computer animation, pp. 137–146 (2006)
33. Zhou, F., De la Torre, F., Hodgins, J.K.: Aligned cluster analysis for temporal segmentation of human motion. In 8th IEEE international conference on automatic face and gesture recognition, pp. 1–7 (2008)
34. Kim, M., Hyun, K., Kim, J., Lee, J.: Synchronized multi-character motion editing. ACM Trans. Graph. **28**(3), 79 (2009)
35. Raptis, M., Kirovski, D., Hoppe, H.: Real-time classification of dance gestures from skeleton animation. In: Proceedings of the 2011 ACM SIGGRAPH/Eurographics symposium on computer animation, pp. 147–156 (2011)
36. Cimen, G., Ilhan, H., Capin, T., Gurcay, H.: Classification of human motion based on affective state descriptors. Comput. Anim. Virtual Worlds **24**(3–4), 355–363 (2013)
37. Hsu, E., da Silva, M., Popović, J.: Guided time warping for motion editing. In: Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on computer animation, pp. 45–52 (2007)
38. Caspi, Y., Irani, M.: Aligning non-overlapping sequences. Int. J. Comput. Vis. **48**(1), 39–51 (2002)
39. Caspi, Y., Irani, M.: Spatio-temporal alignment of sequences. IEEE Trans. Pattern Anal. Mach. Intell. **24**(11), 1409–1424 (2002)
40. Padua, F.L.C., Carceroni, R.L., Santos, G.A.M.R., Kutulakos, K.N.: Linear sequence-to-sequence alignment. IEEE Trans. Pattern Anal. Mach. Intell. **32**(2), 304–320 (2010)
41. Junejo, I.N., Dexter, E., Laptev, I., Perez, P.: View-independent action recognition from temporal self-similarities. IEEE Trans. Pattern Anal. Mach. Intell. **33**(1), 172–185 (2011)
42. Lu, C., Mandal, M.: A robust technique for motion-based video sequences temporal alignment. IEEE Trans. Multimed. **15**(1), 70–82 (2013)
43. Pravdova, V., Walczak, B., Massart, D.L.: A comparison of two algorithms for warping of analytical signals. Anal. Chim. Acta **456**, 77–92 (2002)
44. Tomasi, G., van den Berg, F., Andersson, C.: Correlation optimized warping and dynamic time warping as preprocessing methods for chromatographic data. J. Chem. **18**(5), 231–241 (2004)
45. Tomasi, G.; Practical and computational aspects in chemometric data analysis. Ph.D. Thesis, The Royal Veterinary and Agricultural University (2006)
46. Skov, T., van den Berg, F., Tomasi, G., Bro, R.: Automated alignment of chromatographic data. J. Chem. **20**, 484–497 (2006)
47. Wang, S., Yao, J., Liu, J., Petrick, N., Van Uitert, R.L., Periaswamy, S., Summers, R.M.: Registration of prone and supine CT colonography scans using correlation optimized warping and canonical correlation analysis. Med. Phys. **36**(12), 5595–5603 (2009)
48. Yu, T., Shen, X., Li, Q., Geng, W.: Motion retrieval based on movement notation language. Comp. Anim. Virtual Worlds **16**(3–4), 273–282 (2005)
49. Tang, J.K., Leung, H., Komura, T., Shum, H.P.: Emulating human perception of motion similarity. Comput. Anim. Virtual Worlds **19**(3–4), 211–221 (2008)
50. Gray, P.O.: Psychology, 5th edn. Worth, New York (2006)
51. Wolfe, J.M., Kluender, K.R., Levi, D.M., Bartoshuk, L.M., Herz, R.S., Klatzky, R.L., Lederman, S.J.: Gestalt Grouping Principles. Sensation and perception. 2nd ed., Sinauer Associates. pp. 78–80 (2008)

52. Wang, J., Yagi, Y., Makihara, Y.: People tracking and segmentation using efficient shape sequences matching. In: Zha, H., Taniguchi, R., Maybank, S. (eds) 9th Asian Conferenceon Computer Vision, vol. 5995, pp. 204–213. Springer, Berlin (2009)
53. Wang, J., Bodenheimer, B.: Synthesis and evaluation of linear motion transitions. ACM Trans. Graph. **27**(1), 1 (2008)
54. Wang, Y.-S., Lin, H.-C., Sorkine, O., Lee, T.-Y.: Motion-based video retargeting with optimized crop-and-warp. ACM Trans. Graph. **29**(4), 90 (2010)
55. Prazák, M., Hoyet, L., O'Sullivan, C., Perceptual evaluation of footskate cleanup. In symposium on computer animation, pp. 287–294 (2011)
56. Bouillaguet, C., Chen, H.C., Cheng, C.M., Chou, T., Niederhagen, R., Shamir, A., Yang, B.Y.: Fast exhaustive search for polynomial systems in F2. Cryptographic hardware and embedded systems, pp. 203–218. Springer, Berlin (2010)

S. Ali Etemad received his B.Sc. from the Isfahan University of Technology, Iran, in 2007, and his M.A.Sc. and Ph.D. in electrical and computer engineering from Carleton University, Ottawa, Canada, in 2009 and 2014, respectively. He is currently an adjunct professor at the School of Information Technology, Carleton University. He is also the Head of Machine Intelligence and HCI at GestureLogic Inc. His areas of research are machine learning, pattern recognition, artificial intelligence, and image and video processing with a focus on human motion analysis for interactive multimedia, HCI applications, and wearable technologies. He is a recipient of the Ontario Graduate Scholarship among several other scholarships, awards, and grants. He is an associate editor for the Journal of Robotics and Artificial Intelligence. He is also a reviewer for several journals and has been a technical committee member for various conferences.

Ali Arya received his bachelor's degree in electrical engineering from Tehran Poly-technique, Iran, in 1989 and his Ph.D. in computer engineering from the University of British Columbia, Canada, in 2003. He has more than 10 years of industry experience as project manager and software engineer, has worked as instructor and researcher in the University of British Columbia and Simon Fraser University in Vancouver, Canada, and joined the School of Information Technology, Carleton University, Ottawa, Canada, in 2006 where he is currently an associate professor of interactive multimedia and design. Ali's major research areas are Computer graphics and animation, multimedia systems, human–computer interaction, virtual worlds and collaborative environments, artificial intelligence, and digital art. He is a senior member of IEEE, on the editorial board of International Journal of Computer Games Technology and Journal of Systemics and Informatics, and member of technical and program committees of many conferences in the area of multimedia systems. His research has been funded by NSERC, SSHRC, OCE, and industry partners.