Wavelet Based Keyframe Extraction Method for Motion Capture Data

Wei XIN 辛 慎
Kunio KONDO 近藤 邦雄
Kei TATENO 蝴野 圭
Toshihiro KONMA 今間 俊博
Tetsuya SHIMAMURA 島村 健也

Abstract: This research extracts keyframes from Motion Capture Data. Accurate selection of keyframes is a key point in motion synthesizing, editing, retargeting and motion data compression. Our method consists of two steps, curve selection and curve smoothing. In curve selection, every joint curve is re-represented with a simple numeric sequence based on wavelet transformation. Then the correlation between curves is calculated based on these numeric sequences. Several curves which are least correlated are chosen for keyframe extraction. In the next step, a noise filter is applied to selected curves. Finally the keyframes are selected from the de-noised curves.

Keyword: CG, Animation, Motion capture data, Keyframe extraction

1. Background
In recently years, motion capture has become one of the key techniques for generating character animation. It can record a complete movement with all the detail of live motion, so it saves time and creates more natural movement than manual animation. Motion capture, however, also has its limitation. Motion is usually captured for a given purpose and character. When a set of motion data is recorded, it is difficult to edit the data or apply the data to a character with different skeleton size. To overcome this limitation, many techniques are developed to reuse motion data.

When working with motion capture data, an important part is to decide keyframes in a set of motion data. Many operations such as motion editing, motion synthesizing, motion retargeting and motion data compression are based on selected keyframes.

In order to generate new motion from existing motion data, a common method is to select keyframes from several motions, then compare each pair of keyframes to find transition frames, and finally blend motion segments together using transition frames. Lucas Kovar, Michael Gleicher, and Frederic Pighin [1] automatically construct a motion graph that encapsulates connections among different pieces of motion in the database and then searching this graph for motions that satisfy user constraints.

Keyframe extraction can also be used in retargeting motion of a source character to a target character with a different structure. Park and Shin [2] automatically extract a small number of representative postures from a set of example motions. Then each posture is cloned frame by frame to the target character.

In the field of motion capture data compression, Vadym Voznyuk [3] applied principal component analysis (PCA) to reduce the dimension of motion capture data.


"Realistic" is one big advantage for motion capture. But when motion capture data is applied to a very "cartoon-like" model, this advantage turns to be a disadvantage. Different from manual animation, motion capture lacks for some important elements of traditional animation like squash, stretch and anticipation, which endows characters with greater impact and personality. Kei Tateno [5] proposed method of motion stylization using a timing control method. This method generates exaggerated motion by controlling interpolation between keyframes selected from original motion capture data.

In this paper, we present a novel method for extracting keyframes from motion capture data. Our method firstly selects several least correlated joints curves from original motion capture data. Then a wavelet noise filter is applied to selected curves for clean and smooth curves from which keyframes are extracted.

2. Overview
Motion capture data contains around 50 DOFs (degrees of freedom), which brings great difficulties for keyframe selection. At the same time, some DOFs are closely correlated with others. For example, in walking motion, the right upper arm and right lower arm tend to move with the same rhythm. Thus their joint curves may also show similar shapes. It means that one joint curve from right arm is enough to represent all the important movement of right arm. In order to extract keyframes accurately, we should find several representative joint curves which are not similar to each other. In other words, the high dimensional motion data should be reduced into a few DOFs. The general way is to apply the PCA (Principal Component Analysis) technique on motion data to yield a reduced dimension space. In this research, a novel algorithm is proposed in calculating the correlation of motion data. Figure 1 is the flow chart of the method proposed in this paper. The core of this algorithm is wavelet transformation, which has a lower computation complexity than PCA.

Firstly, using Haar wavelet transformation, original joint curves are transformed into jagged shapes, which can keep the general features of original data. To simplify the calculation of curve comparison, we further omit the amplitude of the jagged approximation by representing these curves with a simple numeric sequence, which can be regarded as index of original curve. Next, the correlation coefficients of curves are calculated out of their indexes. Based on these correlation coefficients, a few least correlated curves are chosen for keyframe extraction.
In the next step, a wavelet noise filter is applied to smooth the selected curves. Many raw motion capture data contain considerable amount of noises. These noises bring difficulties in the process of extraction. So we need to clean the noise before working on the data.

Figure 1. Overview of the algorithm

In the field of signal processing, Fourier transform is often used to de-noise one dimension signals. In our algorithm, we use wavelet transform instead of Fourier transform for this purpose. Practical experience has shown that for many applications, wavelet transforms are as powerful as Fourier transform, yet without some of the limitations of the latter.

After de-noising, curves became smooth and clean. Then the local maximum and minimum value of these curves are selected as keyframes.

3. Implementation

The core of this keyframe extraction method is wavelet transformation. In motion data dimension reduction, wavelet transformation is faster than PCA; in curves de-noising, wavelet outperforms Fourier transform in almost every aspect. The following part is a brief introduction of wavelet transform used in this research.

3.1 Wavelet transform

The traditional method used in analyzing time series is Fourier transform. It is based on the simple observation that every signal can be represented by a superposition of sine and cosine waves. Wavelets can be thought of as a generalization of this idea to a larger family of functions than sine and cosine.

Fourier transformation includes Fourier series expansion, the discrete Fourier transform, and the integral Fourier. The counterparts in wavelet domain are the generalized wavelet series expansion, the discrete wavelet transform, and the continuous wavelet transform. The wavelet series expansion maps a function of a continuous variable into a sequence of coefficients. The wavelet series expansion function \( f(x) \in L^2(\mathbb{R}) \) relative to wavelet \( \psi(x) \) and scaling function \( \phi(x) \) is:

\[
f(x) = \sum_{k} c_{j_0}(k)\phi_{j_0,k}(x) + \sum_{j=j_0}^{\infty} \sum_{k} d_{j}(k)\psi_{j,k}(x)
\]

where \( j_0 \) is an arbitrary starting scale. The \( c_{j,j}(k) \)'s are normally called the approximation or scaling coefficients; the \( d_{j}(k) \)'s are referred to as the detail or wavelet coefficients. This is because the first sum in above equation uses scaling functions to provide an approximation of \( f(x) \) at scale \( j_0 \). For each higher scale \( j \geq j_0 \) in the second sum, a finer resolution function -a sum of wavelets-is added to the approximation to provide increasing detail. If the expansion functions form an orthonormal basis of tight frame, which is often the case, the expansion coefficients are calculated as:

\[
c_{j,k}(k) = \langle f(x), \phi_{j,k}(x) \rangle = \int f(x)\phi_{j,k}(x)dx
\]

And

\[
d_{j,k}(k) = \langle f(x), \psi_{j,k}(x) \rangle = \int f(x)\psi_{j,k}(x)dx
\]

Scaling function

The set of expansion functions is composed of integer translations and binary scaling of the real, square-integrable function \( \phi(x) \); that is, the set \( \{\phi_{j,k}(x)\} \) for all \( j, k \in \mathbb{Z} \) and \( \phi(x) \in L^2(\mathbb{R}) \). Here, \( k \) determines the position of \( \phi_{j,k}(x) \) along the x-axis, and \( 2^j \) controls its height or amplitude. The shape of \( \phi_{j,k}(x) \) changes with \( j \), so \( \phi(x) \) is called a scaling function. By choosing \( \phi(x) \), \( \{\phi_{j,k}(x)\} \) can be made to span \( L^2(\mathbb{R}) \), the set of all measurable, square-integrable functions.

Wavelet functions

The subspace spanned over \( k \) for any \( j \) is \( V_j \), which can be called scaling subspace. The scaling function should obey the four fundamental requirements of multiresolution analysis (MRA)[12].

Given a scaling function that meets the MRA requirements, we can define a wavelet function \( \psi(x) \) that, together with its integer translates and binary scaling, spans the difference between any two adjacent scaling subspaces \( V_j \) and \( V_{j+1} \). \( W_j \) is called wavelet function subspace. We define the set \( \{\psi_{j,k}(x)\} \) of wavelets for all \( k \in \mathbb{Z} \) that spans the \( W_j \) spaces

\[
\psi_{j,k}(x) = 2^{j/2} \psi(2^j x - k)
\]

The scaling and wavelet function subspaces are related by

\[
V_{j+1} = V_j \oplus W_j
\]
where $\oplus$ denotes the union of spaces. The orthogonal complement of $V_j$ in $V_{j+1}$ is $W_j$, and all members of $V_j$ are orthogonal to the members of $W_j$. Therefore

$$\langle \phi_{j,k}(x), \psi_{j,l}(x) \rangle = 0$$

for all appropriate $j, k, l \in \mathbb{Z}$.

Unlike the DFT that takes the original signals in time/space domain and transforms them into frequency domain, the Wavelet transform takes the original signals in time/space domain and transforms them into time/frequency or space/frequency domain. Since the wavelet transform gives a time-frequency localization of the signal, it means that most of the energy of the signal can be represented by only a few DWT coefficients. Therefore, we can transform the motion capture data into wavelet representation without losing of general information.

### 3.2 Curve Selection

#### 3.2.1 Haar wavelet transformation

In order to re-represent joint curves with an easy-to-handle form for correlation calculation, we firstly transform them into jagged shape with the Haar wavelet. The Haar wavelet is the most elementary wavelet. It is defined as wavelet function

$$\psi(x) = \begin{cases} 1 & 0 < x < 0.5 \\ -1 & 0.5 < x < 1 \\ 0 & \text{otherwise} \end{cases}$$

together with a scaling function

$$\phi(x) = \begin{cases} 1 & 0 < x < 1 \\ 0 & \text{otherwise} \end{cases}$$

By setting different decomposition levels, we can decide how precise the simulation is. Figure 2 is the simulation of joint curves at different decomposition levels. When the curve is decomposed at a low level, the reconstructed curve can simulate well even the details of original curve. When decomposed at a high level, the simulation keeps only the general pattern of original curve. In this research, we set the decomposition level to 2 or 3. By doing so, the important features of the motion can be kept while some subtle movements are ignored.

#### 3.2.2 Curve re-representation and correlation calculation

The detailed curve re-representation process is as follows:

1. Transform original curves (Figure 3(a) into jagged shape using Haar wavelet (Figure 3(b)). Decomposition level could be set to adjust the precision of simulation.
2. Represent curves with sequence of $+1$ and $-1$. When the curve tends to increase, set the $Y$ value as $+1$; set the value as $-1$ when decrease (Figure 3(c)). Since we care only the frames where curve changes its direction, the original curve is re-represented in a new form which doesn’t contain $Y$ axis information (Figure 3(d)). This numeric sequence consisting of $+1$ and $-1$ could be regarded as coefficient of original curve.

By multiplying their corresponding coefficient and summing the value up, we get the result of $-6$. Because the possible minimal value is $-8$, we can conclude that these two curves are negative correlated. When the similarity coefficients of joint curves are calculated, we can choose a few curves which are least correlated for keyframes extraction. Local maximum and minimum value of selected curves are keyframes needed. In order to select them correctly, we smooth the curves by applying a noise filter, which is introduced in the next section.
3.3 Motion curve de-noising

The model of noisy signal is:

\[ s(n) = f(n) + \sigma e(n) \]

where time \( n \) is equally spaced. \( e(n) \) is a Gaussian white noise \( N(0,1) \) and the noise level \( \sigma \) is supposed to be equal to 1. The de-noising objective is to suppress the noise part of the signal \( s \) and recover \( f \). If only a limited number of highest coefficients of the DWT spectrum are taken, and then an inverse transform is performed, we can obtain more or less de-noised signal. There are several ways to choose the coefficients that will be kept. The common methods are Hard thresholding, Soft thresholding and Shrinkage.

- **Hard thresholding.** As shown in Figure 4(a), when the coefficients on some or all scales that are below a certain threshold are set to zero while the coefficients is superior to the selected threshold remain unaltered. This is one ‘keep-or-kill’ procedure. Given an increasing threshold, this method brings subjectively disturbing artefacts.

- **Soft thresholding.** When the coefficients on some or all scales below a threshold are set to zero, and additionally, all coefficients above the selected threshold are shrunk by the value of the threshold (Figure 4(b)). This method attenuates the range of the wavelet coefficients and smooths the signal, thus modifying its energy.

- **Shrinkage.** This is a compromise between the above two thresholding methods (Figure 4(c)). It preserves the highest coefficients but has a smooth transition between the cut-off and the maintained coefficients.

In general, the sophisticated shrinkage schemes are computationally very expensive, while the hard thresholding exhibits artefacts in some conditions. Soft thresholding is a good compromise between computational complexity and performance; we choose soft thresholding in curve de-noising.

The de-noising procedure involves three steps:

- **Decompose**

  Choose a wavelet; choose a level \( N \). Compute the wavelet decomposition of the signal \( s \) at level \( N \).

- **Threshold detail coefficients**

  For each level from 1 to \( N \), select a threshold and apply soft thresholding to the detail coefficients.

- **Reconstruct**
Compute wavelet reconstruction using the original approximation coefficients of level N and the modified detail coefficients of levels from 1 to N.

3.4 Keyframe selection
When a few least correlated joint curves are selected and smoothed, their local maximum and minimum points are extracted as keyframes. As shown in Figure 5, the blue and green curves are selected joint curves. Red dots are local maximum and minimum values. And the poses to which arrows pointed are keyframes.

Figure 5. Local maximum and minimum points from selected motion curves are extracted as keyframes.

4. Gait motion keyframe extraction
When the character walks or runs, the body rises and dips at certain points throughout the action. As the leg swings forward and the character is supported on one leg (the passing position), the body rises; as the leading leg makes contact with the ground (the stride), the body is lower. Therefore, to extract keyframe for gait motion, we can work on only the Y translation of Hip, which contains the movement of the body's up and down.

Figure 6 illustrates the keyframes extracted from Hip Y Translation for a walking motion and a running motion respectively. Each of the gait cycle consists of five keyframes, which are the full set of movements needed to make up a smooth 2D gait animation.

This method also can be applied to gait motion displaying different moods, such as tired walk, sneak, cripple and so on. We can find the trace of different mood running or walking motion in the Y translation data of character’s hip bone.

Figure 6. Keyframes extracted from Hip Y translation in motion data. The dashed blue curve is one section of Hip Y translation data; red dots are keyframes selected.

Figure 7 demonstrates the result of our keyframe extraction method. Figure 7(a) is one segment of a cripple walking motion. In this motion, the keyframes are extracted from only the Y translation curve of hip. Totally 54 keyframes are extracted out of 546-frame original motion. We can see in this illustration that there is no redundant frame generated. And the motion clip made with extracted keyframes can resemble the original motion well. In figure 7(b), 16 keyframes are selected from a 111-frame combat motion. Two joint curves are used in extracting keyframes for this motion. Among these 16 keyframes, frame 77 is dispensable, but all of other keyframes extracted are necessary for keeping good timing and consistency for original motion.

5. Experiment result
We used our algorithm in extracting keyframes from a wide range of motion capture data, such as walk, run, jump, dance, sports, combat motion and so on. Besides, this algorithm is tested on data with different DOFs.

In our experiment, selecting two joint curves from motion data for keyframe extraction gives the best result. In general, for motions other than gait motion, one curve is not enough to cover all the characteristics of original movement. While extracting keyframes from more than two curves generates a lot of redundant frames. In this research, the algorithm is performed with Matlab, and the animation illustrations are generated and rendered with Poser.

6. Conclusion
We proposed the method of keyframe extraction from motion capture data using a joint curve re-representation algorithm and wavelet noise filter. We have demonstrated using this method we can make a good motion approximation of original motion capture data. The keyframes extracted can cover almost all of the important features of original motion, and at the same time introduce only the minimum redundant keyframes.
Moreover, we extract the keyframe for gait motion from only one joint curve. By doing so, almost no redundant keyframes are generated.

References
[7] Uros Lotric, Andrej Dohnikan, Neural Networks with Wavelet Based Denoising Layer for Time Series Prediction, Neural Computing and Applications, 14, (2005),pp.11-17,
[8] www.mathworks.com

About the Authors
Wei XIN, Kei TATENO, Kunio KONDO, Tetsuya SHIMAMURA: Saitama University
Toshihiro KONMA: Shobi University