-Adaptive Compression of Human Animation Data

Amr Ahmed, Adrian Hilton, and Farzin Mokhtarian

{*A.Ahmed* | *A.Hilton* | *F.Mokhtarian*}*@eim.surrey.ac.uk*

Centre for Computer Vision, Speech, and Signal Processing,

Department of Electronic Engineering,

University of Surrey, Guildford, Surrey GU2 7XH, UK.

Fax: +44 (0) 1483 686031 Tel.: +44 (0) 1483 686030

Abstract

In this paper we introduce an adaptive motion compression technique using the discrete wavelet transform. Based on the analysis of human animation data, and its frequency contents, the wavelet analysis is utilised to achieve motion compression. It has been found that even with high compression ratios (up to 86%), the compressed animation is visually very close to the original animation. This property is potentially useful for many areas of animation such as motion interpolation/blending and networked virtual environments.

One of our motivations is that motion blending/interpolation between motion capture samples is one of the successful techniques for synthesis of novel realistic human animation. One of the major concerns of that method is the increase of the database size with the large number of samples required to synthesise a wide range of movements. Previous research has addressed this issue by trying to reduce the number of samples required for interpolation. The approach that we introduce in this paper is to reduce the individual sample's size using compression. Integration of both these approaches promises to allow a realistic animation with reduced database size.

In networked virtual environments and on-line games, compressing the animation data can reduce the transmission load and help in achieving real-time performance and realism with reduced cost.

Keywords: ***Human Animation – Data Compression – Virtual Human – Virtual Environment***

1. Introduction

There is an increasing demand for realistic human motion animation in a wide variety of fields and applications; from games and entertainment to simulation and scientific visualisation. Motion capture is one of the most popular methods to produce a realistic and more natural-looking human animation. Editing this animation data has been the focus of many researches [1, 2, 3, 4, 5, 6, 7, 8, 9] to allow the reuse of this data and facilitate the generation of variations in the original motion. Motion blending and interpolation is one of the common and successful techniques for synthesis of realistic human animation from pre-stored samples. It produces reasonable results –especially when using captured data- with relatively low computation cost. However, one of the problems that arise within this technique is the increase of the size of the animation database required to characterise variations in movement. As the capabilities of this method are based on the available motion samples, so, the more variations in the resulting motion, the more motion samples will be required. This issue becomes important with the parametric synthesis of human animation. The more parameters we introduce, the more motion samples will be required.

In this paper, we present an adaptive approach to compress the size of the animation database. The previous approaches to address this problem tried to reduce the number of motion samples [8, 9]. Our approach is to reduce the animation database size by compressing the data of each sample itself. We have utilised the wavelet analysis to achieve the adaptive motion compression with quite small errors.

The compression of animation data is useful in many animation areas not only in building a large database of animation clips for motion blending or interpolation. It is also potentially useful in on-line gaming and networked virtual environments where the volume of data is critical due to the available transmission rates and bandwidths. So, performing such compression can help to reduce the data volume, speed up the transmission and most importantly maintain the realistic appearance of the motion with reduced cost.

In the next section, we give a quick overview of some related work. A brief introduction to the wavelet transform and its properties is given in section 3. In section 4, the proposed adaptive compression technique for animation data is presented. Results and conclusions are presented in sections 5 and 6 respectively.

2. Related Work

Wavelet analysis has been used in computer graphics applications such as multiresolution curve editing [10], image processing and compression [11], and many other applications. However, there is little previous research in applying wavelet analysis to the human motion animation.

In Human Animation, the motion data represents the trajectory of the human skeleton joints. Each joint trajectory is a time varying signal that can be processed by a signal processing tool like wavelets.

Liu et. al. [12] employed wavelets to accelerate the space-time interpolation of their physically-based keyframe animation. On the other hand, Sun [13] used the wavelets to describe/model bipedal locomotion. Wavelets have been used also for smoothing the kinematics' motion data in the biomechanics studies [24]. In [15], we introduced a parametric motion blending approach using wavelet interpolation and blending. Analysis for the motion curves and its frequency contents has shown that we can utilise the wavelet transform to achieve a reasonable level of data compression without disturbing the realism of the generated animation (as shown in sections 4 and 5).

Multiresolution analysis using different techniques has been explored in [7, 16, 17]. However, to the best of our knowledge, the compression issue has not been investigated.

Wavelet compression techniques are already used in image and video compression. We address this issue for human animation because it can be useful in many areas such as motion editing and blending, online games and networked virtual environments. In such situations, compressing the animation data –with minimum visual effect- can reduce the transmission load and help in maintaining the real-time performance with reduced cost.

3. Wavelet Transform

Wavelet transform (like many other transforms) provides a representation of the signal/function with the following main properties:

1. It is localized in time (which preserve the temporal aspects of the signal) and frequency. This provides better local description and separation of the signal features.

2. It can handle non-periodic functions and discontinuities without additional complexity. It is also suitable for analysing transient and non-stationary signals [18].

4. It can be extended to an arbitrary number of dimensions in a straightforward fashion.

6. Adjustable and adaptable. Since there is no unique wavelet, wavelets could be designed to match each application (or group of applications).

7. The number of wavelet coefficients decreases rapidly with the level of resolution, which is a useful property in compression applications.

Numerical calculation of the wavelet transform coefficients is performed efficiently using the filter bank algorithm (discrete wavelet transform) using multiplication and addition operations only (no derivatives or integrals). In most cases, the calculation time is linearly related to the length of the signal [18]. The computation complexity of the fast wavelet transform is O(n) (where 'n' is the signal length) compared to $O(n\log_2(n))$ for the Fast Fourier Transform [19].

One of the important features of wavelet transform is the "Multiresolution Analysis" (MRA) property. MRA means that the signal is analysed at different frequencies with different resolution. For more details we suggest the reader to refer to [20, 21, 22, 18, 23]

In wavelet transform, the signal ***S*** at resolution level ***j*** can be written as:

$$S_j(t) = \sum_k C_{j-1,k}\varphi_{j-1,k}(t) + \sum_k D_{j-1,k}\psi_{j-1,k}(t)$$

Where $\varphi_{j-1,k}(t)$ are called the scaling functions, $\psi_{j-1,k}(t)$ are called the wavelet functions, C_{j-1} are the scaling functions coefficients at resolution level *j-1*, and *k* is the index of the scaling function within that level. Similarly, D_{j-1} are the wavelet functions coefficients at resolution level

j-1.

Using the nested space concept, we find that:

$$S_j(t) = \sum_k C_{0,k}\varphi_{0,k}(t) + \sum_{m=0}^{j-1}\sum_k D_{m,k}\psi_{m,k}(t)$$

The sequence $\{C_0, D_0, D_1, \ldots, D_{j-1}\}$ is called the wavelet representation of the signal *S*. $C_0, D_0, D_1, \ldots, D_{j-1}$ are called the wavelet transform coefficients.

In the matrix form:

$$S_j = C_j = \left[P_j \,|\, Q_j\right] \begin{bmatrix} C_{j-1} \\ \hline D_{j-1} \end{bmatrix}$$

This expression is called the synthesis expression, which is used to reconstruct or reproduce (or synthesise) the higher resolution level from its lower resolution level *j-1* as shown in figure 2. *P* and *Q* are called the synthesis matrices. The expressions that can be used for analysis (or decomposition) of the signal (as shown in figure 1) can be written as follows:

$$C_{j-1} = A_j C_j \quad \text{and} \quad D_{j-1} = B_j C_j$$

Where

$$\begin{bmatrix} A_j \\ \hline B_j \end{bmatrix} = \left[P_j \,|\, Q_j\right]^{-1}$$

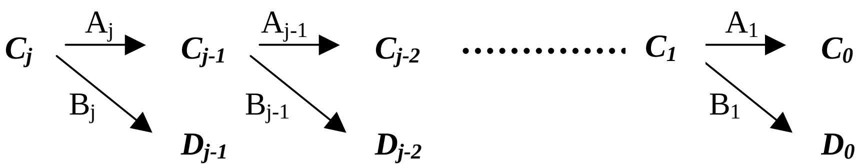


Figure 1: Wavelet decomposition/analysis

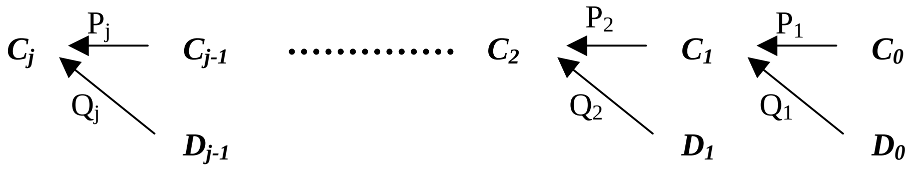


Figure 2: Wavelet reconstruction/synthesis

4. Adaptive Wavelet Motion Compression

The wavelet compression technique is based on using the minimum number of wavelet transform coefficients to produce/synthesise a signal that is as close as possible to the original signal. In the simplest form, this can be done simply by discarding the detail coefficients D_{j-1} then D_{j-2} and so on according to the required compression ratio or acceptable error tolerance. This assumes that the signal content is more concentrated in the coarse level (low-frequency contents). We have found that this assumption is valid in many basic human motions (as shown in section 5). Bruderlin et. al. [24] reported that the waving and knocking motions contain mainly low-frequency contents. We have extended the analysis to some other human motions such as walking, running, dancing, and fighting as well. It has been found that discarding some of the small details coefficients will not introduce a large effect on the motion (as described in the results section). Figures 3 shows the frequency contents for some samples of different human motions.

Our proposed technique is an adaptive compression for human animation data. Human animation data is usually represented by a 3D translation of the root of the skeleton hierarchy plus 3D rotation of each DOF of the hierarchy's joints (including the root as well). So, each motion signal we are dealing with is a curve representing the rotation about one of the 3D axis. The behaviour of these curves/signals is not necessary to be similar. It varies from one DOF to the other and also differs from one motion to the other. According to that, using a constant compression ratio for all DOF's signals (i.e. discarding the same number of coefficients) may introduce a large amount of errors and the resulting motion may appear unrealistic (or even appear incorrect in higher compression ratios).

Our approach is to use an adaptive compression, which means that each signal can be compressed with different compression ratio that maintains a limited error. So, according to the behaviour of each signal, we find some signals have very high compression ratios, and others have small compression rations. However, the overall compression ratio is based on the wavelet coefficients that are used to reconstruct the motion again. The advantage of using this adaptive compression is that it limits the individual signal's error, which results in better results and less visual effects with reasonable overall compression ratios.

The wavelet compression technique can be summarised in the following steps:

1. ***Analysis or decomposition***: In this step, the motion signal will be decomposed to its wavelet representation $\{C_0, D_0, D_1, \ldots, D_{j-1}\}$.

   In our implementation, we have used cubic B-Spline wavelets as recommended in the literature for computer graphics and curve editing [10].

2. ***Compression and synthesis/ reconstruction***: For perfect reconstruction of the original signal, we should use the full wavelet transform coefficients. However, for compression, if we have *m* coefficients, we need to reproduce the signal using *n* coefficients where $n<m$. n can be determined based on one of the following:

   a) Based on the frequency-contents analysis, the values of the details levels coefficients are smaller in the higher details levels (*j-1, j-2,* ,etc.). So, we can simply discard some of the details levels coefficients starting from D_{j-1} then D_{j-2} and so on.

   b) Alternatively, (as in the proposed work) we can use some criteria to select the coefficients that will be discarded. For example, we can discard the coefficients so that the total error is within some specified value. So, given a pre-defined error value *E*, find the minimum number of wavelet coefficients *n* that represents the same signal with error less than (or equal to) *E*. This can be implemented in many ways. For example, we can sort the coefficients in a descending order (based on the magnitude) and then discard from the end the coefficients such that their sum is less than or equal to the specified error. However, for large motion clips, sorting the coefficients will slow down the process. Another more efficient way is to use a binary search algorithm [11] to find the coefficients to be discarded without need to sort all the coefficients beforehand.

Our adaptive technique uses the second option in the compression with the binary search algorithm. In fact we have implemented both options and found that the adaptive one is better for the reasons discussed above.

Regarding the evaluation methods, firstly, we would like to mention that the evaluation of how realistic the animation is, is not an easy process. It is qualitative not quantitative. So, the visual evaluation is a very important issue as it is what the user will actually see. However, we introduce quantitative measures to determine the differences between the original and the compressed motions. Some of these quantitative measures are general and independent of the motion type and some other measures vary according to the type of motion.

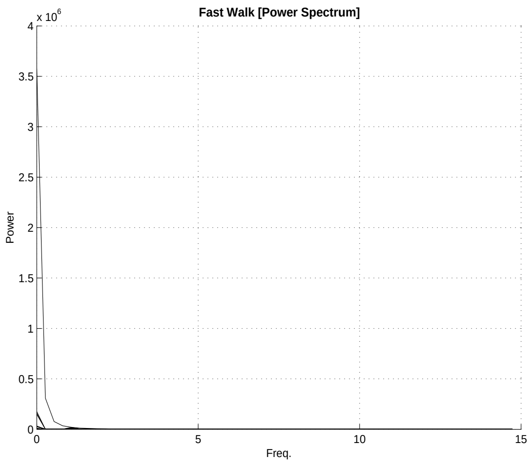
As an example of measures that are independent of the type of motion, we use the difference in 3D positions of joints between the original and compressed motions. This measure indicates how the joints of the compressed motion are shifted from the original motion. It can be formulated as:

$$3D\_Pos\_Error = \sum_{j=1}^{N} \left| OP_j - CP_j \right|$$

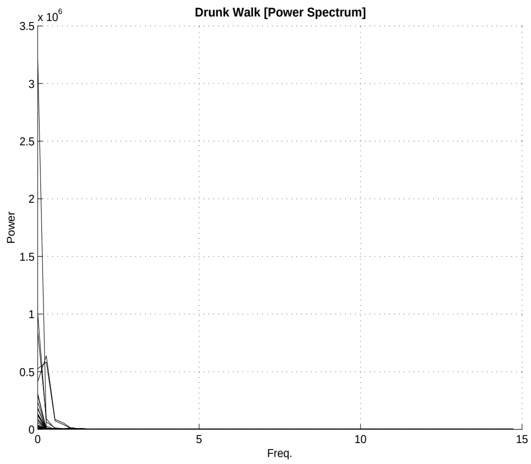
Where OP_j and CP_j are the 3D positions of joint 'j' in original and compressed motions respectively. 'N' is the number of joints in the skeleton.

For some types of motion such as walking and running motions, we can use additional measures such as the vertical feet position as a measure of error. This is an example of the measures that is specific to some motion types. Vertical feet position was used in [15] as a measure of satisfying or violating one of the motion constraints for walking and running which is touching the floor at certain key-times (heel-strike). It is used here to show how the compression will affect motion constraints.

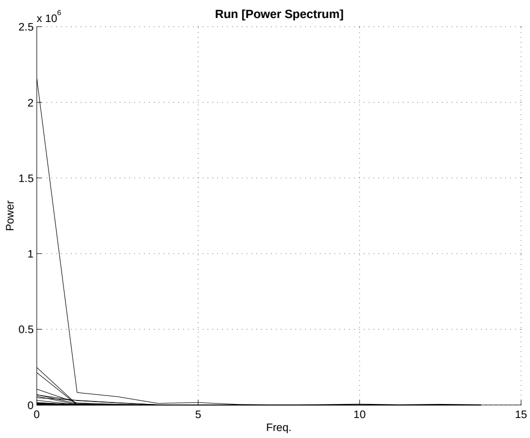
The results of our experiments are shown in the next section, which shows the compression results with the corresponding errors and visual effects.



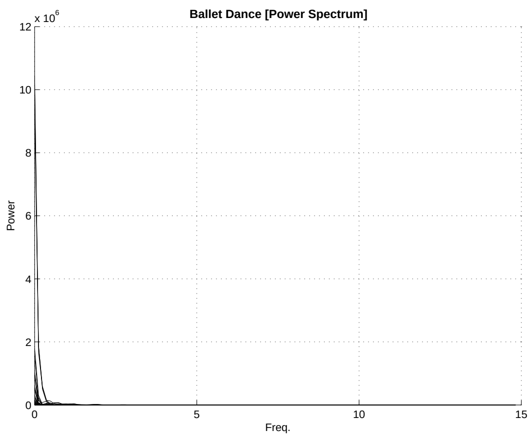
Walk



Drunk Walk



Run



Ballet Dance

Figure 3: Frequency contents of different human motions

5. Results

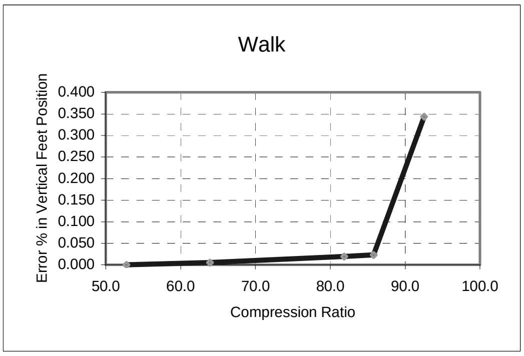
In this section, we show the compression results for some human motion data. As we mentioned before, evaluating the animation is not an easy process because it is naturally qualitative not quantitative. So, the visual evaluation is the key in the evaluation of how realistic the animation is. However, we have introduced some quantitative measures -such as the difference in the 3D positions of joints and vertical feet position- to show the effect of compression on the resulting animation as discussed in the previous section.

The figures below show some compression results and its effect for the human walking and running motions. Figures 4 and 5 show the effect of the compression ratio on the calculated errors in 3D positions and feet vertical position for ‘walking’ and ‘running’ motions respectively. It can be noticed that even with compression ratios up to 75% (more than 86% for walking), the calculated errors are still less than 0.8%. More importantly, the compressed motion is visually very close to the original motion as can be noticed from the snapshots in figure 8. Some samples of these results can be found in the following URL: *http://www.ee.surrey.ac.uk/Personal/ A.Ahmed/EG02/Demos/*

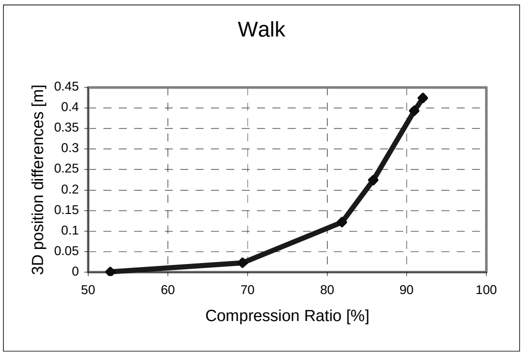
The clips used in our experiments are in BioVision format (BVH), which uses Euler Angles for rotation parameterisation. Table (1) gives an indication of the compression times for the range of 70%-87% compression on a Linux-PC. As our implementation is done using Matlab, we believe that an efficient and optimised 'C' code will improve the timing.

6. Conclusions

It has been shown that wavelet analysis can be useful for compression of human animation as it is for image and video compression. We introduced an adaptive compression technique that utilise the discrete wavelet transform to achieve a reasonable compression ratios with quite small visual effects. Based on studying the frequency contents of the available animation samples (Figure 3 shows some of them), some of the small details coefficients can be discarded from the wavelet representation.

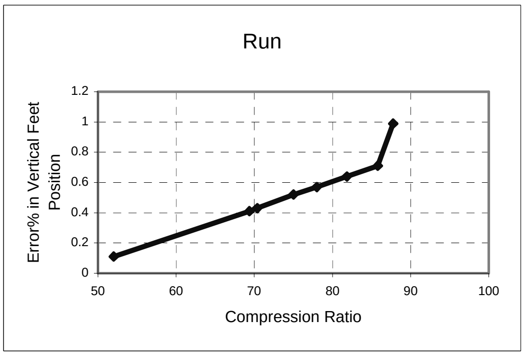


A. Effect of compression on the feet position

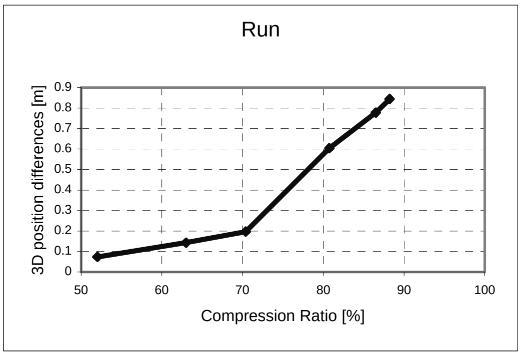


B. Effect of compression on the 3D positions

Figure 4. Effect of compression on ‘Walking’ motion



A. Effect of compression on the feet position



B. Effect of compression on the 3D positions

Figure 5: Effect of compression on ‘Running’ motion

Frames #	Joints #	Total DOFs	Clip Time (sec.)	Compression. Time(sec.)	Compression. Time %
35	57	171	11.66	1.73	14.8%
259	57	171	86.3	8.3	9.6%

Table 1: Indication for compression times [70-87% compression]

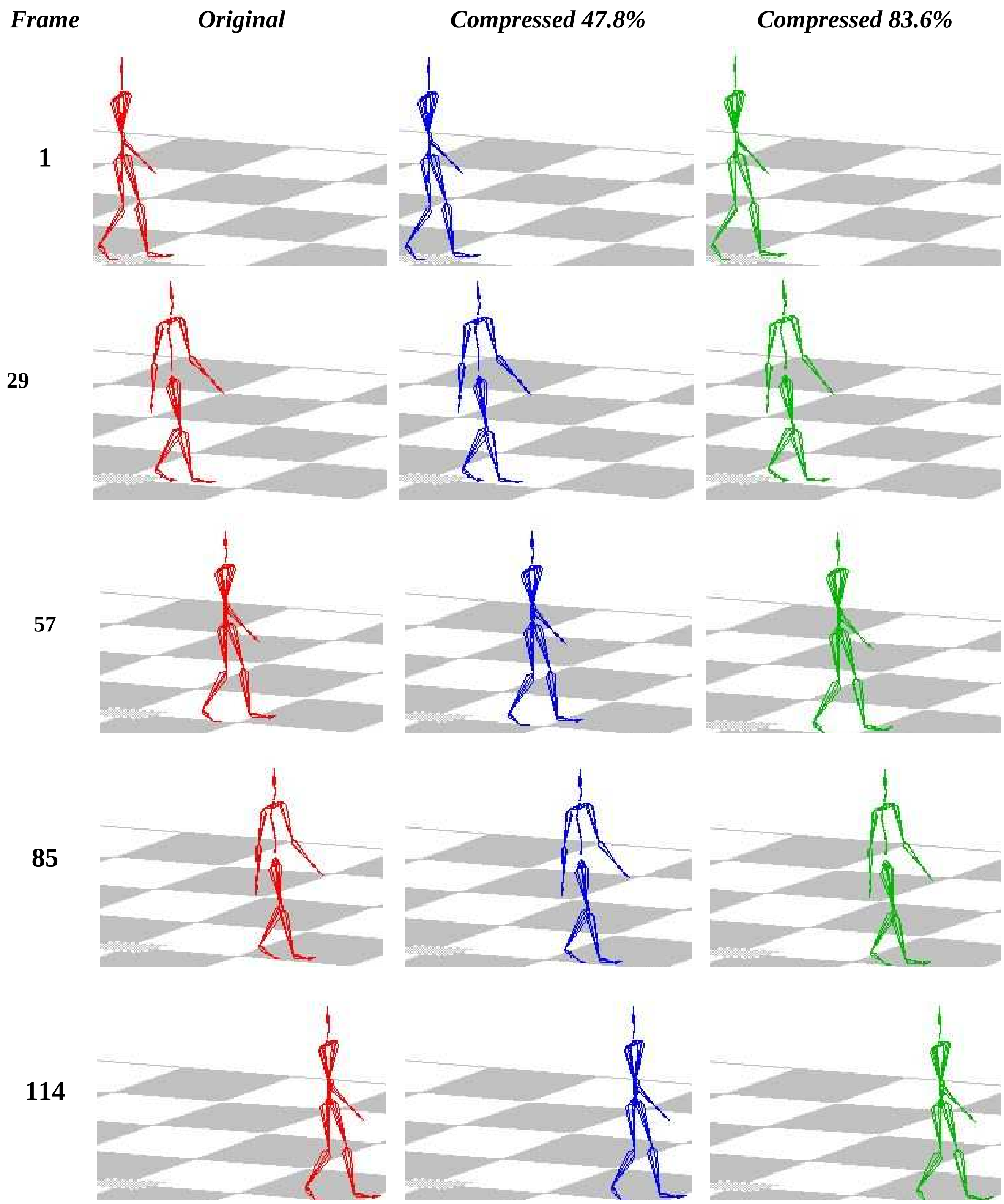


Figure 6: Snapshots of the original (left column), 47.8% compression (middle column), and 83.6% compression (right column) walking motion

The idea of the adaptive technique is to compress each signal (or DOF curve) based on its own behaviour and frequency contents, and controlled by a limited error. The advantage of using this adaptive technique is that it achieves a reasonable overall compression ratio with a limited error within each signal. This improves the resulting animation by reducing the visual effects that may result from the compression process and reduces the other quantitative error measures as well.

For evaluating the resulting animation, although the visual evaluation (qualitative) is the most important one in animation, as it is actually what the user will see, we have introduced two other quantitative error measures. The difference in 3D positions of joints is introduced as a general error measure that indicates how the joint's positions of the compressed motion are shifted from the original motion. The other measure - which is dependent on the motion type and in our case is specific to the walking and running motions- is the vertical feet position.

Wavelet transform has been used for its superior properties over some other popular transforms as discussed in section 2. One of the important properties is the reduced computation complexity and the ability of handling non-periodic, non-stationary signals.

As shown in the results, the compressed motion is visually very close to the original one even with high compression ratios (up to 86% with error $< 0.8\%$). These results are useful in many animation areas such as building a large database of animation clips for motion blending or interpolation. It is also potentially useful in on-line gaming and networked virtual environments where the volume of the data is critical due to the available transmission rates and bandwidths. So, performing such compression can help to reduce the data volume, speed up the transmission and most importantly maintain the realistic appearance of the motion with reduced cost.

The proposed adaptive technique can be improved by taking into account the variable sensitivity of the overall error to each joint's error. For example, the error in the root or hip joints has much more effect than the error in the ankle (as the error will propagate through the hierarchy); so, using an adaptive threshold for each DOF may improve the overall results.

References

1. D.J. Wiley and J.K. Hahn, "Interpolation synthesis of articulated figure motion", IEEE Computer Graphics and Application, 17(6), November 1997.

2. A. Witkin and M. Kass, "Spacetime constraints", In Proceedings of ACM SIGGRAPH, pp. 159-168, August 1988.

3. M. Cohen, " Interactive Spacetime Control for Animation", In Proceedings of ACM SIGGRAPH, pp 293-302, July 1992.

4. M. Gleicher, "Motion editing with spacetime constraints", In Proceedings 1997 symposium on Interactive 3D Graphics, pp 139-148, April 1997.

5. M. Unuma, K. Anjyo, and R. Takeuchi, "Fourier principles for emotion-based Human Figure Animation", In Proceedings of ACM SIGGRAPH, pp 91-96, August 1995.

6. Andrew Witkin and Zoroan Popovic, "Motion Warping", In Proceedings of ACM SIGGRAPH, pp 105-108, August 1995.

7. Armin Bruderlin and Lance Williams, "Motion Signal Processing", In Proceedings of ACM SIGGRAPH, pp 97-104, August 1995.

8. C. Rose, M. Cohen, and B. Bodenheimer, "Verbs and Adverbs: Multidimensional Motion Interpolation", IEEE Computer Graphics and Applications, Vol. 18, No. 5, pp32-40, 1998

9. C. Rose, P. Sloan, M. Cohen, "Artist-Directed Inverse-Kinematics Using Radial Basis Function Interpolation", Eurographics proceedings, Vol. 20, No. 3, 2001.

10. A. Finkelstein and D. H. Salesin, "Multiresolution Curves", In proceedings of ACM SIGGRAPH, pp 261-268, 1994.

11. E.J. Stollnitz, T. De Rose and D.H. Salesin, "Wavelets for Computer Graphics: Theory and Application", Morgan Kaufmann Publishers Inc. 1996.

12. Liu, Z.C., Gortler, S.J., and Cohen, M.F., "Hierarchical spacetime control", SIGGRAPH'94, pp35-84, 1994.

13. W. Sun, "Modelling Bipedal Locomotion using Wavelets for Figure Animation", PhD thesis, Department of Computer and Information Sciences, De Montfort University, UK, May 2000.

14. Adham R. Ismail, Shihab S. Asfour, "Discrete wavelet transform: A tool in smoothing kinematic data", Journal of Biomechanics, Vol. 32, pp317-321, 1999.

15. Amr Ahmed, Farzin Mokhtarian, Adrian Hilton, "Parametric Motion Blending through Wavelet Analysis", Short Presentations Proceedings of Eurographics 2001, pp347-353, ISSN 1017-4656

16. Z. Huang, R. Boulic, and D. Thalmann, "Motion Editing Using Multiresolution Filtering", In Proceedings of MMM99, pp323-334, 1999.

17. J. Lee and S. Y. Shin, "A Hierarchical Approach to Interactive Motion Editing for Human-like Figure", SIGGRAPH, pp39-48, 1999.

18. Alexander D. Pou. and Boca Raton, Fla., "The Transforms and applications handbook", CRC Press in cooperation with IEEE Press, 1996.

19. Wei Sun, Gordon J Clapworthy, "A Wavelet-Based Description of Bipedal Locomotion for Use in Virtual Worlds", 5th International Conference on CAD&CO, 1997, China.

20. Ali N. Akansu and Richard A. Haddad, "Multiresolution Signal Decomposition: Transforms, Subbands, Wavelets", Academic Press Inc., 1992.

21. Yves Meyer, "Wavelets: Algorithms & Applications", SIAM, 1993.

22. Albert Cohen and Robert D. Ryan, "Wavelets and Multiscale Signal Processing", Chapman & Hall, 1995.

23. C. Sidney Burrus, Ramesh A. Gopinath and Haitao Guo, "Introduction to Wavelets and Wavelet Transform: A Primer", Printice-Hall, Inc. 1998.

24. Kenji Amaya, Armin Bruderlin, and Tom Calvert, "Emotion from Motion", Graphics Interface, pp222-229, 1996.