# Key Frame Extraction from Motion Capture Data by Curve Saliency

Eyuphan Bulut
Bilkent University
Department of Computer Engineering
06800 Ankara, Turkey
eyuphan@cs.bilkent.edu.tr

Tolga Capin
Bilkent University
Department of Computer Engineering
06800 Ankara, Turkey
tcapin@cs.bilkent.edu.tr

## ABSTRACT
We propose a new method for extracting key frames from a motion capture sequence. Our proposed approach consists of two steps. In the first step, we propose a new metric, curve saliency, for motion curves that specifies the important frames of the motion. In the second step, we detect the final key frames by clustering the computed important frames. As a result of our experimental results, on the average, by using only 3.7% of all frames as key frames, we can represent the captured motion sequence

## Categories and Subject Descriptors
I.3.7 [**Computer Graphics**]: Three-Dimensional Graphics and Realism: Animation, *Visible line/surface algorithms.*

## General Terms
Algorithms, Management, Performance, Design, Economics, Theory

## Keywords
Virtual human animation, motion capture, key frame extraction

## 1. INTRODUCTION
In the last decade, we have witnessed the rising significance of motion capture for several applications. Movies and games have started to widely use motion capture systems. At the same time, the storage and transmission of motion capture content has become a problem due to their tremendous size. The need for more compact representation has lead researchers to investigate the ways of handling these large data.

Another problem with motion capture is editing of motion sequences. Different from other animation techniques (such as procedural and keyframe animation), it is difficult to edit a motion capture content. Various researchers have proposed solutions, such as stylization, to modify an input motion sequence.

Keyframing is one of the most effective ways of achieving this. The important frames of a motion are selected to be the key frames and the others are computed via the interpolation techniques by using the key frames. To edit a motion capture sequence, this emerges a new problem, "*which frames of the motion will we choose as the key frames*?" Up to now, there have been a number of approaches proposed for the solution of this problem. They essentially differ from each other in terms of the way that they treat motion sequences. We compare these solutions in the related work section.

In this paper, we propose a new approach to find key frames in a motion captured sequence. We treat the input motion sequence as a curve, and find the most salient parts of this curve which are crucial in the representation of the motion behavior. We apply the idea of *saliency* to motion curves in the first part of our algorithm. Then in the second part, we apply key frame reduction techniques in order to obtain the most important key frames of the motion.

The remainder of the paper is organized as follows: Section 2 reviews previous research. In Section 3 we introduce our proposed approach. In Section 4 we present our experimental results. Finally, in Section 5 we summarize our approach for key frame extraction.

## 2. RELATED WORK
There have been various proposed solutions to keyframe extraction in literature. The techniques used mainly fall into three categories: Curve Simplification, Clustering, and Matrix Factorization. The basic approaches of each of these techniques are as follows:

*Curve simplification:* In this method, the motion sequence is represented as a trajectory curve in high-dimensional feature space and the curve simplification algorithms are applied to these trajectory curves. The extracted key frames are the junctions between simplified curve segments.

*Clustering:* Motions are defined with feature sets and the frames of the motion data are clustered in terms of these features. Key frames are selected by selecting a frame in each cluster. (i.e. [12])

*Matrix Factorization:* The frames of motion data are represented as matrices such as feature frame matrices formed by color histograms of frames. Then by using techniques such as singular value decomposition (SVD) [7] and low-order discrete cosine term (DCT) [10], the summary of the motion is constructed.

These three categories differ from each other in terms of the representation of motions and there are various works done in each category. Our algorithm belongs to the first approach, therefore we will survey the previous approaches for curve simplification in detail, and analyze the advantage of our algorithm over these methods.

An initial work of curve simplification belongs to Lim and Thalmann [3]. Their method uses Lowe's algorithm [6] for curve simplification, which represents the values of a single joint over a motion sequence. Starting with the line which combines the start

and end points of the curve, the algorithm divides the line into two sub-lines, if the maximum deviation of any point on the curve from the line is larger than a certain error rate. The algorithm does the same process recursively for each sub-line, until the error rate is small enough for each sub-line. Figure 1 illustrates this idea.

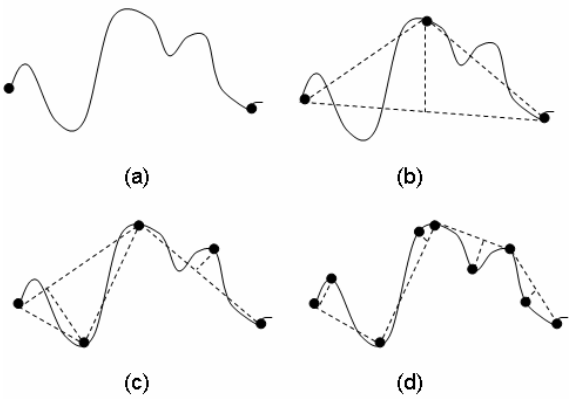


**Figure 1. The steps of curve simplification method used in [3]. From (a) to (d) the line segment is separated from the point which has the longest distance to the line.**

Another approach that aims to find the key frames by dealing with motion curves is presented in the work of Okuda et al. [4] [5]. This approach detects the key frames in motion capture data by using frame decimation. The frames are decimated one by one, according to their importance. When a desired number of key frames are obtained, the algorithm stops.

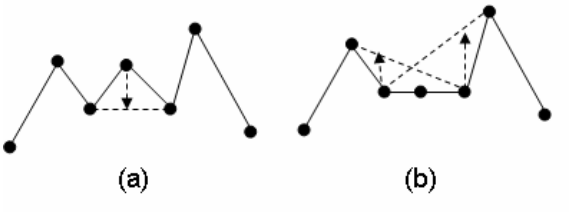


**Figure 2. Decimation of a frame and further updates done for neighbors**

In order to handle the results of each curve (representing a degree of freedom), the authors define a weight function, W(j) which is the weight of $j^{th}$ curve. Then the total error is defined as

$$E^n(k)=\Sigma\ W(j)\ D(j,k) \quad (1)$$

The algorithm deletes the frame with the minimum $E^n(k)$. And the frames are decimated until the number of frames reaches to the desired number of key frames. This approach gives the near-to-optimal result, however it has a higher complexity due to the calculation of errors at each step.

Another work is Matsuda and Kondo's approach [9]. First, the proposed solution finds the fixed frames of the motion which satisfy one of the following:

- Local minimum or maximum value
- One of the end points of a straight line
- A point that has "large angle difference" on both sides. That is, the point which is at least 50% of the amplitude far away from the neighbor frames.

Having the fixed frames of the motion that can not be deleted, the authors apply reduction operations to the other characteristic frames and find the key frames of the motion. However, this method is not optimal, e.g. on the average, 55% of all frames are selected to be key frames of a motion.

# 3. OUR APPROACH FOR KEYFRAME EXTRACTION

Our approach consists of two main steps. The first step is applying a curve saliency metric to the motion curves to measure the importance of each frame. This results in a number of candidate key frames. The second step is reducing the number of candidate key frames by applying clustering methods and selecting only sufficient number of frames from each cluster.

Lee et al. [8] have introduced the approach of mesh saliency to computer graphics. Their work aims to find a metric to define the most important parts of an input mesh that could be used in mesh simplification and viewpoint selection.

The mesh saliency of a vertex v is calculated as follows:

$$S(v) = |G(C(v),\sigma)-G(C(v),2\sigma)| \quad (2)$$

In other words, the saliency of a vertex is defined as the absolute difference value between the Gaussian weighted averages computed at fine (σ) and coarse (2σ) scales.

In the above formula, C is a curvature map from each vertex of a mesh to its mean curvature and C (*v*) denotes the mean curvature of vertex *v*.

## 3.1 Curve Saliency

Inspired from mesh saliency approach, we compute the *curve saliency* of each point (i.e. frame) in the curve in order to find its importance for the representation of the motion.

The *curve saliency* of each point is computed as follows. First, we compute the Gaussian weighted average value of a point assuming a Gaussian distribution with mean 0 and standard deviation σ and centered at that point. Then we calculate a similar value with standard deviation 2σ. The *curve saliency* value for that point then becomes the absolute difference of these two values. If a point is significant for the motion, it is due to its location on the curve. That is, if its value shows a remarkable change according to the values of neighboring points, it is an important point. Therefore, if a remarkable change occurs in the value of the point between the results of the two Gaussians, then its *curve saliency* has a higher value. Figure 3 shows the saliency values on a sample motion curve.

After the calculation of the saliency value for each point on the curve, we define the points having a saliency value greater than average saliency for the motion as *candidate key frames*. Figure 4 shows the result of this process on two sample motion curves.

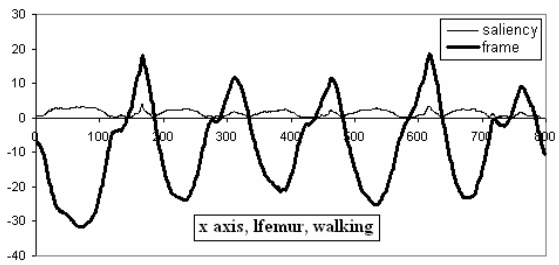


**Figure 3. Saliency values of frames are indicated in the motion curve of x axis angle of left upper leg joint in walking action**

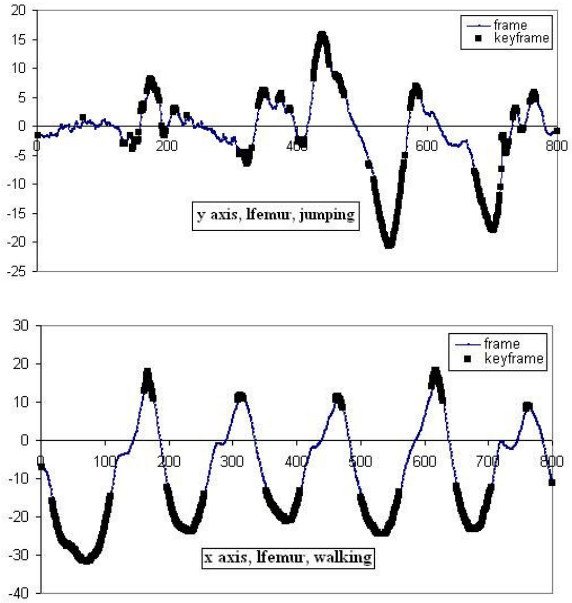


**Figure 4. Candidate key frames are indicated in the motion curves of (above) y axis angle of left upper leg joint in jumping action (below) x axis angle of left upper leg joint in walking action.**

## 3.2 Reduction

Selecting the frames that have higher curve saliency values than the average saliency value results an excessive number of candidate keyframes. Since not all of them are needed in order to represent the characteristics of the motion, we select the important ones among these candidate keyframes.

As it is seen in Figure 3, the candidate key frames form clusters. Let F be the set of all frames and $F_{keyframe}$ be the set of candidate keyframes. Then they are formulated as follows:

$$F = \{f_1, f_2, f_3, \dots f_{n-1}, f_n\}$$

$$F_{keyframe} = \{f_1, f_{k1}, f_{k1+1}, f_{k1+2,} \dots f_{k1+m1}, f_{k2}, f_{k2+1}, f_{k2+2,} \dots f_{k2+m2}, \dots\dots\dots f_{kn}, f_{kn+1}, f_{kn+2,} \dots f_{kn+mn}, f_n\} \quad (3)$$

The set of candidate keyframes include groups of consecutive frames, which we can treat as clusters. When we consider these clusters independent from each other, we notice that one frame can easily reflect the characteristic of the frames in the cluster. Therefore, we only select the frame with local maximum or local minimum value among the frames in the cluster.

$$F_{cluster\ (i)} = f_{ki}, f_{ki+1}, f_{ki+2} \dots f_{ki+mi}$$

$$f_{ki} \le f_{ki+1} \le \dots \le f_{ki+t} \ge f_{ki+t+1} \ge \dots \ge f_{ki+mi} \quad (4)$$

$$f_{ki} \ge f_{ki+1} \ge \dots \ge f_{ki+t} \le f_{ki+t+1} \le \dots \le f_{ki+mi} \quad (5)$$

The situations illustrated in (4) and (5) result the selection of frame $f_{ki+t}$ as keyframe.

Figure 5 shows the local minimum and maximums on a sample curve.

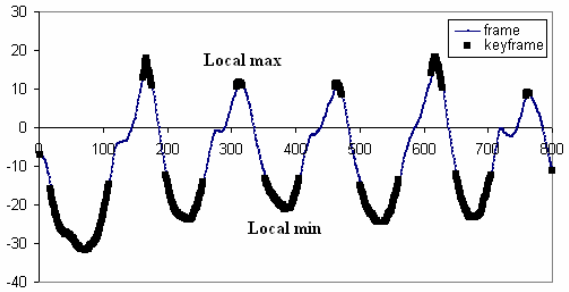


**Figure 5. Decrease the number of candidate key frames by only selecting local minimums or local maximums from each cluster.**

After we apply the reduction of candidate key frames as described above, the resulting number of key frames becomes satisfactory. Figure 6 shows the result of the first reduction operation.

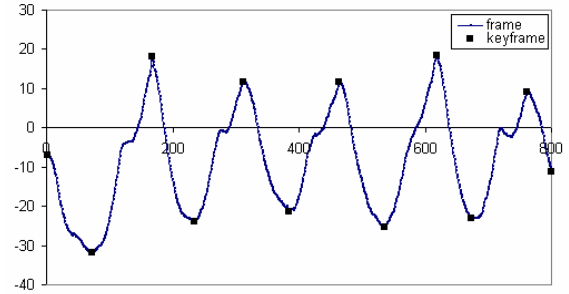


**Figure 6. Decrease the number of candidate key frames by only selecting local minimums or local maximums from each cluster**

Since the joints are represented by three angles, there occur three different sets of key frames for each joint. Let $F_{keyframe\text{-}X}$, $F_{keyframe\text{-}Y}$ and $F_{keyframe\text{-}Z}$ be the keyframe sets of each angle space.

$$F_{keyframe\text{-}X} = \{f_{x1}, f_{x2}, f_{x3}, \dots f_{xm1}\}$$

$$F_{keyframe\text{-}Y} = \{f_{y1}, f_{y2}, f_{y3}, \dots f_{ym2}\}$$

$$F_{keyframe\text{-}Z} = \{f_{z1}, f_{z2}, f_{z3} \dots f_{zm3}\} \quad (6)$$

Since all angle spaces are equally important for the motion, we should consider equal contribution of decided keyframes of each angle space to the final list of keyframes of the whole motion. Therefore, as the next step we combine all the decided key frames of each angle by taking union of them. Since the keyframes are

selected for each angle curve independently, combination of them may result closer keyframes. (i.e. 10 frames) That is why, as the final step, we delete the least important key frames among the ones that are close to each other. Figure 7 shows the combination of all key frames and the final key frames after the deletion of closer ones on the curve of x axis angle.

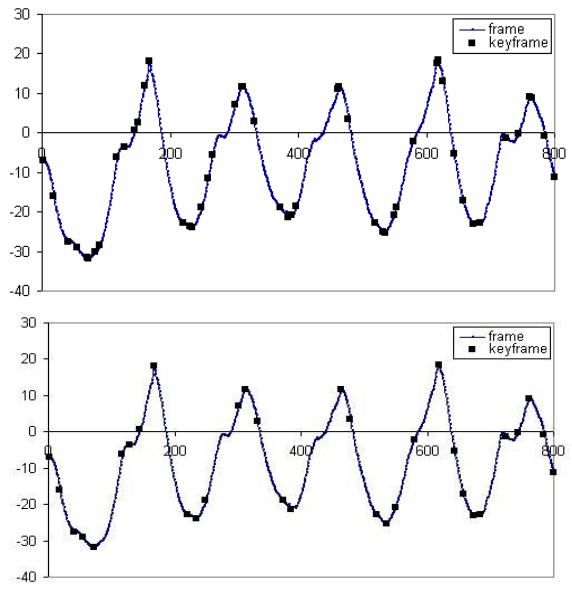


**Figure 7. Combination of all key frames (above) and the final key frames after deletion of close key frames (below) are shown on x axis angle curve of left upper leg joint in walking motion.**

# 4. RESULTS

For experimental results, we have used the motion capture database from CMU [11]. As the first step, we have tested our algorithm on three different motions; walking, jumping and playing with a sword. In general, we have used 800 frames of the motion captured sequences.

Table 1 shows the number of key frames decided for each axis angle independent from each other and as a total in these three actions.

**Table 1. Number of key frames for each axis angle and total in three different actions.**

| | Walking | Jumping | Sword |
|---|---|---|---|
| x | 13 | 16 | 7 |
| y | 31 | 31 | 22 |
| z | 12 | 58 | 9 |
| Total | 51 | 97 | 31 |

Furthermore, when we apply the last step of reduction of key frames on these motions, we obtained the results in Table 2. On the average, we achieve a compression rate of 27 times for the motions.

**Table 2. Final number of key frames decided after reduction of close key frames**

| Graphics | Top | In-between | Bottom |
|---|---|---|---|
| Before | 51 | 97 | 31 |
| After | 33 | 36 | 25 |
| All | 800 | 800 | 800 |
| Ratio | 4.1% | 4.5% | 3.1% |

We have also computed the mean absolute error rate of our algorithm using the below formula:

$$E= [(\Sigma |F^o(k) - F^r(k)|^2 )] / N \quad (7)$$

Here $F^o(k)$ and $F^r(k)$ are the values of joints in original and reconstructed motion respectively. Figure 8 shows the comparison of this error with two previous works (Frame decimation [3] and curve simplification [5]) in a sample walking motion.

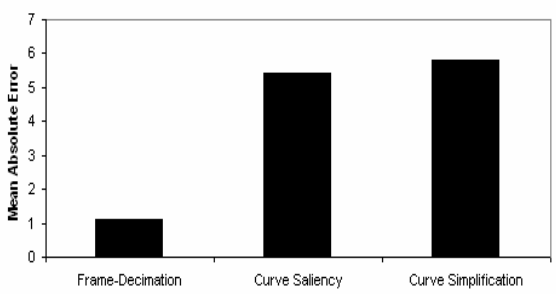


**Figure 8. Comparison of Mean Absolute Errors in three methods**

Our algorithm creates an error rate closer to [3] but remarkable bigger than [5]. This is an expected result because [5] creates the optimum keyframe selection that achieve minimum error. However, our algorithm overcomes both methods in terms of computation. Our algorithm does not only provide a method for keyframe selection, but also shows an efficient way of achieving it. Moreover, since our algorithm decides keyframes by looking the saliency metric of frames in a window of closer frames, it does not need all frames of the motion to decide whether it is a keyframe or not. Therefore our algorithm can achieve fast and efficient keyframe selection in streaming and real-time motions.

# 5. CONCLUSION

In this paper, we have proposed a new approach for key frame extraction from motion capture sequences. We find the most important points of the motion curves via computation of a new curve saliency metric. Curve saliency is computed simply by taking the absolute difference between the Gaussian weighted averages of each point computed at fine ($\sigma$) and coarse ($2\sigma$) scales. Obtaining the candidate key frames from this approach; we get rid of redundant key frames with reduction operations. Based on the experimental results, motion captured sequences can be represented by only 3.7% of all captured frames.

## 6. ACKNOWLEDGMENTS

This work was supported by EC FP6 3DTV Project (Grant no: FP6-511568) and Bilkent University Research Development Grant.

## 7. REFERENCES

[1] K. Huang, C. Chang, Y. Hsu, S. Yang. "KeyProbe: A Technique for Animation Keyframe Extraction" The Visual Computer, Volume 21, pp. 532-541, 2005

[2] Assa, J., Caspi, Y., Cohen-Or, D. "Action synopsis: Pose selection and illustration." ACM Transactions on Graphics (SIGGRAPH) 24(3), 667–676 (2005)

[3] S. Lim and D. Thalmann, "Key-posture extraction out of human motion data by curve simplification", In Proceedings of 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC2001), vol.2, pp. 1167-1169, 2001

[4] S. Li, M. Okuda, S. Takahashi, "Embedded Key Frame Extraction for CG Animation by Frame Decimation", ICME, 2005

[5] H. Togawa, M. Okuda, "Position-Based Keyframe Selection for Human Motion Animation", International Workshop on Network-based Virtual Reality and Tele-existence

[6] D. G. Lowe. "Three-dimensional object recognition from single two dimensional images Artificial Intelligence", 31:355-395, 1987.

[7] Gong, Y., Liu, X: "Video summarization using singular value decomposition". In Computer Vision and Pattern Recognition, pp 174-180 (2000)

[8] C. H. Lee, A. Varshney, and D. W. Jacob. Mesh saliency. *ACM Trans. on Graphics*, 659-666, 2005.

[9] Matsuda K., Kondo K., "Keyframes Extraction Method for Motion Capture Data" Journal for Geometry and Graphics, Volume 8 (2004), No.1 81-90

[10] Cooper, M., Foote, J.: Summarizing video using non-negative similarity matrix factorization. In: IEEE Workshop on Multimedia Signal Processing, pp. 25– 28 (2002)

[11] http://mocap.cs.cmu.edu

[12] Liu, F., Zhuang, Y., Wu, F., Pan, Y.: 3d motion retrieval with motion index tree. Comput. Vis. Image Understand. 92(2), 265–284 (2003)