

# Segmentation and Recognition of Multi-Attribute Motion Sequences\*

Chuanjun Li Peng Zhai S.Q. Zheng B. Prabhakaran

Department of Computer Science  
University of Texas at Dallas, Richardson, TX 75083

{chuanjun, pxz024000, sizheng, praba}@utdallas.edu

## ABSTRACT

In this work, we focus on fast and efficient recognition of motions in multi-attribute continuous motion sequences. 3D motion capture data, animation motion data, and sensor data from gesture sensing devices are examples of multi-attribute continuous motion sequences. These sequences have multiple attributes rather than only one attribute as time series data has. Motions can have different rates and durations, and the resulting data can thus have different lengths. Also, motion data can have noises due to transitions between successive motions. Hence, traditional distance measuring approaches used for time series data (such as Euclidean distances or dynamic time-warped distances) are not suitable for recognition in multi-attribute motion sequences. Hence, we have defined a similarity measure based on the analysis of singular value decomposition (SVD) properties of similar multi-attribute motions. A five-phase algorithm has then been proposed that gives good pruning power by exploiting the proximity of continuous motion data. We experimented this algorithm with data from different sources: 3D motion capture devices, animation motions, and CyberGlove gesture sensing device. These experiments show that our algorithm can segment and recognize long motion streams with high accuracy and in real time without knowing beforehand the number of motions in a stream.

**Categories and Subject Descriptors:** I.5.3 [Pattern Recognition]: similarity measures

**General Terms:** Algorithm

**Keywords:** Pattern recognition, multi-attribute motion, gesture, segmentation, singular value decomposition.

## 1. INTRODUCTION

Multi-attribute motion data is encountered in several applications/devices. For instance, a gesture sensing device such as CyberGlove has several sensors that transmit values to indicate motion of a hand. Devices track the movements

\*Work supported partially by the National Science Foundation under Grant No. 0237954 for the project CAREER: Animation Databases.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MM'04, October 10-16, 2004, New York, New York, USA.  
Copyright 2004 ACM 1-58113-893-8/04/0010...\$5.00.

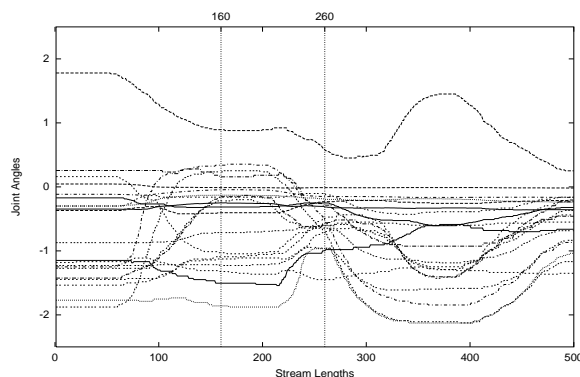


Figure 1: Multi-attribute motion data

of sensors in different axes to capture the motion of a model in virtual reality type of applications. If we consider 3D animations, motion of a model typically involves translation and rotation for different *nodes* of the model. In many of these cases, more than one value is generated at each sampling time, and the generated motion data arrives continuously. For example, Figure 1 describes data from CyberGlove with 22 sensors. Here, the hand motions are based on American Sign Language (ASL) for hearing impaired. The motion consisting of the first 160 samples describes the ASL sign TV, and the second motion shown as samples 260-500 is for ASL sign MILK. The data between sample 160 and sample 260 is transitional noise.

Recognizing patterns in such continuous multi-attribute data or stream data also involves segmentation, i.e., identifying the beginning and the end of a pattern. Segmentation and recognition of stream data pose several challenges:

- A proper similarity definition should consider data sequences of multiple variables or attributes, because data of multiple attributes are aggregate data, and should be considered together to make the motions meaningful.
- Motion patterns are of variable lengths, and similar motions are also of different lengths. Stream data sequences can have some transitional noises (from one motion to the other) which should not be recognized separately as any known pattern motions.
- Since lengths of both stream data and pattern data can be hundreds or even thousands of rows, similar-

ity measure should scale well with the sizes of both stream and pattern data. For instance, 3D motion capture data in [12] have anywhere from 200 to 9000 rows and approximately 60 columns. Even animated simple motions of 3D models can have around 300 rows and 50 columns. Scalability is especially important in the cases where the generated multi-attribute motion sequences are to be recognized in real-time.

- The number of motions in a motion stream may not be known beforehand.
- Since the number of patterns can be very large, non-promising patterns should not be involved in the similarity computation.

**Proposed Approach:** In this paper, we explore the singular value decomposition (SVD) properties of motion matrices, and define a similarity measure based on the SVD properties. We show that this approach effectively segments and recognizes multi-attribute data from: 22-sensor CyberGlove, 3D motion capture data [12], and 3D animation motion data. We also show that the fast execution time of this approach makes it suitable for segmenting and recognizing motion data generated in real-time. We compare the proposed work with related work in Section 6.

## 2. MULTI-ATTRIBUTE MOTION REPRESENTATION

The motion of a subject  $S$  can be identified by the combined motions of  $n$  sampling points  $s_j$  ( $j = 1, 2, \dots, n$ ) on the moving subject  $S$ . For instance, motion data is obtained from a sensor at each sampling point  $s_j$  on the CyberGlove. The motion data generated for a sampling point  $s_j$  at a time  $t_i$  ( $i = 1, 2, \dots, m$ ) can be of different types:

- one degree of freedom (DOF), for example the joint angle data generated by CyberGlove.
- three DOFs. Data includes the  $x, y$  and  $z$  coordinates of the sensor at  $s_j$ .

Hence, motion data can be conveniently represented using a matrix. A row of this matrix represents information from all sampling points at a sampling time  $t_i$  and a column of the matrix represents information from a sampling point  $s_j$ . Since similar motions might have different moving speeds, the matrices of similar motions may have different lengths.

## 3. SIMILARITY MEASURE USING SVD

The known motion patterns used as references for comparison are represented using different matrices  $P_k$ ,  $k = 1, 2, \dots, p$ , where  $p$  is total number of motion patterns in a database. Let  $Q$  be the motion data segmented for recognition. Now, we analyze the SVD properties of motion matrices in order to derive a reasonable similarity measure.

As proved in [7], any real  $m \times n$  matrix  $A$  can be decomposed as  $A = W\Sigma Z^T$ , where  $W = [w_1, w_2, \dots, w_m] \in R^{m \times m}$  and  $Z = [z_1, z_2, \dots, z_n] \in R^{n \times n}$  are two orthogonal matrices, and  $\Sigma$  is a diagonal matrix with diagonal entries being the singular values of  $A$ :  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min(m,n)} \geq 0$ . Column vectors  $w_i$  and  $z_i$  are the  $i^{th}$  left and right singular vectors of  $A$ , respectively.

Both the matrix to be recognized  $Q$  and the reference patterns  $P_k$  typically have many more rows  $m$  than columns  $n$ , and computing SVD of an  $m \times n$  matrix takes  $O(mn^2)$  time and is thus costly. Hence, instead of computing SVDs of  $Q$  and  $P_k$  directly, we map  $Q$  and  $P_k$  into two  $n \times n$  matrices  $M_1$  and  $M_2$  and then compute SVDs for  $M_1$  and  $M_2$ . We define  $M_1$  and  $M_2$  as:

$$M_1 = Q^T \times Q \quad M_2 = P_k^T \times P_k$$

It can be proved that for  $M_1$  and  $M_2$ , the left singular vectors are equal to the corresponding right singular vectors, which are the corresponding right singular vectors of  $Q$  and  $P_k$ , respectively. We can thus decompose  $M_1$  and  $M_2$  as  $M_1 = U\Sigma U^T$ , and  $M_2 = V\Lambda V^T$ , where  $U = [u_1, u_2, \dots, u_n]$ ,  $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$ ,  $V = [v_1, v_2, \dots, v_n]$ , and  $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ . The following convention will be used. We will refer to  $u_i$  of  $U$  as the  $i^{th}$  singular vector of  $M_1$  (corresponding to  $Q$ ) and  $v_i$  of  $V$  as the  $i^{th}$  singular vector of  $M_2$  (corresponding to a reference pattern  $P_k$ ). Also for convenience, the main diagonals of  $\Sigma$  and  $\Lambda$  are referred to as the singular value vectors  $\sigma$  and  $\lambda$  of  $M_1$  and  $M_2$ , respectively.

It should be observed here that the singular values of a matrix are unique, and the singular vectors corresponding to the distinct singular values are also uniquely determined (up to the sign). Although the singular vectors are only unique up to the sign when singular values are distinct, we can consider the SVD of a matrix to be unique. The reason is that for experimental data, positive singular values are practically distinct.

The definitions of  $M_1$  and  $M_2$  indicate that  $u_i$  and  $v_i$  are the same as the corresponding  $i^{th}$  principle components of  $Q$  and  $P_k$ , respectively [18]. Principle components are the coordinate axes along which variances of data are extremes (maxima and minima) obtained by principle component analysis (PCA). Hence maximal or minimal variances of data in  $Q$  and  $P_k$  along  $u_i$  and  $v_i$ . Since the right singular vectors of  $M_1$  and  $Q$  are the same, and the singular values of  $M_1$  are the square of the corresponding singular values of  $Q$  (the same is true for  $M_2$  and  $P_k$ ), hence, if  $M_1 \neq M_2$ , then  $Q \neq P_k$ , i.e., motions represented by  $Q$  and  $P_k$  cannot be similar if  $M_1$  and  $M_2$  are dissimilar to each other.

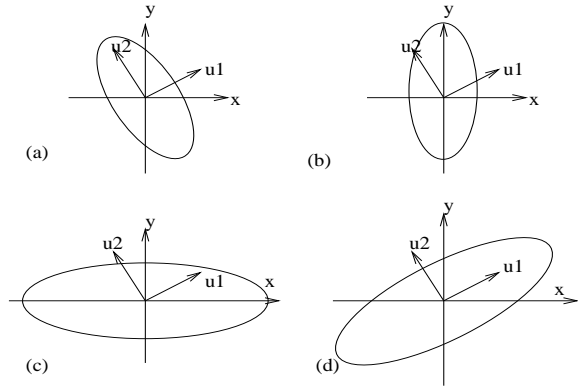
### 3.1 Comparing SVDs

Our intention is to define an appropriate distance measure  $\Psi(Q, P_k)$  to compare the matrix of the motion to be recognized  $Q$  and the matrices of the reference motion patterns  $P_k$ . Since matrices  $M_1$  and  $M_2$  correspond to  $Q$  and  $P_k$ , we now need only to compare the SVDs of  $M_1$  and  $M_2$ .

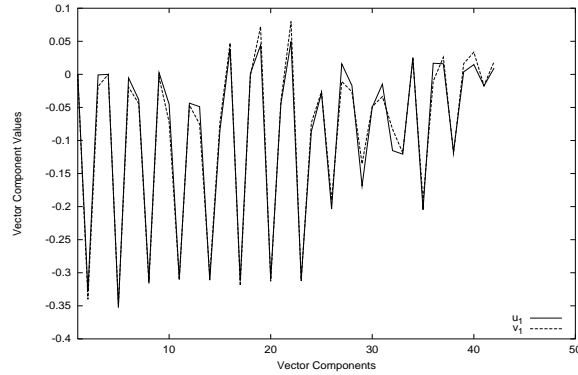
#### *Geometrical Effect of SVD:*

Let us assume that we multiply a column vector  $x$  by matrix  $M_1$ . Earlier, we considered that  $M_1$  has SVD of  $U\Sigma U^T$ . Hence, we have  $M_1 x = U\Sigma U^T x$ . The geometrical effect of the SVD of  $M_1$  on the vector  $x$  is as follows:

- Effect of  $U^T$  is to rotate the point  $x$  from the coordinate system spanned by column vectors of  $U$  to the standard coordinate system by the angles between corresponding axes of the two systems. Figure 2(b) illustrates the effect of  $U^T$  on the *original* ellipse in Figure 2(a).
- Effect of  $\Sigma$  is to stretch the vector  $U^T x$  by a factor of  $\sigma_i$  in the  $i^{th}$  axis direction of the standard coordinate



**Figure 2: Ellipse geometric changes caused by a 2-by-2 matrix**



**Figure 3: First singular vectors  $u_1$  and  $v_1$  for similar motions by two different subjects:  $u_1 \cdot v_1 = 0.995$**

system. Figure 2(c) describes the effect of  $\Sigma$ .

- Finally  $U$  rotates vector  $\Sigma U^T x$  from the standard coordinate system into the ortho-normal basis spanned by the column vectors of  $U$  by the angles between the corresponding axes of the two systems. The final ellipse is shown in Figure 2(d).

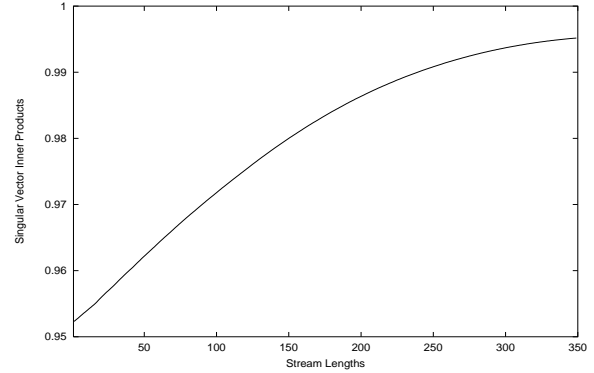
Similar motions should have similar trajectories, hence the geometric structures of the motion matrices as exposed by SVD should be similar. Hence, if  $Q$  and  $P_k$  are for similar motions, then the geometrical shapes of the hyperellipsoid  $E_1 = \{M_1 x : |x| = 1\}$  and  $E_2 = \{M_2 x : |x| = 1\}$  will be similar (following the arguments used for Figure 2).

### 3.2 Similarity Measure $\Psi(Q, P_k)$

For computing the similarity between the motion to be recognized  $Q$  and a pattern  $P_k$ , we consider the effects of SVDs of their  $n \times n$  matrices  $M_1$  and  $M_2$ . In this section, we argue that we need not consider all the SVD components for computing similarity of  $M_1$  and  $M_2$ . We identify the SVD components that have significant influences on similarity and derive a similarity measure that expresses the similarity of  $M_1$  and  $M_2$ .

1. **Effect of  $u_1$  and  $v_1$ :** When  $Q$  and  $P_k$  represent similar motions, the inner products  $|u_1 \cdot v_1|$  is close to 1. The reason is  $|u_1 \cdot v_1| = |u_1| |v_1| |\cos(\theta)| \doteq |u_1| |v_1|$ ,

since the angle  $\theta$  between the two vectors  $u_1$  and  $v_1$  approaches zero or  $\pi$  as Figure 3 indicates when  $Q$  and  $P_k$  are similar. Also, the norms  $|u_1|$  and  $|v_1|$  are 1 by the properties of singular value decomposition.  $|u_1 \cdot v_1| \doteq 1$  holds if  $Q$  and  $P_k$  are for similar motions.  $u_1$  changes as more and more motion data is collected for  $Q$  from the motion being recognized. The inner product  $|u_1 \cdot v_1|$  changes accordingly before it reaches the maximum as shown in Figure 4. However, the inner products of other singular vectors can reach their maxima even before the matrix  $Q$  is completely available in order to be recognized as some pattern  $P_k$  as shown in Figure 5.



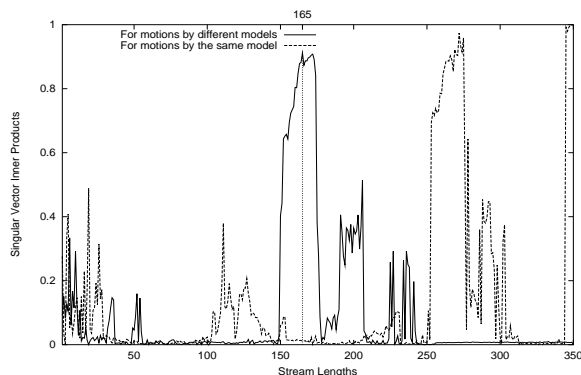
**Figure 4: Changes in inner products of the 1<sup>st</sup> singular vectors of similar motion data by different subjects**

Geometrically speaking, the matrix  $M_1$  stretches a vector  $x$  by the largest singular value  $\sigma_1$  in the direction of its first singular vector  $u_1$ . The stretching factors can be very small in the other directions. For these reasons, we consider only the effect of the first singular vectors for similarity measure. We do this by computing  $|u_1 \cdot v_1|$ .

2. **Effect of  $\sigma$  and  $\lambda$ :** Let  $\vec{\sigma} = \sigma/|\sigma|$  and  $\vec{\lambda} = \lambda/|\lambda|$ . Following the same arguments used for  $|u_1 \cdot v_1|$ , the dot product of the normalized singular value vectors  $\vec{\sigma} \cdot \vec{\lambda} \doteq 1$ .

Since  $\sigma_1$  and  $\lambda_1$  are positive and larger than other singular values,  $\vec{\sigma} \cdot \vec{\lambda}$  does not approach zero (even though  $|u_1 \cdot v_1|$  does for two different motions). In order for  $\vec{\sigma} \cdot \vec{\lambda}$  to have similar contribution to the similarity measure as  $|u_1 \cdot v_1|$  has, we scale  $\vec{\sigma} \cdot \vec{\lambda}$  by  $\eta$  ( $0 < \eta < 1$ ):  $(\vec{\sigma} \cdot \vec{\lambda} - \eta)/(1 - \eta)$ .  $(\vec{\sigma} \cdot \vec{\lambda} - \eta)/(1 - \eta) = 1$  if  $\vec{\sigma} \cdot \vec{\lambda} = 1$ , and  $(\vec{\sigma} \cdot \vec{\lambda} - \eta)/(1 - \eta) = 0$  if  $\vec{\sigma} \cdot \vec{\lambda} = \eta$ . We set  $\eta = 0.9$  because our experiments show that  $\vec{\sigma} \cdot \vec{\lambda}$  is larger than 0.9.

3. **Combining the Effects:** When  $Q$  and  $P_k$  are for similar motions, both  $|u_1 \cdot v_1|$  and  $(\sigma \cdot \lambda)/(|\sigma||\lambda|)$  approaches 1, so the similarity measure should also approach 1. However, if  $Q$  and  $P_k$  are not for similar motions, both  $|u_1 \cdot v_1|$  and  $(\sigma \cdot \lambda)/(|\sigma||\lambda|)$  may not approach 1, we should ensure the similarity measure does not approach 1. Hence, we define the similarity



**Figure 5: Irregular changes in inner products of the 25<sup>th</sup> singular vectors. The inner product reaches its maximal at stream length 165, rather than at stream length 350 when the stream  $Q$  matches motion  $P_k$  by a different model.**

measure,  $\Psi(Q, P_k)$ , to be a product of both the effects:

$$\Psi(Q, P_k) = |u_1 \cdot v_1| \times (\bar{\sigma} \cdot \bar{\lambda} - \eta) / (1 - \eta) \quad (1)$$

A counter-example can easily show that the triangle inequality required for a metric does not hold for Eq. 1, hence Eq. 1 is a nonmetric measure.

Let  $Q = \bar{U}\Sigma^{1/2}U^T$  and  $\bar{V}\Lambda^{1/2}V^T$ . Then the left singular vectors  $\bar{u}_i = Qu_i/\sigma_i^{1/2}$ ,  $\bar{v}_i = P_kv_i/\lambda_i^{1/2}$ . If motions of  $Q$  and  $P_k$  follow the same path, but in different directions,  $M_1$  and  $M_2$  would be the same, yet  $\bar{u}_i$  and  $\bar{v}_i$  would not be the same. Just as using  $u_1$  and  $v_1$  for  $\Psi(Q, P_k)$ , we use  $\bar{u}_1$  and  $\bar{v}_1$  to further verify whether motions  $Q$  and  $P_k$  are in the same direction. Since  $\bar{u}_1$  and  $\bar{v}_1$  might not be of equal length, we uniformly compute  $n$  elements for each of them, and normalize the  $n$  elements by the corresponding norms of the computed  $n$  dimensional vectors. Let the normalized  $n$  dimensional vectors be  $u'_1$  and  $v'_1$ . If motions of  $Q$  and  $P_k$  follow similar paths but in different directions,  $|u'_1 \cdot v'_1|$  would be close to one, otherwise it would be much less than one.

In other words, the proposed similarity measure serves as a faster way of deciding whether motions are following similar paths. To ensure there are no false matches, we will go through a verification phase as discussed in Section 4.

## 4. FIVE PHASE ALGORITHM

For motion sequences generated in real time, computation of the defined similarity measure cannot be afforded for every pair of matrices  $Q$  and  $P_k$ , especially when there are a large number of reference motions. We propose to compute the similarities of  $Q$  and only those patterns  $P_k$ 's which are possible to have high similarity with  $Q$ . Let  $\delta(x)$  be the variance of the components of first singular vector  $x$ , and  $\rho(y_1)$  be  $y_1/|y|$ , where  $y_1 = \sigma_1$  (or  $\lambda_1$ ), and  $y = \sigma$  (or  $\lambda$ ). If two matrices  $Q$  and  $P_k$  are for similar motions, then  $\delta(u_1)$  should be close to  $\delta(v_1)$  as indicated in Figure 3, and the normalized first singular value  $\rho(\sigma_1)$  of  $Q$  should also be close to the normalized first singular value  $\rho(\lambda_1)$  of  $P_k$ . If  $\delta(u_1)$  is distant from  $\delta(v_1)$ , or  $\rho(\sigma_1)$  is distant from  $\rho(\lambda_1)$ , we can conclude that  $Q$  and  $P_k$  are not for similar motions,

or the similarity of them cannot be high enough to recognize  $Q$  as  $P_k$ , thus  $\Psi(Q, P_k)$  does not need to be computed.

Based on the above discussion, we have proposed the following pattern recognition and segmentation algorithm. This algorithm has five phases:

- *Preprocessing Phase*
- *Early Pruning Phase*
- *Similarity Comparison Phase*
- *Verification Phase*
- *Segmentation Phase*

### Pre-processing Phase:

The pre-processing phase is carried out by the following steps.

1. Compute variance  $\delta(v_1)$ ,  $\lambda$  and  $\rho(\lambda_1)$  for every pattern  $P_k$ . Create a list  $L_\delta = (\delta(v_1), P_k)$  sorted by  $\delta(v_1)$ , and another list  $L_\rho = (\rho(\lambda_1), P_k)$  sorted by  $\rho(\lambda_1)$ .
2. Compute  $\Psi(P_i, P_j)$  for  $i, j = 1, 2, \dots, p$ , and  $i < j$ . Rearrange  $L_\delta$  so that patterns with  $\Psi(P_i, P_j) \geq C$  are grouped together, where constant  $C \doteq 1$  can be set to 0.99 to allow for large variations in motions following similar paths. Compute  $v'_1$  for all patterns in each group of list  $L_\delta$ .
3. Let  $W_0 = 0$ , and initialize the  $n \times n$  elements of  $M_1$  to zeros. Let the set of candidate patterns  $\bar{P}$  be empty, and let  $W$  be the maximum number of samples any single motion can have.

### Early Pruning Phase:

The early pruning phase proceeds by comparing the variance values  $\delta$  and the normalized first singular values  $\rho$ .

4. Collect data of the next  $K$  samples to form a matrix  $\Delta$ .  $M_1 = M_1 + \Delta^T \times \Delta$ ,  $W_0 = W_0 + K$ . Compute SVD of  $M_1$ , then compute  $\delta(u_1)$  and  $\rho(\sigma_1)$  for  $M_1$ , and go to step 5.
5. Search  $L_\delta$  for such patterns that satisfy  $\delta(u_1) - \epsilon_1 < \delta(v_1) < \delta(u_1) + \epsilon_1$ , and search  $L_\rho$  for such patterns that satisfy  $\rho(\sigma_1) - \epsilon_2 < \rho(\lambda_1) < \rho(\sigma_1) + \epsilon_2$ . If such patterns cannot be found, go to step 4, otherwise put the found patterns into  $\bar{P}$  and go to step 6.

$\epsilon_1$  and  $\epsilon_2$  are two similar motion variation tolerances of small positive values for  $Q$  to be recognized as  $P_k$  and can be set by experiments. The smaller the variations of similar motions, the smaller the  $\epsilon_1$  and  $\epsilon_2$ .

### Similarity Comparison Phase:

The similarity comparison phase proceeds by determining  $\Psi(Q, P_k)$  for the  $P_k$  identified in the early pruning phase. This phase proceeds as follows.

6. Compute  $\Psi(Q, P_k)$  for every pattern  $P_k$  in  $\bar{P}$ . Update  $\Psi$  to be the highest  $\Psi(Q, P_k)$ , and let  $P_{max}$  be the corresponding  $P_k$ . If  $W_0 < W$ , go to step 4. Otherwise, go to step 7.

## Verification Phase

The verification phase proceeds in the following manner.

7. If there are no other patterns in  $L_\delta$  grouped with  $P_{max}$ , go to step 8. Otherwise, compute  $u'_1$  for  $Q$ , and compute  $|u'_1 \cdot v'_1|$  for every pattern grouped with  $P_{max}$ , and  $Q$  is recognized as the pattern with the highest  $|u'_1 \cdot v'_1|$  value.

## Segmentation Phase:

Multi-attribute motion data stream can have several concatenated motion patterns. Recognizing a motion pattern implies that the concatenated motion pattern stream has to be segmented. Hence, for segmentation, we carry out the following step.

8. If there is more streaming data, let  $\Psi = 0$ ,  $W_0 = 0$ . Initialize the  $n \times n$  elements of  $M_1$  to zeros, and set  $\bar{P}$  to be empty. Go to step 4 but further collecting of data starts from the sample right after the segmentation point.

## 4.1 Algorithm Complexity

Since the sampling point with the highest similarity is between the band bounded in step 5, correct segmentation and recognition are guaranteed, and similarities are not needed to be computed for the patterns outside the band. For exact matching, we can search the sorted patterns in step 5 for only those patterns whose  $\delta(v_1)$  and  $\rho(\lambda_1)$  are closest to those of  $Q$ . Let  $p$  be the total number of patterns,  $n$  be the number of matrix columns, and  $K$  be the number of samples considered together when updating matrix  $M_1$ . The complexities of the different phases of the algorithm are as follows.

1. *Pre-processing Phase:* It takes  $O(p^2n + pm^2m')$  time, where  $m'$  is the maximum number of rows among all patterns. Since the pre-processing needs to be done only once and can be pre-computed, its computation time can be excluded in real time applications.
2. *Early Pruning Phase:* It involves searching the sorted lists  $L_\delta$  and  $L_\rho$ . Searching the sorted patterns takes  $O(\log p)$  time for the first time. Then, it takes  $O(1)$  time due to the gradual changes in  $\delta(u_1)$  and  $\rho(\sigma_1)$ . Adding  $K$  more rows to update  $M_1$  takes  $O(Kn^2)$  time.
3. *Similarity Comparison Phase:* Computing SVD of  $M_1$  takes  $O(n^3)$  time. Computing the similarity measure takes  $O(n|\bar{P}|)$  time.  $\bar{P}$  is the set of patterns identified for similarity computation during the early pruning phase. (In contrast, [14] needs  $O(n^2p)$  for similarity computation).
4. *Verification Phase:* Computing  $|u'_1 \cdot v'_1|$  takes  $O(n)$  due to the limited number of patterns grouped together with each pattern.
5. *Segmentation Phase:* Initialization takes  $O(n^2)$  time.

Overall, the time complexity after the pre-processing phase is  $O(Kn^2 + n^3)$ .

## 5. PERFORMANCE EVALUATION

We used the proposed algorithm to segment and recognize motions from different sources:

- Streaming data of three Degree of Freedoms (DOFs) describing the changing 3-D positional information of the sampling points of a moving subject. At each sampling time, the 3D world coordinates of each sampling point are collected and put together to form a row of data. We are assuming that the positional data of all sampling points can be collected at each sampling time, so that each row of data has the same number and order of data, and a motion matrix can be formed for a motion.
- Streaming data of one DOF generated by the virtual hand of CyberGlove, a fully instrumented glove that provides up to 22 high-accuracy joint-angle measurements [4].

As Figure 9 shows, when  $Q$  matches a pattern  $P_k$ ,  $|\delta(v_1) - \delta(u_1)|$  is much less than 0.0005, and  $|\rho(\lambda_1) - \rho(\sigma_1)|$  is much less than 0.002. To allow for large variations in similar motions, we double the observed variations and set  $\epsilon_1$  and  $\epsilon_2$  in the early pruning phase to 0.001 and 0.005, respectively in all the experiments. Since we do not prune any patterns which fall into either of the two ranges determined by the variation tolerances, a smaller value of one tolerance will not affect the pruning efficiency very much. Small changes in two variation tolerances will not affect the pruning efficiency very much and will not prune similar motions with large variations either.

### 5.1 Recognition in Data of three DOFs

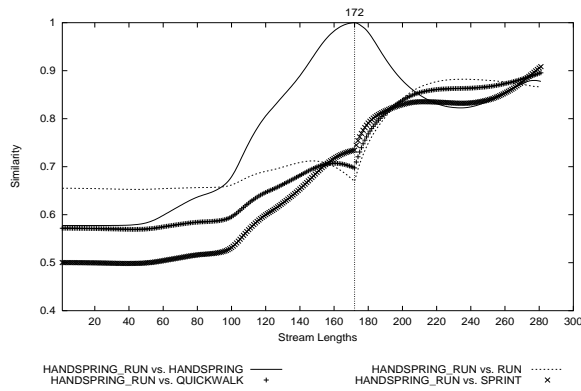
We used both physical model motion capture data and animation motion data to evaluate the performance of the algorithm.

- The physical model motion data used in this paper is made public by the Motion Capture Lab (MoCap Lab) at the Ohio State University [12].
- Animation motion data was extracted from Virtual Reality Modeling Language (VRML) files. VRML is an ISO/IEC standard language to represent 3D virtual worlds on WWW. A VRML virtual world is described by a hierarchical scene graph [17].

#### 5.1.1 Recognizing Physical Model Motion Capture Data of Three DOFs

Here, motions of people with reflective markers of different sizes are captured by several cameras, and the captured 2D marker positions are reconstructed into 3D world coordinates by the optical motion capture system Vicon Workstation software [12]. We used data for 7 distinct motions by 4 subjects with each motion having 19 markers. The 7 captured motions are HANDSPRING, QUICKWALK, RUN, SPRINT, SWAGGER, BREAKDANCER, and DANCEMOTION. The captured motions have 120 samples per second, and each motion lasts for a different duration.

We generated different streaming data by combining any two of the above 7 capture motions, with every other rows of patterns motions being used in streaming data. All motions can be successfully segmented and recognized at correct locations. For example, Figure 6 shows the similarities of three



**Figure 6: Recognition of HANDSPRING in the HANDSPRING\_RUN stream**

patterns and the stream HANDSPRING\_RUN as more data is collected for the stream. Stream HANDSPRING\_RUN is the motion HANDSPRING of length 172 followed by a motion RUN of length 109. As Figure 6 shows, although the reference pattern HANDSPRING has a different length as the HANDSPRING in the stream data HANDSPRING\_RUN, the stream can still be recognized as HANDSPRING at the exact length of 172.

### 5.1.2 Recognizing VRML Model Data of Three DOFs

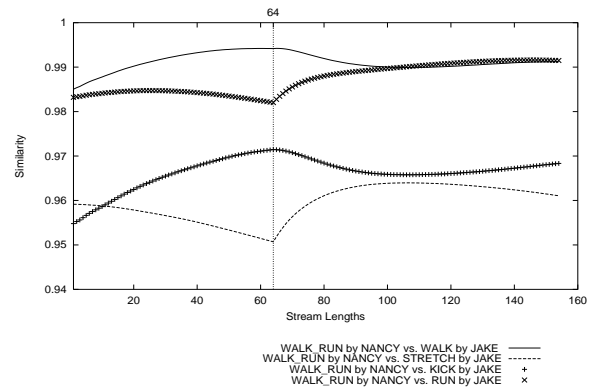
H-Anim is a specification based on VRML to define a standard way of representing interchangeable humanoids in online virtual environments. An H-Anim humanoid model is a hierarchical collection of segments (such as forearm, hand) that are connected to each other by joint (such as shoulder, elbow) [8]. We used two VRML H-Anim models: NANCY and JAKE, created by 3Name3D and Matthew T. Beitler, respectively. These two models have different geometries, and can represent people with different body sizes and proportions. 14 common joints such as shoulders, elbows, wrists, knees, and ankles are chosen as each model’s sampling points.

Four different motions were extracted for each model: STRETCH, KICK, WALK, and RUN. STRETCH and KICK involve the motions of arms and legs of the models in different directions. One set of the four motions is designed for JAKE as reference patterns, and another set of continuous motions with different motion rates are generated for JAKE as the stream data to be separated and recognized. Combinations of the four NANCY motions are taken as stream data. All stream data are for two successive different motions.

When motions in the stream carried out by JACK are to be recognized using the patterns carried out by the same model JACK, although lengths of motions and the patterns can be different, the similarity of a motion in the stream and its corresponding pattern can reach the maximum value one. For the NANCY model, the similarity reaches its highest when the motion in the stream is recognized to be a pattern by the different model JACK as shown in Figure 7. As with the motion capture data, all motions in the stream data of VRML modes can be successfully segmented and recognized.

## 5.2 Recognition in Motion Data of One DOF

In order to evaluate the performance of the algorithm on



**Figure 7: Recognition of the NANCY WALK motion in the NANCY WALK\_RUN stream. Four patterns are generated by the VRML model JAKE**

one DOF motion data, motions including various American Sign Language (ASL) signs were generated using CyberGlove.

For motion patterns, we generated two sets of motion data. Each set comprises one hundred different individual motions, and each motion in one set has a similar motion in the other set. Motions in one set were carried out faster than those in the other set, and the motion durations for each sign in each data set are different. The one hundred different motions in each set include ASL signs for TV, MILK, BUS, 24, 60, 50, 70, 40, 80, 100 and 90 among other ASL signs. We computed the similarity of each motion in one data set and the corresponding motion in the other data set. These two sets of motion data are used as two set of patterns in the database.

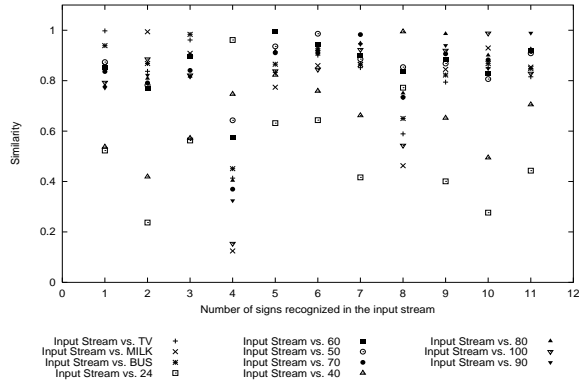
Two continuous motion streams with different speeds and lengths were generated and used as the input stream data. The input stream data includes the above 11 signs mentioned in the order of TV, MILK, BUS, 24, 60, 50, 70, 40, 80, 100 and 90. It should be noted here that *transitional noises* occur between successive ASL signs in the generated input streams as shown in Figure 1. (There were no transitional noisy data in the motion data of three DOFs evaluated above). Figure 8 shows the similarity measures of the short input stream with some of the different reference patterns. The figure also demonstrates how each pattern was identified in the input.

Just as Figure 8 illustrates, individual motions in the two motion streams with different speeds and lengths can be segmented and recognized with 100% accuracy using both sets of motion patterns. It should be observed here that there were no cumulative errors in spite of the following facts:

- Transitional noises were present between two successive sign motions
- Lengths of motions in the stream and patterns were different

## 5.3 Pruning Efficiency of the Early Pruning Phase

The early pruning phase checks patterns for the component variances  $\delta(v_1)$  of the first singular vectors and the



**Figure 8: Similarities of some patterns and the short input stream when signs in the input stream are recognized to be the corresponding patterns with highest similarities.**

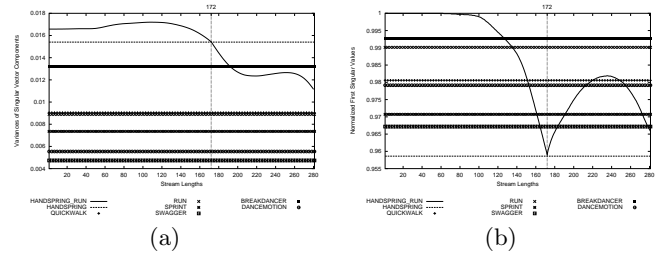
normalized first singular values  $\rho(\lambda_1)$ . If  $\delta(v_1)$  and  $\rho(\lambda_1)$  of some patterns are out of a range of those of the input stream at each stream length, then it is impossible for these patterns to have high similarities with the stream. These reference patterns are considered to be non-promising and hence *pruned* for similarity computations at that time. Figure 9 shows  $\delta(u_1)$  and  $\rho(\sigma_1)$  of the stream at different stream lengths, comparing with those of the 7 patterns of our VRML data. Only three patterns fall into the default ranges of  $\delta(u_1)$  or  $\rho(\sigma_1)$ , and hence only these three patterns are used during the similarity comparison phase. Our experiments show that 55–79% of patterns can be pruned during the early pruning phase. The actual pruning power depends on the values of  $\epsilon_1$  and  $\epsilon_2$ . If we have the same exact motions in the stream as some patterns, only the patterns whose  $\delta(v_1)$  and  $\rho(\lambda_1)$  are respectively closest to the  $\delta(u_1)$  and  $\rho(\sigma_1)$  of the stream need to be used for the similarity computation phase.

## 5.4 Computational Efficiency

We tested the CPU time spent on recognizing signs in the stream using the patterns of long durations. We used a Microsoft WindowsXP Professional machine with Intel Pentium processor running at 2.5 GHz for these tests. In our MATLAB implementation, each update and SVD computation of matrix  $M_1$  takes about 0.8 msec. It takes our algorithm only about 4.4 msec to compute all the similarities after each update of  $M_1$ . In comparison, it takes the Ridge-Climbing algorithm using weighted-sum SVD [14] as explained in Section 6 about 115.6 msec to compute the similarities between the stream and the 100 pattern signs. When there are a large number of patterns, more than 20 fold computational time can be saved by our algorithm.

## 6. RELATED WORK AND COMPARISON WITH THIS WORK

For individual similarity matching, Euclidean or weighted Euclidean distance, dynamic time-warping and longest common subsequence distance [5, 3, 13, 10, 2, 16] have been used, and much of the work concentrates on reducing the dimensionalities of time series or multi-attribute data for indexing purpose.



**Figure 9: Component variances  $\delta(u_1)$  of the 1<sup>st</sup> singular vectors (a) and the normalized 1<sup>st</sup> singular values  $\rho(\sigma_1)$  (b) of stream HANDSPRING\_RUN compared with those of the 7 patterns.**

Discrete Fourier Transform (DFT) [5] has been used to reduce the dimensionality of time series data. Dimensionality can be reduced by retaining only the first few coefficients. Dimensionality reduction has also seen application of Discrete Haar Wavelet Transform (DWT) in [3, 13]. Haar wavelet functions are used to represent time series data in terms of the sum and difference of a mother wavelet and part of wavelet coefficients can be truncated to reduce the dimensionality. One drawback of DWT is that the data sequence length is required to be in integral power of two, which is obviously too strong for practical scenarios. Piecewise Aggregate Approximation (APP) and Adaptive Piecewise Constant Approximation (APCA) proposed in [10, 2] are also applicable only for dimensionality reduction of time series data. Singular Value Decomposition (SVD) has been used to reduce the dimensionality of time series data [10, 2], and it has been shown in [10] that SVD can provide better pruning power than other techniques and scale up well to database size. The drawback is its high computation time in the case of individual matching. [1] addresses the approximate similarity search of time series data by clustering with SVD.

Efforts of indexing multi-attribute data can be found in [16]. Dynamic time warping (DTW) and the Longest Common Subsequence (LCS) are used for similarity measures in [16]. Both DTW and LCS have a computational complexity of  $O(wd(m+n))$ , where  $w$  is a matching window size,  $d$  is the number of dimensions, and  $m, n$  are the lengths of two data sequences. If  $m$  or  $n$  are quite different,  $w$  has to be a significant portion of  $m$  or  $n$ , and the computation can be even quadratic in the length of the sequences.

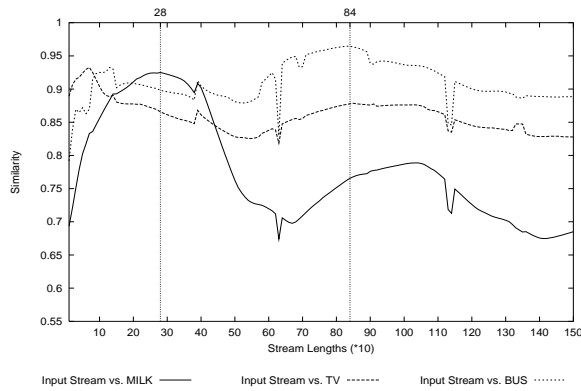
Separation and recognition of continuous stream data are considered in [6, 14]. Prediction method by fast Fourier transform (FFT) in [6] is used to restrict the search space. Here, only time series data is considered, and fixed sampling time intervals are assumed so that Euclidean distance or weighted Euclidean distance can be used to measure the similarity of two equal-length series. In [14], sequential scan is used for each similarity computation, and a weighted-sum SVD is defined as below for measuring the similarity of two data sequences of different lengths:

$$\Psi(Q, P_k) = \min(\Psi_1(M_1, M_2), \Psi_2(M_1, M_2))$$

where

$$\Psi_1(M_1, M_2) = \frac{(\sum_{i=1}^n \sigma_i(u_i \cdot v_i)) / (\sum_{i=1}^n |\sigma_i|)}{(\sum_{i=1}^n \lambda_i(u_i \cdot v_i)) / (\sum_{i=1}^n |\lambda_i|)}$$

The above similarity definition includes some noise com-



**Figure 10: Failure of recognizing the motion MILK in the stream after motion TV has been recognized by using the weighted-sum SVD.**

ponents since Figure 5 indicates that not all the inner products of singular vectors should be considered in the definition of the distance measure. For the pattern set with shorter durations and less motion accuracy, Figure 10 shows that the second sign MILK in the stream cannot be recognized by using the similarity as defined in [14].

Segmentation and recognition of continuous stream data have also been addressed in the speech [9], handwriting [11] and gesture recognitions [15] communities. Hidden Markov Models (HMMs) are used, and sufficient grammar information is required for HMM states. Speech and handwriting address uni-attribute data like the time series data. Multi-attribute ASL motions are considered in [15], and five-word sentences are segmented at the word level and recognized by using HMMs with 92-98% word accuracy. The number of words in a sentence is required to be known beforehand, so are the grammar constraints or forms of sentences. If the number of words in a sentence is assumed to be known, and the forms or grammar constraints are known for the experimental cases, our proposed approach can be expected to outperform HMM-based methods. This comparison will be addressed in the future.

## 7. CONCLUSIONS AND DISCUSSIONS

In this paper, we have considered streaming sequences of multi-attribute motions. Our goal was to (i) recognize patterns and (ii) segment the continuous motion streams without knowing the number of motions in a stream. We examined the properties of singular vectors and singular values of data of similar motions. It was observed that if two motions are similar to each other, then the acute angle between their first singular vectors approaches zero, and the same observation holds for their singular value vectors. Based on this observation, we defined a concise similarity measure based on the SVD properties. Computation of the proposed similarity is independent of the lengths of the pattern and stream data.

For handling the recognition and segmentation of multi-attribute motions generated in real-time, we also outlined a five-phase algorithm based on the proposed similarity measure using SVD components. This five-phase algorithm uses component variances of the first singular vectors and the normalized first singular values to prune non-promising pat-

terns for similarity computations. Experiments were carried out using motion capture data, virtual model motion data, and ASL sign motion data. These experiments show that the five-phase algorithm can segment and recognize different motions in multi-attribute data streams of both one DOF and three DOFs with 100% accuracy even in the presence of brief transitional noisy data.

We are aware of the need for evaluating the proposed approach with any of the well-known performance evaluation metrics used for the segmentation-recognition task (e.g. equivalents to the word error rate used in automatic speech recognition), and are aware that the lack of such a standard evaluation represents a limitation of the paper to be addressed in the future. A systematic comparison with other approaches is needed to be performed in the future.

## 8. REFERENCES

- [1] V. Castelli, A. Thomasian, and C.-S. Li. CSVD: Clustering and singular value decomposition for approximate similarity search in high-dimensional spaces. *IEEE Transactions on knowledge and data engineering*, 15(3):671–685, 2003.
- [2] K. Chakrabarti, E. J. Keogh, S. Mehrotra, and M. J. Pazzani. Locally adaptive dimensionality reduction for indexing large time series databases. *ACT Transactions on Database Systems*, 27(2):188–228, 2002.
- [3] K.-P. Chan, A. W.-C. Fu, and C. Yu. Haar wavelets for efficient similarity search of time-series: With and without time warping. *IEEE Transactions on knowledge and data engineering*, 15(3):686–705, 2003.
- [4] Cyberglove datasheet [Online]. Available: [www.immersion.com/3d/docs/cyberglove\\_datasheet.pdf](http://www.immersion.com/3d/docs/cyberglove_datasheet.pdf)
- [5] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos. Fast subsequence matching in time-series databases. In *SIGMOD*, pages 419–429, May 1994.
- [6] L. Gao and X. S. Wang. Continually evaluating similarity-based pattern queries on a streaming time series. In *ACM SIGMOD*, pages 370–381, Jun 2002.
- [7] G. H. Golub and C. F. V. Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, Maryland, 1996.
- [8] Specification for a standard humanoid, 1999. H-Anim Working Group of the Web3D Consortium, Inc.
- [9] X. D. Huang, Y. Ariki, and M. A. Jack. *Hidden Markov Models for Speech Recognition*. Edinburgh University Press, 1990.
- [10] E. J. Keogh, K. Chakrabarti, M. J. Pazzani, and S. Mehrotra. Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and Information Systems*, 3(3):263–286, 2001.
- [11] A. Kundu, Y. He, and P. Bahl. Recognition of handwritten words: First and second order hidden markov model based approach. *Pattern Recognition*, 22(3):283–297, 1989.
- [12] Recipe for motion capture. MoCap Lab, Ohio State University.
- [13] I. Popivanov and R. J. Miller. Similarity search over time series data using wavelets. In *Proceedings of The 18th International Conference on Data Engineering*, pages 212–221, Feb 2002.
- [14] C. Shahabi and D. Yan. Real-time pattern isolation and recognition over immersive sensor data streams. In *Proceedings of The 9th International Conference on Multi-Media Modeling*, pages 93–113, Jan 2003.
- [15] T. Starner, J. Weaver, and A. Pentland. Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1371–1375, 1998.
- [16] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, and E. Keogh. Indexing multi-dimensional time-series with support for multiple distance measures. In *SIGMOD*, pages 216–225, Aug 2003.
- [17] VRML97 functional specification and VRML97 external authoring interface (EAI) international standard, 1997, 2002. The Web3D Consortium, Inc.
- [18] M. E. Wall, A. Rechtsteiner, and L. M. Rocha. Singular value decomposition and principal component analysis. In D. Berrar, W. Dubitzky, and M. Granzow, editors, *A Practical Approach to Microarray Data Analysis*, pages 91–109. Kluwer, Norwell, MA, 2003.