https://doi.org/10.20965/jaciii.2012.p0013

Paper:

Similarity Retrieval of Motion Capture Data Based on Derivative Features

Worawat Choensawat*, Woong Choi, and Kozaburo Hachimura***

*School of Science and Engineering, Ritsumeikan University
1-1-1 Noji Higashi, Kusatsu, Shiga 525-8577, Japan
**Department of Information and Computer Engineering, Gunma National College of Technology
580 Tobamachi, Maebashi, Gunma 371-8530, Japan
E-mail: {gr0011es@ed., hachimura@media.}ritsumei.ac.jp, wchoi@ice.gunma-ct.ac.jp

[Received July 23, 2011; accepted November 25, 2011]

In this paper, we propose (1) a method of similarity retrieval of motion capture data in which a new feature extraction technique is introduced for the improvement of similarity search precision, as well as (2) a method to reduce the search time on a large database by using lower bound Dynamic Time Wrapping (DTW). For similarity search, joint speed has been mainly used as features of a particular motion. Our method differs from others in that we use not only the magnitude of speed but also the pattern of speed change. We measure the pattern of changing joint speed in a short period of time with the derivative of joint speed. In our experiments, we found that our proposed feature extraction can improve search precision and time. The average precision was greater than 90% and its computation time was 10 seconds on a dataset of 225 motion clips with a total of 81,851 frames from CMU's database. The experiments showed that we can improve search precision using our proposed feature extraction technique compared to the retrieval method without using this method. For search time, our experiment shows that our retrieval method using the lower bound DTW can efficiently reduce the amount of search data.

Keywords: motion capture, content based retrieval, dynamic time wrapping, derivative feature

1. Introduction

Nowadays, Motion Capture (MoCap) data has been widely used for producing animations in a variety of applications such as animated movies, video games, simulations, and in the research field of human body analysis. A large number of movies use a MoCap system for CG effects to replace traditional hand-drawn animation. With advanced computer graphics, most video games often use MoCap data to animate in-game characters such as collectors, crafters, historians, and other human-like characters in Guild Wars. The production of 3D animated content requires very precise data because MoCap provides accurate and semantically rich data.

Similarity retrieval of MoCap data has received substantial and increasing attention for several years. One of the large MoCap databases provided by Carnegie Mellon University (CMU) [1] has a collection of 2,605 motion clips in 6 categories and 23 sub-categories. This database has been provided on the Internet and is free for use in research projects. On the website, a user needs to know keywords for searching motion clips annotated in advance. Because of the lack of a searching tool with an example motion clip, most users are weary to search motions similar to their queries. For applications, such as film production, creators make an effort to select appropriate motions to generate high quality animation. Therefore, an efficient similarity retrieval system is very important for the effective reuse of MoCap data because the more similar motions matching a motion query, the more realistically animated characters can be produced.

Searching a large database for similar motions is not an easy task due to a tradeoff between search time and similarity retrieval accuracy. Similar motions may not be numerically similar due to the following factors: (1) clips in a database vary in length, (2) personal styles, and (3) movement styles.

Firstly, a MoCap database commonly consists of motion clips of highly varying length in motion categories such as walking, running, and jumping. Secondly, people usually develop their own movement habit as their personal style. Thirdly, movement styles are types of movement such as slow walk, normal walk, and fast walk.

Dynamic Time Warping (DTW) is a popular method for dissimilarity measure between two sequences which may vary in length such as MoCap data [2, 3]. Although, DTW can measure the similarity between two time series which may vary in time and space, the main drawback of the DTW method is its quadratic computational complexity, especially in long query motions. To alleviate its time complexity, an entire MoCap sequence will be segmented into small motion clips. However, the accuracy of similarity retrieval will be affected by the accuracy of the segmentation techniques used. Alternatively, the lower bound DTW proposed in [4] successfully reduces the number of candidates by filtering out unrelated motions.

This paper proposes a feature extraction algorithm for searching a MoCap database for all similar motions to a query motion based on joint speed and the pattern of the change of joint speed over a short interval. The proposed algorithm aims for high accuracy retrieval of similar MoCap data. We compare the proposed features with geometric features which were introduced by Müller et al. [5]. Our experimental results demonstrate that our features are more effective for retrieving motions within the same motion category of a query motion.

In Section 2, we describe some related approaches of feature selection and extraction of MoCap data as well as indexing and retrieval of time series data. We also discuss the advantages and disadvantages of these methods before introducing a novel content-based retrieval method. In Section 3, we describe in detail our proposed similarity retrieval method using a framework consisting of feature extraction, candidate selection and search result ranking.

To facilitate retrieval from a large motion database, we implement a graphical user interface for allowing users to set up query parameters as well as a motion viewer for displaying resulting motions, described in Section 4. In Section 5, for evaluating our proposed content-based retrieval method, we compare our method by using average precision over the list of search results. Our conclusion and future work are presented in Section 6.

2. Related Work

There are a number of different approaches for the similarity retrieval of MoCap data as found in [5–11]. The motion map approach clusters and shows key frames in a 2D grid map [11]. Sakamoto et al. introduced the method of motion map by using Self-Organizing Maps (SOMs) for the indexing and retrieval of motion data. The SOMs was trained with the motion data. After training, the SOMs is used as the motion indexing map of the motions on which the similarity metric and retrieval are based. This method can perform only retrieval on the query examples that have exact node-to-node matching in the indexed motions.

Kovar and Gleicher [8] aimed at reusing and processing motion capture material for synthesis of new, realistic motions from example motions. To accomplish the goal, they solve the fundamental problem of identifying and extracting suitable motion clips from the database on hand. For a short query motion, the task is to retrieve all clips in the database containing parts or aspects similar to the query. Base on their work, the criterion point of similarity is that corresponding frames should have similar skeleton poses. As a result, two motions may be regarded as similar if they represent variations of the same action or sequence of actions. This concept was also used in the development of "Geometric Feature" by Müller et al. [5].

2.1. Features of Motion Sequences

A human motion can be represented as a sequence of joint angles, e.g., whether quaternion-based or Euler angle-based representations. Due to the high dimensionality of MoCap data, PCA-based compression of pose-base features set was chosen for alleviating this problem as discussed in the next sub-section.

Recent works [12–14] have exploited not only the properties of a pose but also the kinematic properties of a motion. This allows a wide range of properties to serve as the basis for the parameterization. Some of their works also included the position of the end effectors (the head, the legs, and the arms), the average, minimum, or maximum angular speed of a joint, or features of aggregate quantities such as the center of mass. Lee et al. [13] and Kovar et al. [12] used root positions and orientations for selecting appropriate transition points among poses or intervals of motions. Wang and Bodenheimer [14] proposed the use of the linear combination between joint angles and velocities as a feature set for synthesis as well as evaluation of motion transitions. However, they obtained coefficients for the feature set through a cross-validation of user studies and these coefficients might depend on the particular motion database they used.

For geometric features, Müller et al. [5, 6] presented a method for the feature extraction and matching of motion data by using geometric relations between the body points of a pose. As an example of this kind of feature, consider the pose whether the right foot lies in front of or behind the plane spanned by the left foot, the left hip joint and the center of the hip (the root). Although this method showed good results for some types of motion data, it did not consider quantitative properties such as velocity. Moreover, this method requires users to select different features according to the characteristics of the motion types. Such features are right/left foot in front, right/left foot raised, legs crossed and so on. These features must be pre-defined for distinguishing between movements.

Krüger et al. [9] showed a comparison among the following feature sets: (1) joint angles encoded in either quaternion-based or Euler angle based representation; (2) a combination of joint angles and joint velocity; and (3) features based on kinematic properties as addressed in [12, 13]. The feature sets of Krüger et al. consist of varying parameters such as (1) the number of considered joints and (2) including/excluding temporal information. The temporal information can be encoded with a window of frames, e.g., the authors extended their feature set of three frames and five frames. Their findings are as follows. First, the lower number of joints allows the fastest searches. Second, there are no significant differences on searching results with an increasing number of joints. Lastly, some applications need to include temporal information. In their experiments, they pay attention to the close numerical similarity of motions while our paper focused on motions that share the same action category while having different variations. For example, walking motions may have several variants such as turn-right walk and turn-left walk.

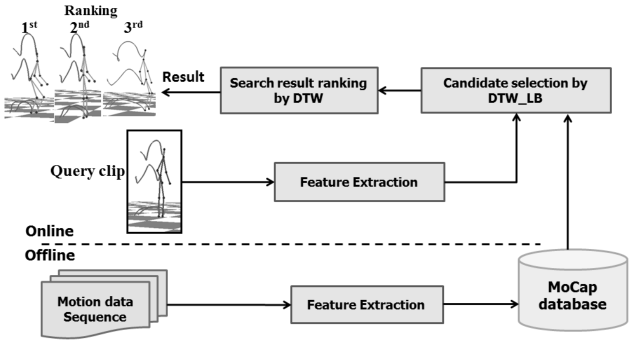


Fig. 1. Overall framework.

2.2. Dimensionality Reduction Approach

Due to high dimensional human motion sequences, principal component analysis and singular value decomposition, widely used dimensionality reduction and feature extraction methods, are also used in the preprocessing of MoCap data. Some search algorithms were based on a weighted PCA-based pose representation that allows flexible and efficient pose-to-pose distance calculations. For example, Forbes et al. [7] used a weighted PCA in the searching of motion data, and Barbiĉ et al. [15] used PCA techniques to cut a long motion data stream into segments of a single behavior. PCA, therefore, has presented a method for quickly searching long, unsegmented motion clips for a sub-instance that most closely matches a short query clip.

Onuma et al. proposed FMDistance [10] which is a distance function for similarity retrieval with a low-dimensionality feature set. The feature set consists of the average kinetic energy of each joint angle in logarithms. The total approximate kinetic energy is expanded by each of the approximately 70 angles in the data. Retrieval with a low number of features yields a fast search time but it is not able to search a subsequence in a motion.

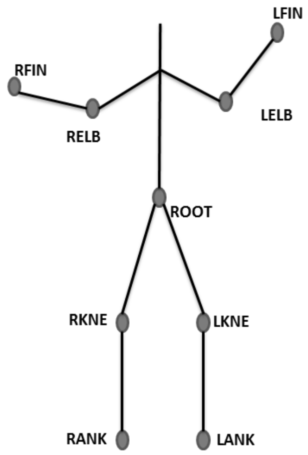


Fig. 2. Nine joints for feature extraction consisting of RANK: right ankle, LANK: left ankle, RKNE: right knee, LKNE: left knee, RELB: right elbow, LELB: left elbow, RFIN: right hand, LFIN: left hand, and ROOT.

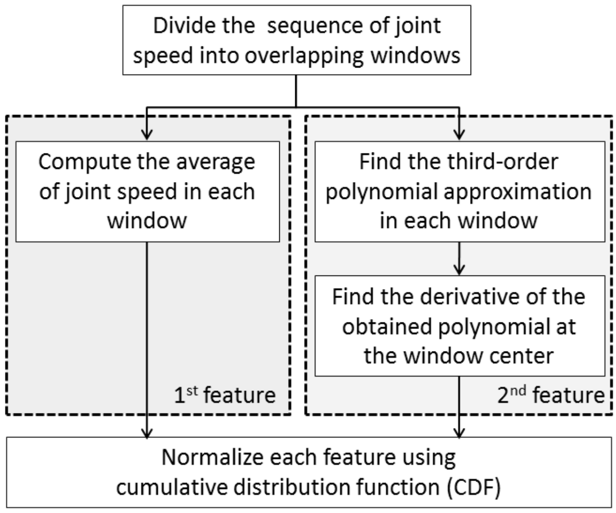


Fig. 3. A flowchart of feature extraction.

3. Similarity Retrieval of MoCap Data

In this section, we describe our method for content-based retrieval of MoCap data. Its framework is shown in **Fig. 1**. There are three main procedures, namely, feature extraction, candidate selection, and search result ranking. Basically, a database of the features extracted from a collection of motion data is done offline. We also update the database with new motion data in an offline process.

3.1. Feature Extraction

A motion is a frame sequence, with each frame defining a posture. The most commonly used formats of motion capture data are C3D, TRC, ASF/AMC and BVH. The ASF/AMC and BVH formats store hierarchical skeleton data, while the C3D and TRC format store 3D coordinates. For each format, the number of recorded joints can be varied to meet specific objectives and applications, and sometimes depends on the type of motion capture system.

For capturing full human body motion, 24–40 markers are commonly used. In our method, we selected nine joints of the TRC format at the frame rate of 120 Hz. The human model of nine joints is shown in **Fig. 2**. As found in Johansson's research [16], these joints of human limbs and one root are sufficient for identification of human movement.

Figure 3 shows the processes of our feature extraction approach. We divide a whole motion capture sequence into a number of overlapping windows. The motion sequence in a window is called a sub-motion. Let w be the number of frames in a sub-motion. The overlapped area between two adjacent sub-motions equals to $w/2$ as shown in **Fig. 4**. From each sub-motion, we extract two features: one is the average of joint speed and the other is a pattern of change of joint speed named as a derivative feature.

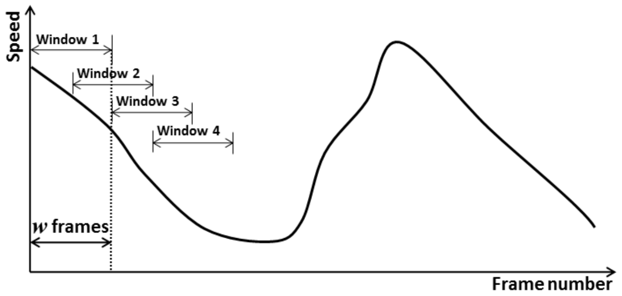


Fig. 4. Sub-motions with size of w and its overlapped area equal to $w/2$.

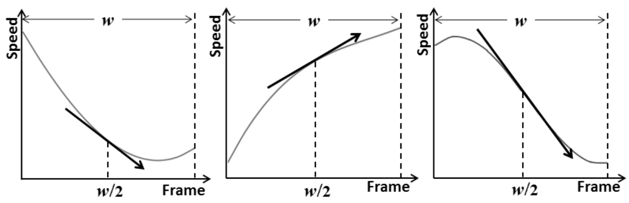


Fig. 5. Derivative feature representing a pattern of change of joint speed.

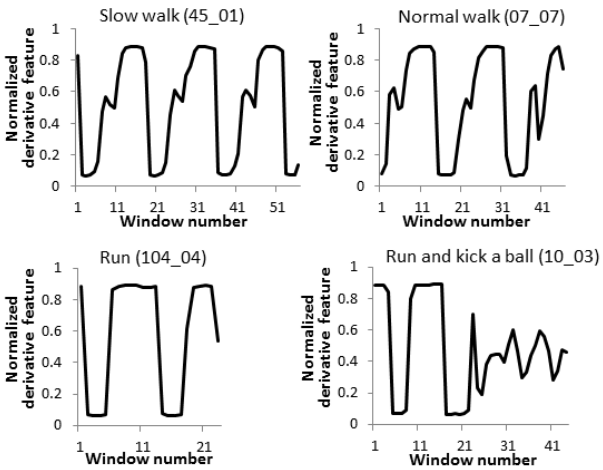


Fig. 6. Derivative feature showing the pattern of joint speed of the right ankle for various kinds of motions.

3.1.1. Average Joint Speed

We use the average speed, μ_s^j, at j-th joint of s-th sub-motion as shown below:

$$\mu_s^j = \frac{1}{w}\sum_{i=1}^{w} v_{s,i}^j \quad \ldots \ldots \ldots \ldots \quad (1)$$

where w is the window size and $v_{s,i}^j$ is the magnitude of velocity at frame i in s-th sub-motion of j-th joint. We use the average speed because the magnitude of speed is invariant to movement directions and the average value of joint speed from a window can reduce data size and noise. For example, a motion of running straight in one direction is regarded as being similar to a motion running in another direction.

3.1.2. Derivative Feature

In the previous sub-section, we explained how to extract a feature representing the magnitude of velocity. However, we focused not only on the magnitude but also on the pattern of change of joint speed. In this sub-section, we describe how to extract a derivative feature from a small overlapping window. Derivative features reflect a pattern of change of joint speed and it is invariant to the magnitude of joint speed.

First, we apply a third-order polynomial fitting method to approximate joint speed in a window as follows:

$$F(t) = c_3t^3 + c_2t^2 + c_1t + c_0. \quad \ldots \ldots \quad (2)$$

We then calculate its derivative at the center of the window. The derivative feature is defined below:

$$\alpha_s^j = 3c_3t^2 + 2c_2t + c_1 \quad \ldots \ldots \ldots \ldots \quad (3)$$

where $t = w/2$.

The derivative α_s^j represents the slope of the joint speed at $w/2$. With the use of third-order polynomial, the estimation value is sufficient for discovering the shape of the considering curve in a window because the objective of such a shape is to improve the search accuracy when combining with the joint speed. **Fig. 5** illustrates how a derivative feature reflects a pattern of change of joint speed where the leftmost pattern is more similar with the rightmost than the center. The arrow lines represent the derivative feature, which is actually a slope at the mid-point of the curve in a window.

We explain the derivative feature by giving an example; **Fig. 6** illustrates how walking, running and kicking a ball are all different actions. The patterns in **Fig. 6** are extracted from the motion clips from CMU where their file-name identifies, for example, their movement Slow walk, actor ID 45, and motion ID 01. We can easily distinguish kicking a ball from walking and running. However, it is difficult to distinguish between walking and running motions by using only their derivative feature. This can be solved by using both the average magnitude of joint speed and the derivative feature. **Fig. 7** shows the two dimensional time series of both the average of joint speed and the derivative feature.

In summary, for a motion sequence M composed of N sub-motions and J joints, we can define a feature set of a motion $\mathscr{F}(M)$ as follows:

$$P_s = \{(\mu_s^j, \alpha_s^j) : j \in [1:J], s \in [1:N]\} \quad . \quad (4)$$

$$\mathscr{F}(M) = \{P_s; s \in [1:N]\} \quad \ldots \ldots \ldots \quad (5)$$

where P_s is the feature of the s-th sub-motion of which μ_s^j and α_s^j are the average speed and the corresponding derivative of j-th joint, respectively.

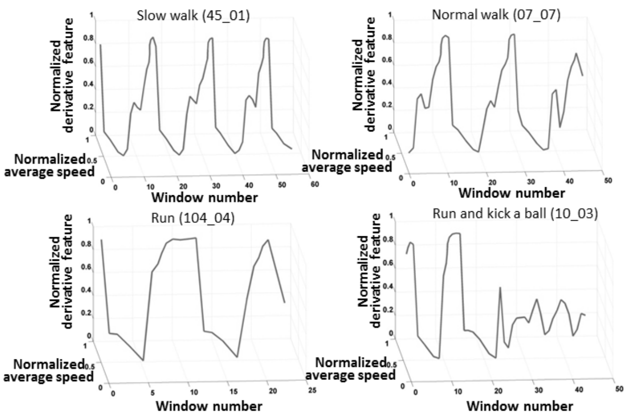


Fig. 7. Two dimensional time series of the average joint speed and the derivative feature of the right ankle for various kinds of motions.

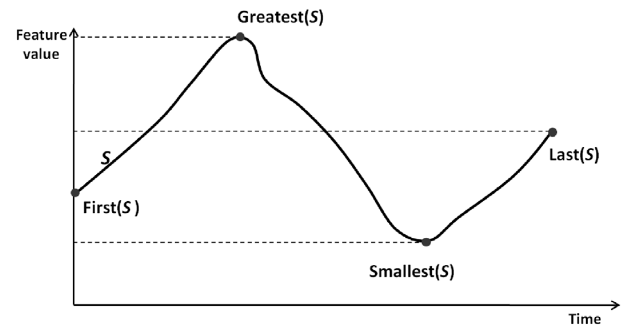


Fig. 8. Lower bound DTW.

3.1.3. Feature Normalization

In the previous sub-sections, we described how to extract features from a motion. A set of motion features consists of the average of joint speed and the derivative of that speed. The features of motions do not distribute uniformly in the database and have different scale ranges.

The goal of feature normalization is to represent all motion features in the same range of scale and in normal distribution. For this purposed, we used the Cumulative Distribution Function (CDF) of the Gaussian distribution feature normalization. The CDF of each feature at each joint will be in the range of 0 to 1. For the computation of CDF, the mean and standard deviation of each feature at each joint are obtained from all motion data in the database. After that, we will compute the CDF of all features by using the algorithm in [17].

3.2. Candidate Selection

We employ the DTW algorithm to measure the dissimilarity distance between a query motion and the motions in a database. DTW is a robust distance measure for time series because it can compensate for motion with varying lengths and preserves ordering.

Due to the computation time complexity of DTW, we need to discard non-relevant motion clips as early as possible. We adopt the lower bound DTW proposed in [4], referred to as DTW_LB, to filter out non-relevant motion clips since this technique satisfies the triangular inequality and guarantees no false dismissals. DTW_LB extracts a 4-tuple feature vector from each feature as indexing attributes. The distance function D_{tw_lb} is a lower bound distance function that consistently returns a distance smaller than or equal to D_{tw}, the time warping distance.

$$D_{tw} \geq D_{tw\_lb} \quad \ldots\ldots\ldots\ldots\ldots \quad (6)$$

From a time series S in **Fig. 8**, we get $Feature(S)$ which is a 4-tuple and defined as follows: $Feature(S) = < First(m), Last(M), Greatest(M), Smallest(M) >$ where $First(S)$ and $Last(S)$ are the values at the first and last positions in sequence S, respectively. $Greatest(M)$ and $Smallest(M)$ are defined as the values at the highest and the lowest positions in sequence S, respectively.

Given two sequences S and Q, the lower-bound distance function $D_{tw\_lb}(S,Q)$ is defined by a uniform norm, L_∞ as follows:

$$\begin{aligned} D_{tw\_lb}(S,Q) &= L_\infty(Feature(S), Feature(Q)) \\ &= \max \begin{cases} |First(S) - First(Q)| \\ |Last(S) - Last(Q)| \\ |Greatest(S) - Greatest(Q)| \\ |Smallest(S) - Smallest(Q)| \end{cases} \end{aligned} \quad (7)$$

For any two sequences S and Q and tolerance ε, the following equation always holds

$$D_{tw} \leq \varepsilon \Rightarrow D_{tw\_lb} \leq \varepsilon. \quad \ldots\ldots\ldots\ldots \quad (8)$$

From Eq. (8), by defining a tolerance ε, we can filter out non-relevant motions whose D_{tw_lb} exceeds the tolerance value ε.

3.3. Search Result Ranking

After applying DTW_LB to a candidate selection, we finally measure the dissimilarity between a query motion and motion data in the candidate set for sorting them. This procedure is called search result ranking. The motion dissimilarity measure $D_{tw\_opt}(S,Q)$ is defined as the average of the local distance metric in the optimal path and is obtained by DTW [18].

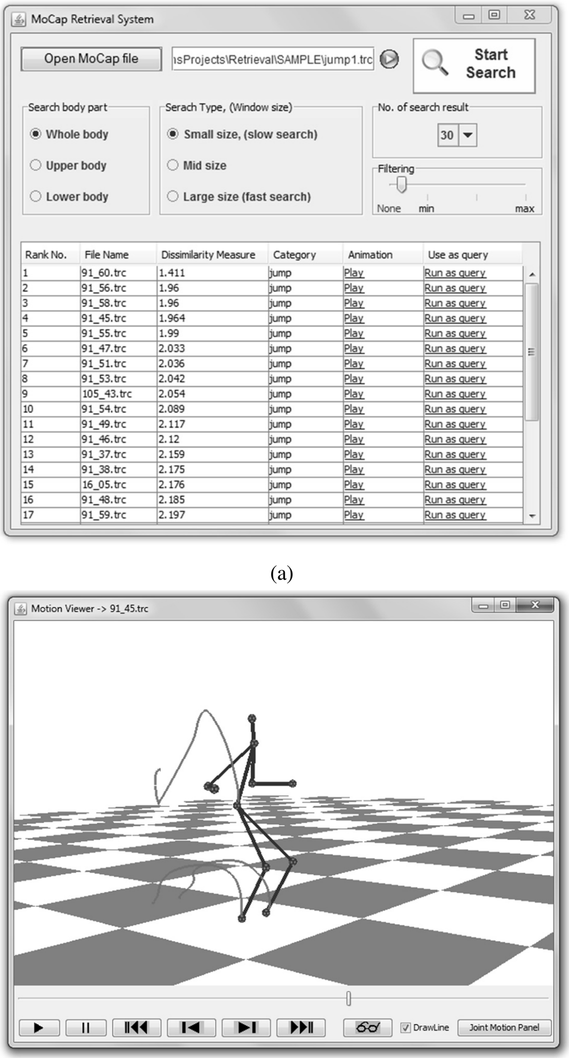
$$D_{tw\_opt}(S,Q) = \frac{1}{L} D_{tw}(S,Q) \quad \ldots\ldots\ldots \quad (9)$$

where L is the length of an optimal path of DTW. An advantage of D_{tw_opt} is for the case of any two similar motions of much different length [8]. We use the average value rather than the total in order to measure distance regardless of path length.

We define the dissimilarity of two sub-motions, P_u and P_v, by summing the Manhattan distance of all joints:

$$D(P_u, P_v) = \sum_j^J (d(\mu_u^j, \mu_v^j) + d(\alpha_u^j, \alpha_v^j)) \quad \ldots \quad (10)$$

where J is the number of considered joints and $\{(\mu_u^j, \alpha_u^j) : j \in [1:J]\}$ are the features of P_u. The search result ranking



(a)

(b)

Fig. 9. GUI of our MoCap retrieval system (a) main window, (b) motion viewer: motion display window.

is then a sort of all motions in ascending order of dissimilarity from the query motion.

4. Implementation of Retrieval System

We implemented a retrieval system with a GUI that provides functions to help users in adjusting search parameters optimizing between search time and search accuracy, as well as in displaying motion clips. **Fig. 9** is a screenshot of the system. In the main window of the system shown in **Fig. 9(a)**, users begin with selecting a motion query by using a file browser dialog. When users click the play button, the selected motion query will be played back in a motion viewer as shown in **Fig. 9(b)**. Users can display the movement trajectories of any joints from the first till last poses; for example, the movement of both legs is shown by the two lines. This function helps users to confirm whether they selected the correct file.

Before starting the search, users can set up query parameters concerning the body parts, the search type, and the filtering value.

i. For a body part search, users can emphasize either a lower/upper body part or the whole body.
ii. For the search type, this function adjusts the precision of search results and corresponds to the window size defined in Section 3.1.
iii. For the filtering value, this function applies the DTW_LB to select a set of candidates by setting a threshold as described in Section 3.2.

After completing a query search, the system shows the resulting motions in ascending order of the dissimilarity distance to the query motion. **Fig. 9(a)** shows the attributes of resulting motions consisting of rank number, file name, dissimilarity measure, and category as defined by its movement type (such as walking, jumping, running, etc.). Users can select the number of resulting data and select resulting motion clips for displaying data on the motion viewer.

We implemented the system on the following hardware and software platform: The hardware platform for the experiments is an Intel Core i7 2.8 GHz laptop equipped with 8 GB of DDR3 RAM. The application was implemented in JAVA (JVM by using the option "-Xmx 1024m," which sets the application to use a maximum of up to 1024 MB of memory), and we used MySql as the DBMS.

5. Evaluation

We evaluated the following aspects of our method:

i. Accuracy of the search tool and its computational time.
ii. Effectiveness of the derivative feature.
iii. Effectiveness of candidate selection by using DTW_LB.
iv. User satisfaction.

We used 225 motions selected from the well-known CMU database [1] as a data set. Since segmentation is not in the scope of this paper, we selected MoCap clips that contain only one kind of movement. The data set consists of five categories which are common motions in our daily life, such are walking, running, kicking a ball, swinging a golf club, and jumping, as shown in **Table 1**.

5.1. System Evaluation

We used the leave-one-out validation method for the evaluation where one motion is extracted from the database and used as query data. The effectiveness of the proposed method was measured using precision evaluation. In general precision is defined as:

$$precision = \frac{No.\ of\ relevant\ data\ retrieved}{No.\ of\ data\ motions}.$$

Table 1. Data set of motion clips from the CMU database.

	All	Walk	Run	Jump	Kick	Golf
No. of clips	225	113	52	44	6	10
No. of actors	25	19	5	5	2	1
Avg. frames	362	425	186	400	333	446
Min. frames	127	243	127	210	289	363
Max. frames	923	894	761	923	382	561

We use $P(N_R)$, a measure for similarity retrieval [19], which is a measure of the ratio of relevant data in the top N_R retrieval results.

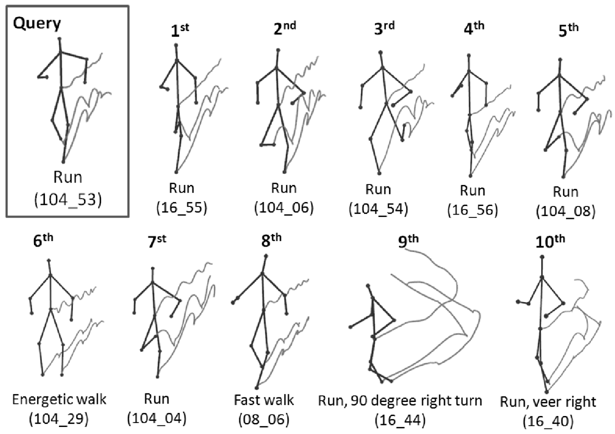
$$P(N_R) = \frac{No.\ of\ relevant\ data\ in\ the\ top\ N_R\ rank}{N_R}$$

For our evaluation, we used the average $P(N_R)$, $\overline{P(N_R)}$, as a measure of the leave-one-out method choosing a query data one by one from the database.

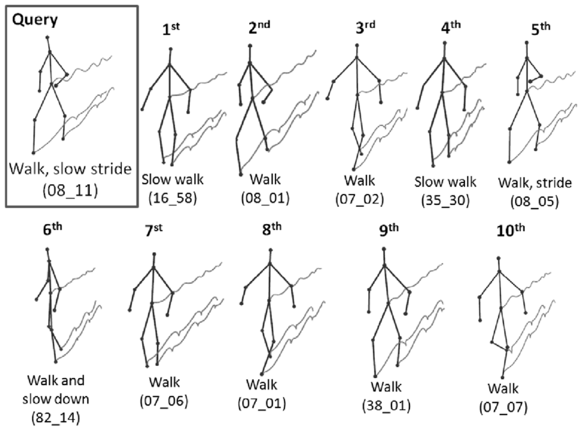
Figure 10 shows the top 10 ranked results for a 'running' query (a), a 'slow stride walking' query (b), and a 'jump' query (c). **Fig. 11** shows the comparison of average $P(N_R)$ tested against window sizes of 8, 16, 32, and 48 frames per clip, where window size of 16 yields the highest accuracy of 91.38%. By using window size of 16 frames, the proposed feature extraction can improve the overall search accuracy of $P(N_R)$ by 5.58% ($p = 0.001$ by paired t-test). **Fig. 12** shows the mean and standard error of the retrieval accuracy of five categories.

For the effectiveness of our feature selection, we compared our proposed features with the geometric features [5] as previously mentioned in Section 2. Geometric features are a class of Boolean features expressing geometric relations between specified body parts of a pose. We employ five geometric features which are adequate for walk and run motions. The selected geometric features consist of 1) left foot in front, 2) left knee bent, 3) right knee bent, 4) left elbow bent, and 5) right elbow bent. For 'left foot in front,' we identify whether the left foot lies in front or behind the plane spanned by the right foot, the right hip and the center hip (root). We also used an angel of 120° to classify the respective body part as bent or stretched. Then, we measured motion dissimilarity by using D_{tw_opt} as described in Section 3.3. **Fig. 13** shows the comparison of motion dissimilarity measures of examples between our proposed features and geometric features.

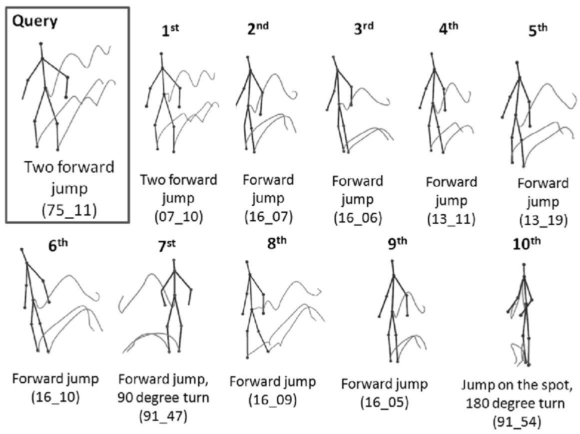
As shown in **Fig. 13**, motion dissimilarity measures in the same category are considerably small in our method regardless of how variant the motion pairs are. In contrast, it is larger between different categories of motions such as walking and running, which shows the greatest dissimilarity measure. On the other hand, using geometric features for queries failed to distinguish the differences derived from different categories of motion. This is because the poses in walking and running may be similar despite their motion speeds being rather different, which demonstrates that it is insufficient to consider only pose difference in motion retrieval.



(a) 'running motion' query (top left)



(b) 'walking motion' query (top left)



(c) 'jumping motion' query (top left)

Fig. 10. The resulting top 10 motions retrieved by our method. The motion paths are represented by three trajectory lines for the ROOT, LANK, and RANK joints.

We also measure a candidate ratio of DTW_LB in order to evaluate the filtering effects of DTW_LB by examining the trend of false alarms. Candidate ratio is defined as follows:

$$candidate\ ratio = \frac{Number\ of\ candidate\ motions}{Number\ of\ total\ motions}$$

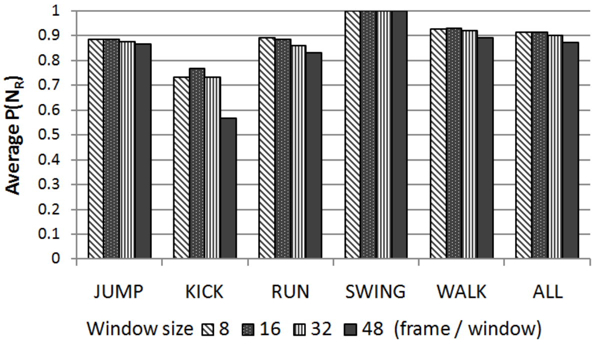


Fig. 11. Comparison of average $P(N_R)$ tested against window sizes of 8, 16, 32, and 48 frames using our proposed method.

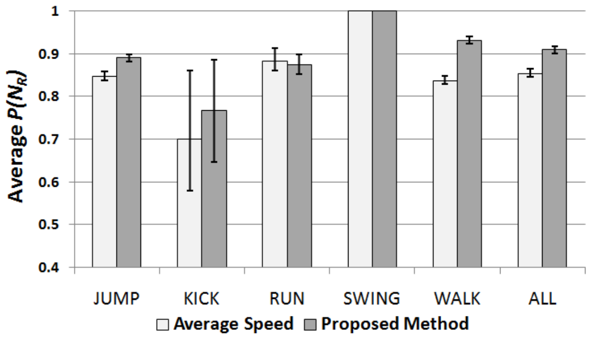


Fig. 12. Comparison of average $P(N_R)$ between using only the average speed feature and our proposed derivative features based on a window size of 16 frames per window.

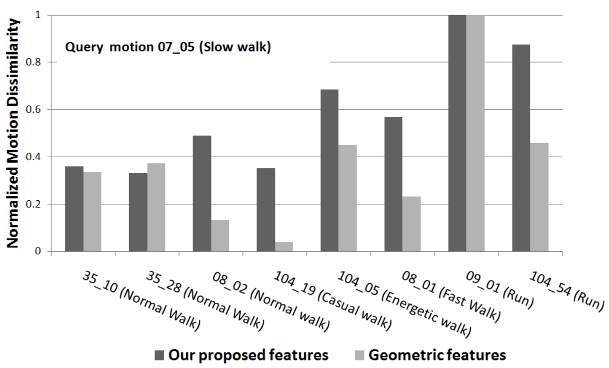


Fig. 13. Comparison of motion dissimilarity measures using our proposed features (window size of 16 frames) and geometric features in [5]. The dissimilarity measures are normalized to the same range of scale between 0–1.

Applying DTW_LB to motion retrieval, we can reduce the number of search data as shown in **Fig. 13**. For all types of movements, the average candidate ratio with the tolerance of one unit increasing from 5 to 11 will range from 0.10 to 0.98. For example, with tolerance set at 7 for a 'running' query we can reduce the number of search data by 57% as shown in **Fig. 14**.

Using this filtering technique, **Fig. 15** shows that the overall accuracy measured by $P(5)$ with a tolerance of 5

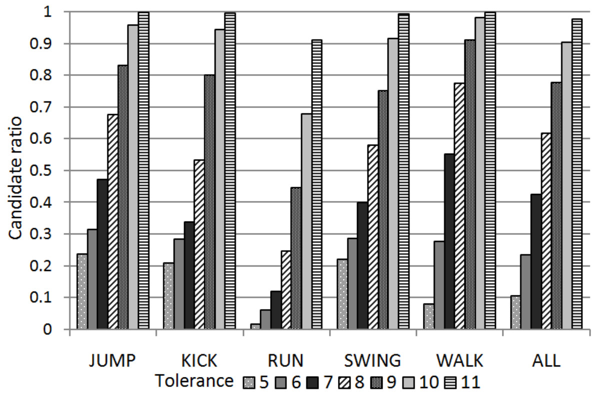


Fig. 14. Candidate ratio using different tolerances between 5 to 11.

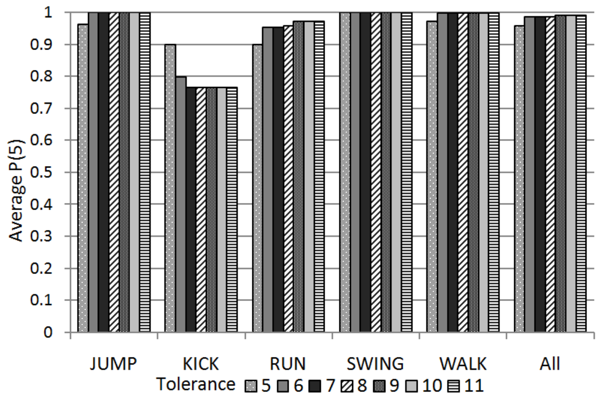


Fig. 15. Comparison of average $P(5)$ tested against the tolerance between 5 to 11 for the same window size, $w = 16$.

was 95.0%, and that with a tolerance ≥ 6 accuracy was 99.5%. However, the candidate ratio of running motions is below 10% with tolerance set at 5 and 6. We found that a tolerance of ≥ 7 yields the candidate ratio greater than 20% for all categories. For the kicking motions, the experiment showed that a tolerance of 5 yields the best result.

We evaluated the computational time comparison between a baseline method and our, the proposed method with/without DTW_LB filtering. By using DTW_LB filtering, we set the tolerance $\varepsilon = 7$. For the baseline method, we extracted only joint speed of every frame without the derivative feature. **Table 2** shows the running time using query motion 06_06, a walking motion with 402 frames in length. The quantity 'execution time' is the total of the following processing times: 1) database query time, 2) feature extraction, 3) candidate selection, and 4) search ranking result. We calculate the average joint speed and the derivative feature of joint positions proposed in Section 3.

5.2. User Evaluation

The purpose of this user evaluation was to investigate user satisfaction of our retrieval system as described in

Table 2. Comparison of the execution time between the baseline method and our system with different window sizes.

	Baseline	Our proposed method without filtering / (with filtering, $\varepsilon = 7$)			
		Window size			
		8	16	32	48
Exe. time (sec.) per query	52	32(21)	19(14)	10(8)	8(6)

Section 4. This evaluation was conducted on usefulness, accuracy and response time of the system. The evaluation was conducted on 20 subjects from the Department of Media Technology, Ritsumeikan University. Fifty percent of the subjects have experience in using MoCap data.

Subjects were asked to complete tasks by finding the top three most similar motion clips to given query clips. Before starting the evaluation, subjects were presented with a video demonstration of the system that they would use to complete their given tasks. After subjects completed their tasks, they were asked to complete a post-task questionnaire. The post-task questionnaire consists of 12 questions assessed with five-point scales, where 1 = strongly disagree and 5 = strongly agree. This post-task questionnaire is the focus of this study. Each subject took approximately 45 minutes to complete the evaluation.

The results of the evaluation are shown in **Table 3** and **Fig. 16**. **Table 3** shows the questions used in our questionnaire, their means and standard deviations, and their t-scores in each subject group. On average, all users agreed with the query results by using the three different window sizes, and the user-rated scores correspond well to that of accuracy shown in **Fig. 11**, where a window size of 16 frames had the both highest accuracy and satisfaction score. There were no significant score differences between the two groups of subjects except for the satisfaction of system response time. Experienced subjects were more satisfied with the system response time than inexperienced subjects. This could be because, from a post-task interview, experienced subjects were more aware of the complexity of content-based MoCap retrieval systems.

Table 3. Means, standard deviations, and T-scores for the post-task questionnaire.

Question	Experienced M (SD)	Inexperience M (SD)	t
1. This software is easy to use.	4.50 (0.53)	4.80 (0.42)	0.10
2. It was easy to post queries to the system.	4.80 (0.42)	4.89 (0.33)	0.30
3. This system help me to complete the given task.	4.40 (0.70)	4.56 (0.73)	0.28
4. It was easy to navigate the search results.	4.60 (0.70)	4.44 (1.01)	0.35
5. The system has a good response time.	4.10 (0.88)	3.20 (0.79)	0.02*
6. The search results satisfied my needs.	4.10 (0.70)	4.00 (0.94)	0.17
7. It was easy to determine if results were relevant to a task.	4.30 (0.67)	4.00 (0.67)	0.17
8. I was satisfied with the accuracy of search results using window size = 16.	4.50 (0.71)	4.30 (0.67)	0.30
9. I was satisfied with the accuracy of search results using window size = 32.	4.20 (0.63)	4.00 (0.67)	0.25
10. I was satisfied with the accuracy of search results using window size = 48.	4.10 (0.74)	3.80 (0.92)	0.21
11. Increasing the window size can improve the search time significantly.	3.90 (0.88)	4.20 (0.92)	0.25
12. Overall the software was satisfying to use.	4.40 (0.70)	4.00 (1.05)	0.19

*represents significant differences between experienced user and inexperience user ($^* p \leq 0.05$)

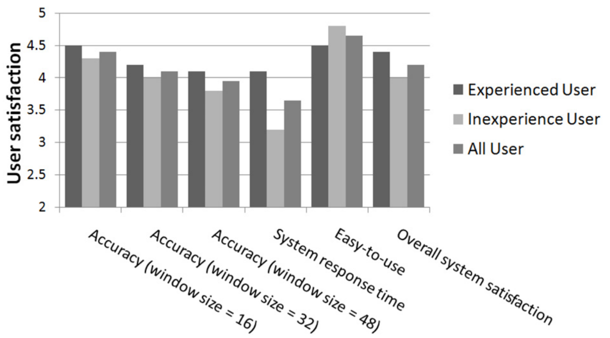


Fig. 16. Retrieval system user satisfaction.

6. Conclusion and Future Work

We have presented a similarity retrieval approach for querying a MoCap database by using the average joint speed and a derivative feature over a small time interval. These features can discriminate between various movements extracted from a MoCap database. We apply an overlapping window technique to emphasize the important movements, discard noises, and also utilize a third-order polynomial curve fitting to illustrate the pattern of the joint speed of a sub-motion.

One merit of the derivative of join speed of a sub-motion is that it is continuous and numeric as time series data, so that existing dissimilarity measures can be applied. We emphasize that our method requires none of the heuristic dissimilarity distance metrics as addressed in [20]. For example, we can apply both a conventional DTW and DTW_LB to measure the distance similarity among derivative features.

The proposed retrieval system considerably increases the query speed by filtering a number of motions dissimilar to a query motion. Our filtering technique is the lower bound DTW, named DTW_LB. This filtering technique can discard non-relevant motion data without false dismissal. The performance of DTW_LB is measured by the candidate ratio. Our experiments showed a fast and effective motion search.

In experiments, we also conducted a feature-selection comparison between our proposed method and other existing methods. First, we have proved that by using both the derivative feature and the average joint speed the mean search accuracy is higher than that of using only joint speed (by about 5%). Secondly, the use of our proposed features outperformed the use of geometric features in terms of considerably different speed between varying walk styles and running.

From a user's point of view, the user satisfaction of our retrieval system shows 1) their strong agreement that the system is easy to use, 2) their agreement that the resulting clips by using our system are similar to the queries, and 3) the agreement of the experienced users that the response time of querying is acceptable. Despite the rather slow response time of the system rated by inexperience users, the overall results of the evaluation are satisfactory. Inexperience users also rated the overall mean accuracy above 4 (out of 5) where searches with the window sizes of 16 and 48 frames are highest and lowest, respectively.

For our future work, our goals include the extension to a combination of motion portions by using Boolean operators for motion queries as used in SQL queries. This can be described in a Boolean expression as in a sequence of frames composing of portions of different motion clips, for example, "kicking a ball followed by jumping." Another example of a combination of two or more query motions is the movement of the lower body part like walking and the upper part like waving a hand. Hence, it will increase our system's usability for iterative retrieval.

Acknowledgements

This work is supported in part by the Globol COE (Center of Excellence) Program "Digital Humanities Center for Japanese Arts and Cultures" at Ritsumeikan University and the Grant-in-Aid for Scientific Research (B) No.2230039, both from the Ministry of Education, Science, Sports and Culture. We would like to thank Prof. Ross Walker for his kind proofreading and advice in the English expressions.

References:

[1] CMU, "MoCap Database," 2003.
[2] A. Bruderlin and L. Williams, "Motion signal processing," In SIGGRAPH' 95: Proc. of the 22nd Annual Conf. on Computer Graphics and Interactive Techniques, pp. 97-104, 1995.
[3] T. Yu, X. Shen, Q. Li, and W. Geng, "Motion retrieval based on movement notation language: Motion Capture and Retrieval," Computer Animation. Virtual Worlds, Vol.16, pp. 273-282, July 2005.
[4] S.-W. Kim, S. Park, and W. W. Chu, "An Index-Based Approach for Similarity Search Supporting Time Warping in Large Sequence Databases," In Proc. of the ICDE 2001, pp. 607-614, 2001.
[5] M. Müller, T. Röder, and M. Clausen, "Efficient content-based retrieval of motion capture data," ACM Trans. on Graphics, Vol.24, No.3, pp. 677-685, 2005.
[6] B. Demuth, T. Röder, M. Müller, and B. Eberhardt, "An Information Retrieval System for Motion Capture Data," In Proc. of the 28th European Conference on Information Retrieval (ECIR), Vol.3936 of LNCS, pp. 373-384. Springer, 2006.
[7] K. Forbes and E. Fiume, "An efficient search algorithm for motion data using weighted PCA," In Proc. of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation, SCA'05, pp. 67-76. ACM, 2005.
[8] L. Kovar and M. Gleicher, "Automated extraction and parameterization of motions in large data sets," ACM Trans. on Graphics, Vol.23, pp. 559-568, August 2004.
[9] B. Krüger, J. Tautges, A. Weber, and A. Zinke, "Fast local and global similarity searches in large motion capture databases," In Proc. of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 1-10. Eurographics Association, 2010.
[10] K. Onuma, C. Faloutsos, and J. K. Hodgins, "FMDistance: A fast and effective distance function for motion capture data," In Short Papers Proc. of EUROGRAPHICS, 2008.
[11] Y. Sakamoto, S. Kuriyama, and T. Kaneko, "Motion map: image-based retrieval and segmentation of motion data," In Proc. of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation, SCA'04, pp. 259-266, Aire-la-Ville, Switzerland, Switzerland, 2004.
[12] L. Kovar, J. Schreiner, and M. Gleicher, "Footskate cleanup for motion capture editing," In Proc. of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation, pp. 97-104, ACM, 2002.
[13] J. Lee, J. Chai, P. Reitsma, J. Hodgins, and N. Pollard, "Interactive control of avatars animated with human motion data," ACM Trans. on Graphics, Vol.21, No.3, pp. 491-500, 2002.
[14] J. Wang and B. Bodenheimer, "Synthesis and evaluation of linear motion transitions," ACM Trans. on Graphics (TOG), Vol.27, No.1 p. 1, 2008.
[15] J. Barbič, A. Safonova, J.-Y. Pan, C. Faloutsos, J. K. Hodgins, and N. S. Pollard, "Segmenting motion capture data into distinct behaviors," In Proc. of the Graphics Interface, GI'04, pp. 185-194, 2004.
[16] G. Johansson, "Visual perception of biological motion and a model for its analysis," Perception and Psychophysics, No.2, 1973.
[17] J. A. Rice, "Mathematical Statistics and Data Analysis," Duxbury Advanced Series, 3rd edition, 2007.
[18] J. Aach and G. M. Church, "Aligning gene expression time series with time warping algorithm," Bioinformatics, pp. 495-508, 2001.
[19] H. Müller, W. Müller, D. M. Squire, S. Marchand-Maillet, and T. Pun, "Performance evaluation in content-based image retrieval: overview and proposals," Pattern Recognition Letters, Vol.22, pp. 593-601, April 2001.
[20] X. Jianfeng, K. Haruhisa, and Y. Akio, "Content-Based Retrieval of Motion Capture Data Using Short-Term Feature Extraction," IEICE Trans. on Information and Systems, Vol.92, No.9, pp. 1657-1667, 2009.

Name:
Worawat Choensawat

Affiliation:
Ph.D. Candidate, School of Science and Engineering, Ritsumeikan University

Address:
1-1-1 Noji Higashi, Kusatsu, Shiga 525-8577, Japan

Brief Biographical History:
2002 Master degree of Information Technology, King Mongkut's Institute of Technology Ladkrabang, Thailand
2004- Lecturer, Faculty of Information Technology, Dhurakij Pundit University, Thailand
2009- Research Assistance, Global COE: Digital Humanities Center for Japanese Arts and Cultures, Ritsumeikan University

Main Works:
• "Description and Reproduction of Stylized Traditional Dance Body Motion by Using Labanotation," Trans. of the Virtual Reality Society of Japan, Vol.15, No.3, pp. 379-388, 2010.

Membership in Academic Societies:
• Association for Literary and Linguistic Computing (ALLC)

Name:
Kozaburo Hachimura

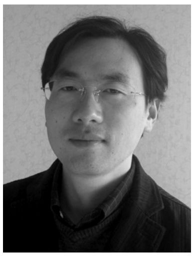
Affiliation:
Professor, College of Information Science and Engineering, Ritsumeikan University

Address:
1-1-1 Noji Higashi, Kusatsu, Shiga 525-8577, Japan

Brief Biographical History:
1978- National Museum of Ethnology
1983- Kyoto University
1994- Ritsumeikan University

Main Works:
• "Segmentation of dancing movement by extracting features from motion capture data," J. of the Institute of Image Electronics Engineers of Japan, Vol.37, No.3, pp. 303-311, 2008.

Membership in Academic Societies:
• Information Processing Society of Japan (IPSJ)
• The Institute of Electronics, Information and Communication Engineers (IEICE)
• The Virtual Reality Society of Japan (VRSJ)



Name:
Woong Choi

Affiliation:
Assistant Professor, Department of Information and Computer Engineering, Gunma National College of Technology

Address:
580 Tobamachi, Maebashi, Gunma 371-8530, Japan

Brief Biographical History:
2005 Received Ph.D. degrees from Tokyo Institute of Technology
2005- Post-Doc. at Center for Promotion of the COE, Ritsumeikan University
2010- Assistant Professor, Department of Information and Computer Engineering, Gunma National College of Technology

Main Works:
• "Quantification of Dance Movement by Simultaneous Measurement of Body Motion and Biophysical Information," Int. J. of Automation and Computing, Vol.4, No.1, pp. 1-7, January 2007.
• "Generation of Character Motion by Using Reactive Motion Capture System with Force Feedback," J. of Advanced Computational Intelligence and Intelligent Informatics, Fuji Technology Press, Vol.12, No.2, pp. 116-124, 2008.

Membership in Academic Societies:
• Institute of Electrical and Electronics Engineers (IEEE)
• The Virtual Reality Society of Japan (VRSJ)