Human Motion Retrieval from Hand-Drawn Sketch

Min-Wen Chao[1],Chao-Hung Lin[1],Jackie Assa[2],Tong-Yee Lee[1], *Senior Member*, *IEEE*
National Cheng Kung University, Taiwan[1]
Tel Aviv University, Israel[2]

Abstract—The rapid growth of motion capture data increases the importance of motion retrieval. The majority of the existing motion retrieval approaches are based on a labour intensive step in which the user browses and selects a desired query motion clip from the large motion clips database. In this work, a novel sketching interface for defining the query is presented. This simple approach allows users to define the required motion by sketching several motion strokes over a drawn character, which requires less effort and extends the users' expressiveness. To support the real-time interface, a specialized encoding of the motions and the hand-drawn query is required. Here, we introduce a novel hierarchical encoding scheme based on a set of orthonormal spherical harmonic basis functions (SHs), which provides a compact representation, and avoids the CPU/processing intensive stage of temporal alignment used by previous solutions. Experimental results show that the proposed approach can well retrieve the motions, and is capable of retrieve logically and numerical similar motions, which is superior to previous approaches. The user study shows that the proposed system can be a useful tool to input motion query if the users are familiar with it. Finally, an application of generating a 3D animation from a hand-drawn comics strip is demonstrated.

Index Terms—motion retrieval, spherical harmonic function, sketching interface.

1 Introduction

Over the last decade, we witness an explosion in the usage of motion capture (mocap) data in games and animations. The manipulation of mocap data by methods such as motion retargeting [1], motion overview [2] and motion synthesis [3], [4], [5], [6], [7], [8], [9] requires an easy, efficient and accurate retrieval of similar motions from a large data repository. Therefore, it is important to provide an appropriate interface to generate queries for motion retrieval. Previous approaches utilize an existing motion clip as a query term. This generally requires users to examine a large number of examples in the motion repository by reviewing the motions, which is very tedious and time-consuming [10], [11], [12], [13], [14]. A different common retrieval method is by using the motion textual description such as "walking" or "running". Although it is very efficient in text matching as well as retrieval, textual descriptions cannot always sufficiently express 3D motions and requires manual work in annotating the motions in the repository.

Motion lines sketches is an effective technique to convey motion, as shown in traditional comics [15]. Following this observation, we propose a novel sketching interface which allows drawing of motion lines over a character figure, and show that these details are sufficiently expressive for locating similar motions. Our proposed method allows iterative refinement of the selections, restricting the motion to fit a more accurate pose description. By combining with a fast encoding of the query and motion repository, the proposed system can be used in interactive scenarios.

Conveying a motion by using sketches presents a new challenge in determining the desired temporal sequence of the motion details. However it overcomes the known time-warp effort required by many of the existing motion retrieval systems [10], [11], [16], [17]. This effort usually requires either the repository motion or the query sequence to be warped in many temporal scales, so that different motion speeds would not filter out suitable results. The key idea behind the proposed scheme is representing the motion trajectories by using a complete set of SHs, which demonstrates several suitable properties. The trajectory represented by a few SHs (a coarse but smooth approximation) is shown to be similar to the motion strokes. Moreover, the SH encoding reduces the data description dimensions and presents a compact shape descriptor, reducing both the storage size and search time. Finally, the inherent properties of rotation-invariant and multi-scale nature of SHs encoding allows an efficient matching and indexing of the motion database.

While the usage of SHs in general data retrieval was suggested before [18], [19], to the best of our knowledge, the proposed approach is the first to utilize SHs in the encoding of mocap data, i.e., a time-varying character motion data. Our work therefore presents two major contributions: 1) offering a natural sketching interface for describing the query, and 2) introducing a novel motion encoding method for retrieving motion in a coarse-to-fine manner.

2 Related Work

Motion indexing and retrieval, first proposed by Liu et al, had become crucial to the reuse of motion capture data in games and animations. Over the time, several methods were suggested, which can be classified into two categories, content-based retrieval [10], [11], [16], [17], [20] and numerical-based

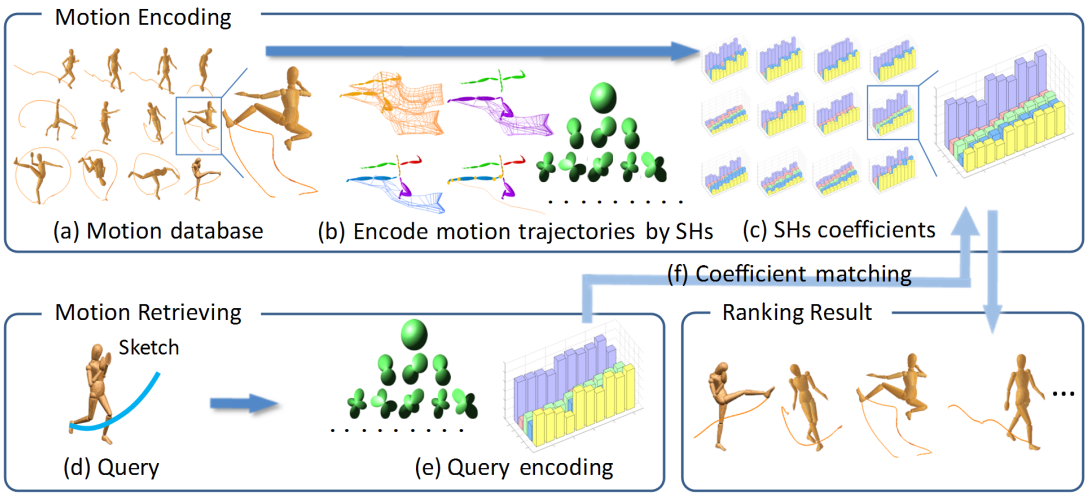

Fig. 1. System workflow. An illustration of (a) motion database and (b) encoding motion trajectories by SHs; (c) SHs coefficients (3^{rd} level of articulated hierarchy); (d) drawing a sketch as query; (e) encoding query by SHs; (f) coefficient matching and ranking.

retrieval [12], [13], [14], [21], [22], [23], [24], based on the measurement used to measure motion similarity (i.e., logical or numerical similarity).

The methods based on numerical similarity usually measure the distance between poses as a difference of the joint positions. One of the main challenges in the numerical-based approaches is how to handle any temporal difference between the query motion and the motions in the database. One of the leading methods for handling this challenge is dynamic time warping (DTW) which exhaustively compares the query and result clip frames to non-linearly align the time in both clips. As a result, DTW had become essential to every motion indexing structure and is applied to many motion repositories [10], [11], [16]. The work of Chiu et al. introduces a two-stage motion retrieval approach [10]. They identify the start and end frames of possible candidate motion clips using a pose-based index and calculate the similarity between the query motion clip and each candidate clip using DTW. The work of Forbes and Fiume suggests a search algorithm based on weighted principle component analysis (wPCA) [11]. Each motion in the database is represented as a high-dimensional parametric curve in a wPCA space. They apply a space-search strategy to prune away non-relevant motions, as a preliminary step before applying the DTW on the remaining results to locate the best matching answers. Kovar et al. propose a multi-step search strategy [16]. They first reproduce new queries from previously retrieved motions in order to find more relevant motions. This approach can extract a family of related motions that are particularly suitable for the motion blending applications. The pairwise comparison of motions in these methods for the database motions is time-consuming which makes the related applications infeasible for a large data set. To avoid the DTW effort, our method encodes the trajectories and the spatiotemporal variances of motions by SHs. Thus, the process of time-warping is not required and the query time and storage space is considerably reduced.

Logical similarity was introduced by Müller et al. [13] and later extended by Lin [23] and Müller et al. [24] (i.e., content-based retrieval). Logical similarity uses boolean geometric pose properties which describes geometric relations between several specified body joints. Using this representation, a complex motion retrieval problem is translated to a simple bit string matching problem, which is significantly faster than comparing the original numeric vector describing the motion, and easier to maintain. Nevertheless, the selection of proper geometric relations in addition to a motion query highly affects the performance and quality of the results. Recently, an approach based on motion pattern extraction and matching was suggested by Deng et al. [14]. After picking a motion clip as query input, this approach can efficiently retrieve logically similar motions. Our method includes a sketching interface for describing motion query is provided. Users do not need to specify a short motion clip and even additional geometric relations. The work of Ho et al. introduces an indexing approach for multiple characters [25]. They represent and encode topological relationships of multiple characters by rational tangles. Thus, this approach can be applied to scenes with close interactions between multiple characters.

Recently, sketch-based interface research was suggested in several fields. A number of researches propose sketching interfaces for 3D modeling [26], [27], [28] and motion creation [7], [29], [30]. In the work of Davis et al. [29], an interface for creating 3D articulated figure animation is presented. This system infers 3D poses from 2D sketches using the information of bone lengths. However, to create a motion, several successive key poses must be drawn by users. Thorne et al. introduce a more convenient approach where a motion is created by sketching a continuous sequence of lines, arcs, and loops [7]. They are parsed and then mapped to a parameterized set of motions. A sketch system for creating Kung-fu motions is presented in the work of Li et al. [30]. They use sketches to describe the first and last postures as. Next, they retrieve the similar postures as well as the in-between motions by

projecting the frames and the 3D database motion trajectories to 2D plane, and matching them in that space. As a result, the 2D plane projection causes significant motion information to be disregarded during the query. In this paper, a single sketch containing some motion strokes (i.e., similar to traditional comics) is taken as query input. The motion trajectories are encoded by SHs and directly compared in 3D.

3 System Overview

The proposed system, shown in Figure 1, consists of two major parts: *motion encoding* and *motion retrieval*. The main idea is to encode the motion trajectories of the query and the clips in the database by using a small set of SHs (Figure 1(b), (e)). This allows an efficient indexing of the motions and a fast retrieval, by matching the SHs coefficients of the motions. To support complex motion clips which contain several actions, we begin by splitting such clips into sub-clips. Each sub-clip contains only one action. In our implementation, the segmentation is done by locating the key poses. As shown by Barbic et al. [31], other robust motion segmentation approaches can be used for this purpose as well. Next, we encode the full-body motion according to the character body coarse to fine hierarchy. This hierarchy is organized in 4 levels: full body, upper/lower body, all the body main limbs (leg, arm, etc), and the body joints. This separation introduces 39 trajectories for each clip, which are used as the motion description. This description supports queries describing movement in several scale - both full body motion and up to movement of the various joints, and lets our system has the flexibility and ability of retrieving logically and numerically similar motions. To retrieve motions, the user sketches the desired motion (Figure 1(d)). The 3D trajectory of query is inferred from the 2D motion strokes, and then encoded too by SHs. Utilizing the inherent multiresolution property of SHs, a coarse-to-fine coefficient matching is adopted to speed up the retrieval.

4 Spherical Harmonics

To encode the motions in database, a small set of spherical harmonics are used to represent the motion trajectories. Following is a brief introduction to spherical harmonics (SHs). For a more comprehensive overview of SHs, we refer the readers to [32]. A spherical harmonic of degree l and order m, $Y_l^m(\theta,\phi)$, is defined as:

$$Y_l^m(\theta,\phi) = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}} P_l^m(\cos\theta) e^{im\phi} \tag{1}$$

where l and m are integers that satisfy $l \geq 0$ and $|m| \leq l$; $\theta \in [0,\pi]$ and $\phi \in [0,2\pi)$ represent latitude and longitude, respectively. The associated Legendre polynomial P_l^m in Eq. (1) is defined as:

$$P_l^m(x) = \frac{(-1)^m}{2^l l!}(1-x^2)^{m/2}\frac{d^{l+m}}{dx^{l+m}}(x^2-1)^l \tag{2}$$

The spherical harmonic functions constitute a complete orthonormal system on a sphere. Any square-integrable functions $f(\theta,\phi)$ on a sphere can be expressed by a linear combination of these:

$$f(\theta,\phi) = \sum_{l=0}^{\infty}\sum_{m=-l}^{l} a_l^m Y_l^m(\theta,\phi) \tag{3}$$

where a_l^m is the coefficient of spherical harmonic $Y_l^m(\theta,\phi)$. Given a maximum degree (or called bandwidth) l_{max}, an orthonormal system expanded by spherical harmonics involves $(l_{max}+1)^2$ coefficients. For a function with n spherical samples (θ_i,ϕ_i) and their function values $f_i = f(\theta_i,\phi_i)$, $1 \leq i \leq n$ (i.e., f_i is calculated by Eq.(3)), these coefficients a_l^m can be obtained by solving a least-square fitting [33]:

$$\begin{pmatrix} y_{1,1} & y_{1,2} & \cdots & y_{1,k} \\ y_{2,1} & y_{2,2} & \cdots & y_{2,k} \\ \vdots & \vdots & & \vdots \\ y_{n,1} & y_{n,2} & \cdots & y_{n,k} \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_k \end{pmatrix} \tag{4}$$

where $y_{i,j} = Y_l^m(\theta_i,\phi_i)$ and $b_j = a_l^m$, $j = l^2+l+m+1$ and $k = (l_{max}+1)^2$. Any spherical function can be represented by three explicit functions $f(\theta,\phi) = (f_x(\theta,\phi), f_y(\theta,\phi), f_z(\theta,\phi))$ and the coefficients calculated by Eq. (4) are 3-tuple vectors $a_l^m = (a_{lx}^m, a_{ly}^m, a_{lz}^m)$. By utilizing the fact that the L2-norm of SHs coefficients is rotation-invariant [32], we represent a motion as a spherical function $f(\theta,\phi)$ and then encode it as:

$$SH(f(\theta,\phi)) = (\{\|a_0^m\|\}_{m=0}^0, \{\|a_1^m\|\}_{m=-1}^1, \cdots, \{\|a_{l_{max}}^m\|\}_{m=-l_{max}}^{l_{max}}) \tag{5}$$

This motion encoding is compact and has several useful properties for retrieval which will be discussed in Section 8.1.

5 Motion Encoding

We define the sequence of connected joint locations over time as a 'trajectory surface' – a manifold in spatial-temporal space. The key idea behind the proposed scheme is representing the trajectory surface by SHs. In this section, we first describe the preprocessing phase (Section 5.1) followed by the SHs encoding phase (Section 5.2).

5.1 Data preprocessing

To reduce the sensitivity to different character size and proportions, and to use a common coordinate system, we first normalize and align the motion data. The normalization is performed by rescaling the skeleton to have a standard distance between root and neck joints. Since the SHs representation is translation-sensitive, we align the motions by translating the character root in all poses to the origin of a unit sphere $S(\theta,\phi)$, while the origin of unit sphere is set to the origin of world coordinate. The poses in a motion (i.e., the significant characteristics) are preserved in this preprocess and thus, the motion retrieval will not be affected. If the absolute details are required, they can be added in the encoding of the root joint. Next, to retrieve similar motions, the motion is captured by a multi-level structure (see Figure 2). The first level is the whole body. Then, the second level includes two body parts: upper body and lower body, and the following level includes the body limbs (arms and legs) and head. In the last level,

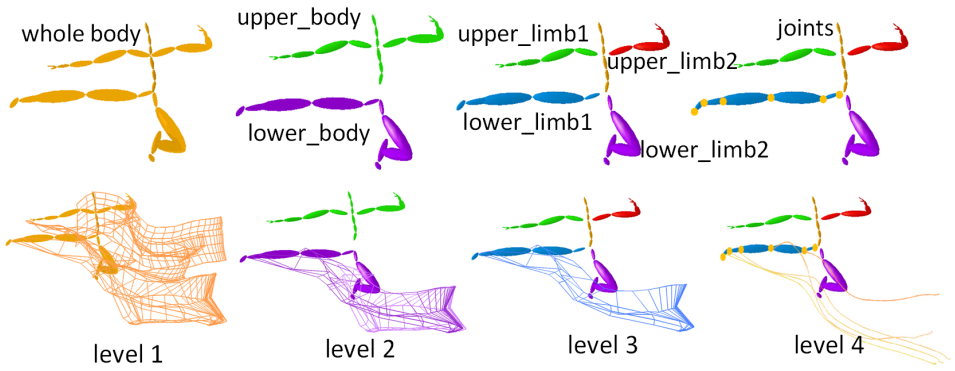

Fig. 2. An illustration of the different encoding levels (top) and their respective motion trajectories (bottom).

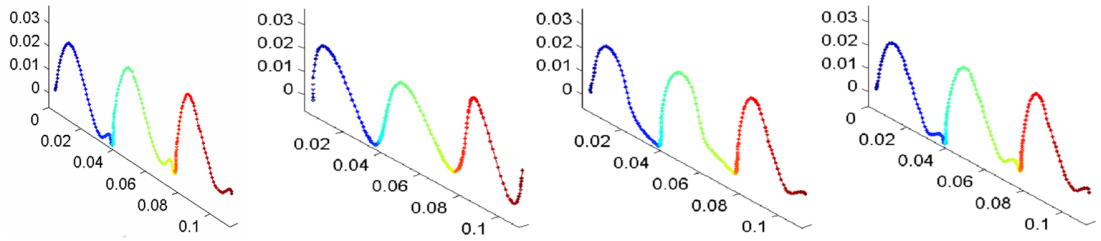

Fig. 3. The joint trajectory (left ankle of walking motion) reconstruction. From left to right: The original joint trajectory, the reconstruction results of different setting of $l_{max} = 2, 9, 29$, respectively.

each node contains the trajectory of a joint of the original motion, and all joint trajectories are saved in this level. In this manner, the proposed system is capable of retrieving logically similar motions by using the information of higher levels of articulated hierarchy (1^{st} or 2^{nd} level) and retrieving numerical similar motions by using the information of lower levels (3^{rd} or 4^{th} level). A demonstration of this property will be shown in the Section 8.2.

5.2 Encoding motion trajectory with spherical harmonic functions

The trajectory surface of a given motion is first transformed to a spherical system in which the spherical coordinate (θ, ϕ) of each joint position is calculated, the trajectory surface is approximated by SHs (using Eqs. (3)-(5)). The L2-norm of SHs coefficients is used to encode the joint trajectories. Each motion M is encoded as:

$$SH(M) = (SH(M_{level1}), SH(M_{level2}), SH(M_{level3}), SH(M_{level4})) \quad (6)$$

where M_{leveli} represents the trajectory surface of the nodes in level i. Note that the original joint trajectories can be reconstructed/approximated by these coefficients with SHs:

$$f'(\theta,\phi) = \sum_{l=0}^{l_{max}} \sum_{m=-l}^{l} a_l^m Y_l^m(\theta,\phi) \approx f(\theta,\phi) \quad (7)$$

The trajectories approximated by SHs with different maximum degree of l_{max} are shown in Figure 3. Using additional coefficients can achieve a more accurate reconstruction, whereas fewer coefficients have similar effect to curve smoothing (which is similar to the user-drawn motion stroke in shape). In our experiments, the maximum degree l_{max} is set to 4 and the L2-norm of coefficients $|a_l^m|$ encodes the motion trajectory. Since the coefficients a_l^m are complex conjugate of the coefficients $-a_l^m$, i.e., $|a_l^m| = |-a_l^m|$, we only use the coefficients a_l^m with $m \geq 0$ (i.e., 15 coefficients) to encode the trajectory in a node for the purpose of efficient retrieval and storage. As a result, given an n_{joint}-joints action, $15 * (n_{joint} + 8)$ coefficients in total are used to encode and index the action regardless of its temporal length and speed.

6 Sketching Interface

The sketching interface allows users to define the motion by simply drawing a motion line [34] (called "motion stroke") describing the character motion. The motion stokes are drawn on a selected character pose and a selected view. The main goal of the method is to infer the corresponding 3D trajectories from the hand-drawn strokes. We begin by allowing the user to select a desired character pose and camera view from a set of predefined key-poses and their default camera viewpoints. Next, the user specifies the motion by adding strokes to the character. Using a single key posture for motion retrieval is insufficient to distinguish between motions, as shown in figure 4. In out method, we propose using motion lines, on top of the selected posture, to better define the desired motion. For example, in Figure 4, we add different motion lines to retrieve different motions.

Next we predefine the motion from the motion strokes in the following manner. Each of the hand-drawn motion strokes is approximated to an ellipsoid. This type of quadric surface is

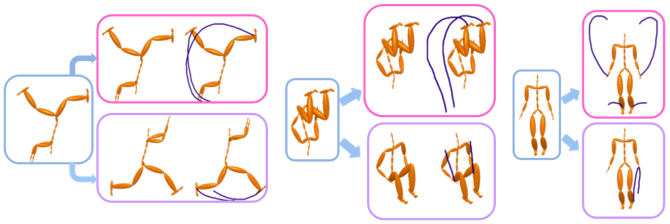

Fig. 4. Different motions may have similar key postures. For example, the motions of hand spring and leap (left), the motions of backflip and sitting (middle), and the motions of jumping jack and leap.

a good candidate for trajectory fitting of many hand-drawn sketches, and is relatively robust to the camera viewpoint differences. We assign each stroke to the joint nearest to its starting point. The corresponding joint is called active joint, denoted as $Joint_A$ shown in Figure 5. In addition, we define a pivot joint, denote as $Joint_P$, for each active joint. In Figure 5, for example, the active joint is the ankle joint, and its pivot joint is the hip joint. An ellipsoid is defined by a center $C:(x_c,y_c,z_c)$ and three principle axes: $axis_a$, $axis_b$ and $axis_c$. The center is set to the position of pivot joint $C = Joint_P$. We first determine the ellipse formed by the first two axes $axis_a$ and $axis_b$, and then determine the third axis $axis_c$. Here, two reference points V_{ref1} and V_{ref2} with the maximum and minimum pressures are used to determine the ellipse. We assume that the length of the major axis $|axis_a|$ is the length of revolving bones (from the pivot joint to the active joint), and these two reference points lie on the ellipse. The depth (z-coordinate) of the reference points are inferred from the stroke pressure $P:[0,1]$ (see Figure 5). The full $(P=1)$ and empty pressure $P=0$ represent the movement limits of revolving bones in the +z- and -z-direction, respectively. The depth of these two extreme end points ($P=1$ and $P=0$) is calculated by

$$len_z = 2\sqrt{|axis_a|^2 - |Joint_A - Joint_P|^2_{xy}} \tag{8}$$

where $||^2_{xy}$ represents the distance in xy-plane. Then, for an arbitrary point X with pressure $P(X)$, its z-coordinate z_x is calculated by

$$z_x = z_{Joint_P} + (2P(X)-1) \times len_z/2 \tag{9}$$

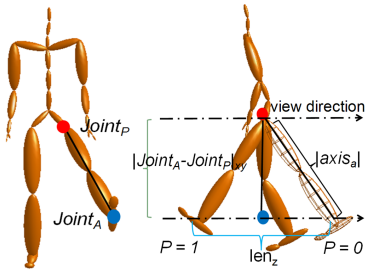

Fig. 5. An illustration of pivot and active joints (left) and the depth calculation (right).

To find an ellipse under the constraints of $|axis_a| = \|z_{Joint_A} - Z_{Joint_P}\|$, the axis $axis_a$ lying between $\overrightarrow{CV_{ref1}}$ and $\overrightarrow{CV_{ref2}}$ and the ellipse passing through the reference points V_{ref1} and V_{ref2}, a binary search strategy is adopted. In each step, the major axis is set to the middle of the search range $\lfloor \overrightarrow{CV_a}, \overrightarrow{CV_b} \rfloor$. Then, we find two ellipses with $|axis_b| = b_1$ and $|axis_b| = b_2$, respectively. If $b_1 > b_2$, we will search the left half range $\lfloor \overrightarrow{CV_a}, \overrightarrow{CV_m} \rfloor$. IF $b_2 > b_1$, we will search the right half range $\lfloor \overrightarrow{CV_m}, \overrightarrow{CV_a} \rfloor$. This process will continue until an ellipse is found, (i.e., $b_1 = b_2$). Next, we determine the length of $axis_c$ which is affected by the stroke bounding box (i.e., the 3D box with the smallest space which the stroke lies within) as shown in Figure 6. Therefore, the length of $axis_c$ must satisfy the following requirement: the projection of ellipsoid (project to the working plane) just encloses the motion stroke. Similarly, a binary search strategy is adopted here. The ellipsoid formed by an estimated $|axis_c|$ is first projected to the working plane, and then check to see if the requirement is satisfied. If not, the ellipsoid formed by $|axis_c|/2$ or $3|axis_c|/2$ is checked. This process continues until the requirement is met (i.e., $b_1 = b_2$).

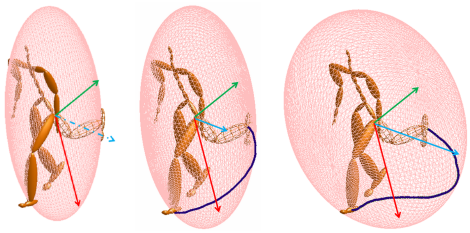

Fig. 6. The ellipse in the working plane (left) and two examples of the axis c (middle and right).

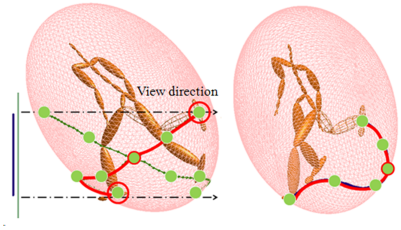

Fig. 7. An example of motion stroke projection. Left: two possible motion paths (red and green paths); Right: the correct motion path (the red path).

Once the ellipsoid is created, the 3D path of motion stroke can be obtained by simply projecting the motion stroke to the ellipsoid in +z or -z direction. Therefore, two possible 3D paths on the ellipsoid can be obtained (see Figure 7 Left). The desired path will be the one which is close to the active joint as shown in Figure 7 Right. In Figure 8, we show the results of lifting motion strokes with different stylus pressures to 3D. The pressure is represented by line width. In this example, the stroke with higher pressures generates a motion path with larger lifting. Note that specifying depth of the hand-drawn stroke by using stylus pressure would be not easy for some beginners. An alternative approach is to design a set of patterns with various line widths. The line width means the pressure and depth. In sketch, the user selects a pattern for a motion

stroke. Then, the sketched path is lifted according to the line width. The depth calculation is similar to Eqs. (8) and (9).

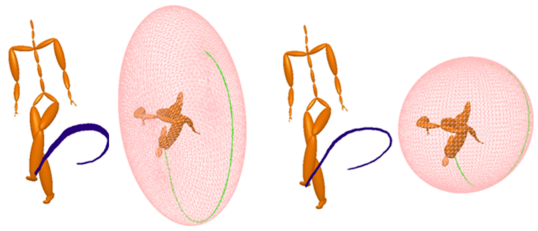

Fig. 8. 3D trajectories inferred from motion strokes with different stylus pressures.

7 Indexing and Ranking

Our encoding is multi-scale. This property allows an efficient coarse-to-fine indexing and ranking scheme. A 31-joints skeleton motion is represented by 39-nodes as shown in Figure 2, each encoded by using 15 coefficients to describe its motion: $\{(a_0^m)_{m=0}^0, \cdots, (a_4^m)_{m=0}^4\}$ (each $|a_l^m|$ is the coefficient of degree l and order m.). To efficiently retrieve similar motion, the following ranking strategy is used. The three lowest-frequency coefficients are used to define the criteria of the retrieved population. The rest of coefficients (i.e., $(|a_2^m|)_{m=0}^2$, $(|a_3^m|)_{m=0}^3$, $(|a_4^m|)_{m=0}^4$) are used to rank the resulting motions within the returned motions population. This provides a good set of alternatives from which the user can select the motion which is best suited for his purposes from a larger set of motions which are similar to the query. To better focus the retrieval on the action characteristics, we consider the node significance by applying a simple weighting scheme based on the motion area suggested by Kwon and Lee [35]. Specifically, for the query, we determine a weight for each node according to its movement. For example, the nodes of the lower body have larger weights in a kicking motion, and the nodes of the upper body have larger weights in a punching motion. As a result, a higher contribution is given to distance between nodes with larger saliency, as shown next:

$$dist(M,Q) = w_i \| SH(M_{node_i}) - SH(Q_{node_i}) \|, \quad node_i \in \text{a specific level of hierarchy} \quad (10)$$

where M and Q represent a motion in the database and query, respectively.

8 Results and Discussion

8.1 Properties with SHs motion encoding

Motion matching based on our encoding introduces several properties. First and foremost, our method provides a metric in which similar motions have small distance and dissimilar ones have larger distance. To demonstrate this property, two sets of similar motions are tested. The encoding results are shown in Figure 9. The walking motions of normal-speed, fast, slow, leg-wide, and backward have distinguishable coefficients. The jump-kicking motions from running, spinning and walking also have distinguishable coefficients. Second, the SHs coefficients are inherently rotation-invariant. As shown in Figure 10, the coefficients remain unchanged when rotations are applied to a motion. Compared to the traditional wavelet transform, the SHs encoding has no restriction on the data size and the property of rotation invariant is intrinsic. Although some advanced works on wavelet transform had solved these two problems by using additional processes, the SHs encoding is more suitable in motion encoding, in terms of simplicity. Third, our method does not require any time-warping and time alignment processing since the similarity is computed regardless of motion speeds and lengths. Figure 11 shows an experiment on similar action motions which are temporally different. The generated coefficients of the motion trajectories are similar. Another example is shown in Figure 12 where two walking motions are presented and the coefficients of a single cycle of walking sequence are similar to that of multiple cycles of walking sequence. These two examples also demonstrate how repetitive motions are efficiently encoded and retrieved. Fourth, the SH encoding leads to a reduction of dimensionality in shape description since only a small set of SHs coefficients is used.

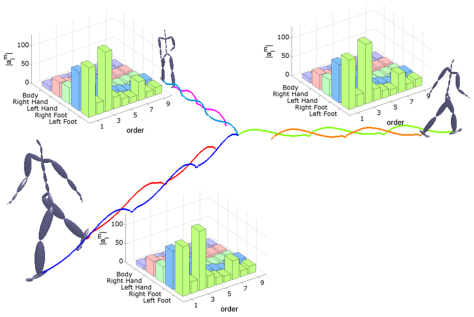

Fig. 10. Demonstration of rotation invariance. The coefficients remain unchanged when rotations are applied to a motion.

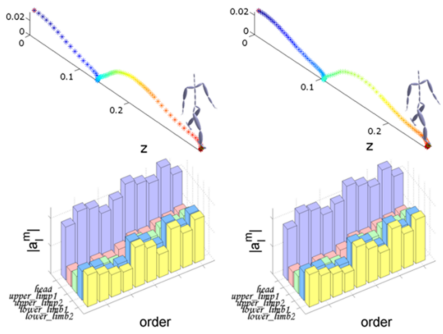

Fig. 11. Encoding of time-warped motions. Top: The time-warped walking motions (points represent the location of ankle joint at each frame in the clip); Bottom: The encoding results.

8.2 Encoding Results

To demonstrate the feasibility of the proposed encoding approach, a database containing 21 motion classes was tested

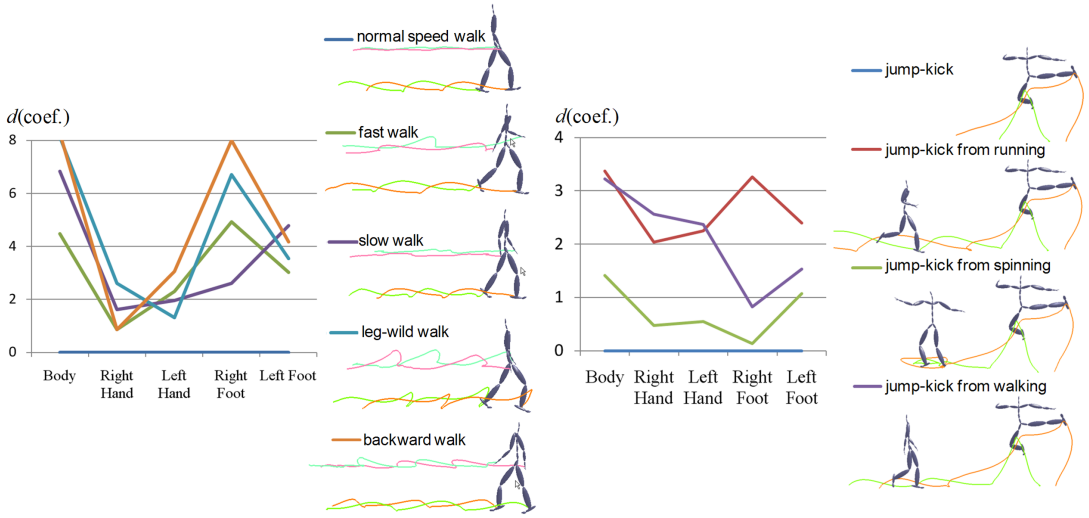

Fig. 9. Left: the SH coefficients of various similar walk motions including normal-speed, fast, slow, leg-wild, and backward walking motions. Right: the SH coefficients of various similar jump-kick motions including normal jump-kick, jump-kick from running, spinning and walking. The y-axis represents the difference of coefficients between the normal-speed walk (left) or normal jump-kick (right) and the tested motions.

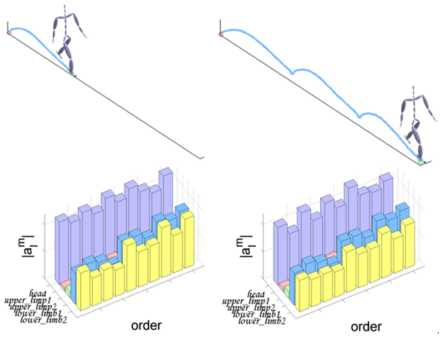

Fig. 12. Encoding (bottom) of repetitive motions (top). Left: a single walk cycle; Right: three cycles of a walk sequence.

(see Figure 13). The motions are encoded and plotted on a 2D frame according to their coefficients. The axes are determined by the principle component analysis (PCA) of the joints' coefficients. The motions are displayed by colors. The result shows that similar motions can easily be grouped by their coefficients and thus, the proposed encoding approach can well distinguish these motions.

8.3 Retrieval Evaluation and Applications

Database and Timing. We tested our approach on a database containing about 6000 clips with 1009700 frames from the public CMU motion database [36]. All experimental results are evaluated on a PC with a 2.13 GHz CPU and 2.0GB memory. Our system takes on average 18 milliseconds to search the database. The time to process a query is linearly dependant on the size of the database.

Retrieval evaluation. To evaluate the retrieval accuracy of our method, we compare with the geometric indexing approach of Müller et al. [13] and the DTW-based approach suggested by Keogh et al. [17]. Here, instead of using the sketching interface, we use a selected database motion as a query in a similar manner to the evaluation of [13]. These approaches are tested on two datasets containing 30 manually selected clips with similar motion (see Figures 14 and 15). Some of

TABLE 1
Accuracy comparison between our approach using SH coefficients in level 2 and 3, and the related approaches [13], [17]. The tested dataset and query are shown in Figure 14. The accuracy measurement, precision η_s, recall η_n, and accuracy τ, are used with optimal cutoff thresholds, and a numerical-based ground truth (the top figure) and a logical-based ground truth (the bottom figure)

	$\eta_s(\%)$	$\eta_n(\%)$	$\tau(\%)$
SH Level 2	60.78	100	70
SH Level 3	100	100	100
G.F.	100	85.71	93.33
DTW	100	64.29	83.33

	$\eta_s(\%)$	$\eta_n(\%)$	$\tau(\%)$
SH Level 2	100	100	100
SH Level 3	75	100	80
G.F.	100	50	70
DTW	85.71	100	90

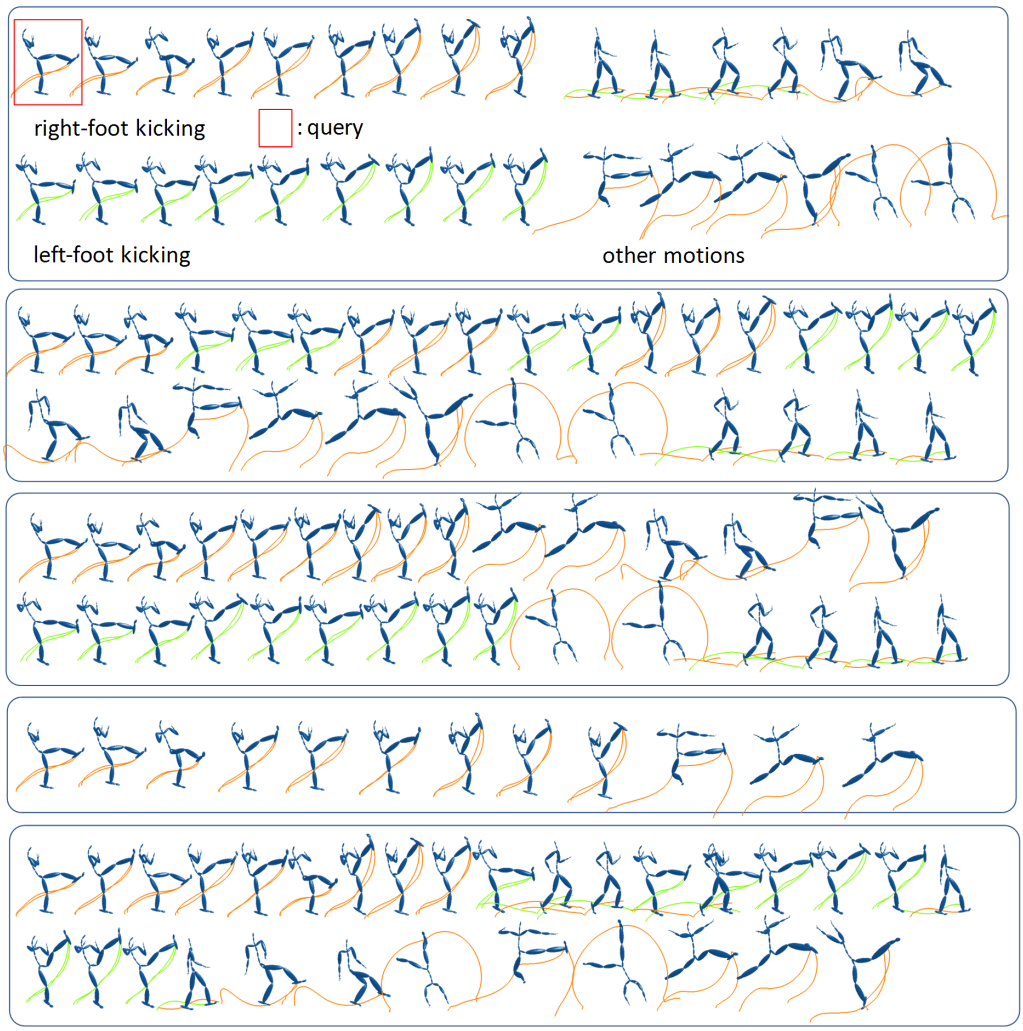

Fig. 14. The evaluation of motion retrieval. 1^{st} row: the selected dataset; 2^{nd} - 3^{rd} rows: our approach using the coefficients of the lower part in level 2 and the lower limbs in level 3, respectively. The weights for other parts are set to 0.0; 4^{th} row: retrieve by using geometric relations [13]. The geometric relations F_l^3, F_l^4 (left/right foot raised, please refer to Table 1 in [13]), F_l^7, F_l^8 (left/right knee bent) are selected; 5^{th} row: retrieve by using DTW. The clip marked by red rectangle is the query. The trajectories of right-foot and left-foot are displayed by red and green colors, respectively.

these clips capture different character actions. Figure 14 shows a retrieval result for a simple kicking query (marked by red quadrangle). The dataset contains left-foot kicking, right-foot kicking and other semantic motions. Figure 15 shows a query of a punch motion. The dataset contains left-hand punching, right-hand punching and other motions. From the rankings, we can see that our approach can well retrieve the motions and is capable of retrieve logically or numerical similar motions. For example in Figure 14, the query is a motion of left-foot kicking. Using the coefficients of upper part in level 2, we extract all kicking clips regardless of right-foot or left-foot kicking (i.e., logically similar motions). Using the coefficients of upper limbs in level 3, we extract all right-foot kicking clips (i.e., numerical similar motions). In the evaluation of retrieval accuracy (see Table 1), the commonly-used measurements precision η_s, recall η_n, and accuracy τ are used. They are defined as $\eta_s = \frac{TP}{TP+FP}$, $\eta_n = \frac{TP}{TP+FN}$, and $\tau = \frac{TP+TN}{TP+TN+FP+FN}$, where TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative, respectively. To fairly compare the approaches, the optimal cutoff threshold and the correct classifications (i.e., ground truths) defined based on numerical and logical measurements are tested. If the logical-based ground truth is used, our approach using the coefficients in level 2 and the logical-based approach [13] obtain better results (see the top table in Table 1). If the numerical-based ground truth is used, our approach using the coefficients in level 3 and the numerical-based approach [17] obtain better results (see the bottom table in Table 1). This experiment shows that the proposed approach with the ability of providing both numerical-based and logical-based retrieval is superior to previous approaches.

Search by sketches. In our system, the user can sketch some motion strokes as query input. Figure 16 shows the retrieval results using the sketch system. For clearly evaluating

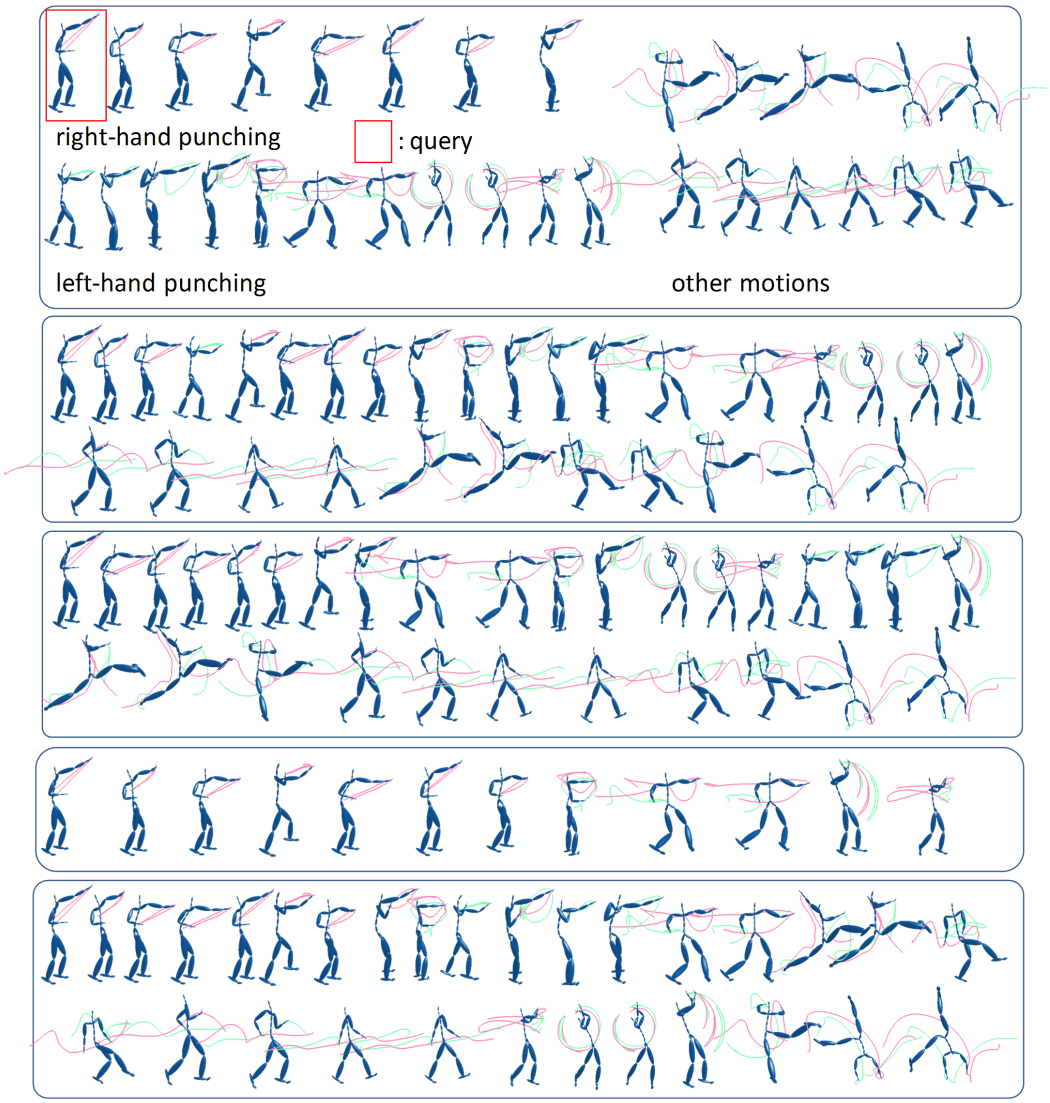

Fig. 15. The evaluation of motion retrieval. 1^{st} row: the selected dataset; 2^{nd} - 3^{rd} rows: our approach using the coefficients of the upper part in level 2 and the upper limbs in level 3, respectively; 4^{th} row: retrieve by using geometric relations [13]. The geometric relations F_u^1, F_u^2 (left/right hand in front), F_u^7, F_u^8 (left/right elbow bent) are selected; 5^{th} row: retrieve by using DTW. The trajectories of right-hand and left-hand are displayed by red and green colors, respectively.

the retrieval results, we refer the reader to the material in http://140.116.80.113/Motion Retrieval/Demo.mp4. Note that our sketching system uses a stylus and tablet, which allow us to explore additional attributes of the stroke, such as the pressure the user applies at each section of the motion stroke. Although it could be not easy to use stylus with pressure to draw motion strokes for novice tablet beginners, the motion stroke can become a good alternative to input motion query once the users are familiar with this input tool.

User study. In order to evaluate the proposed sketch interface, we conducted a user study involving 29 computer graphics students and researches with ages ranging from 21 to 37 years old. It is likely unworkable to use a dataset containing too many motion categories in the user study. Participants are likely to lose their patience/concentration in a long user study. Therefore, at least the motion categories tested in the related works [13], [16], [37], [21], [22], [23], [24] are all included in our user study. Participants were required to sketch motion lines to retrieve 16 motions (including kicking, running, walking, jumping, squatting, cartwheel, jete, backflip, spinning, hand spring, and punching (see the top figure in Figure 17)) by the proposed system after about 15 minutes introduction. Some of the sketched lines and retrieval results are shown in Figures 17_Bottom and 16, respectively. As expected, there exists some variance among peoples drawings. However, these drawings can retrieve similar motion clips, since their SH encodings are similar to that of retrieved clip. In this study, our goal was to test if the users can successfully retrieve the

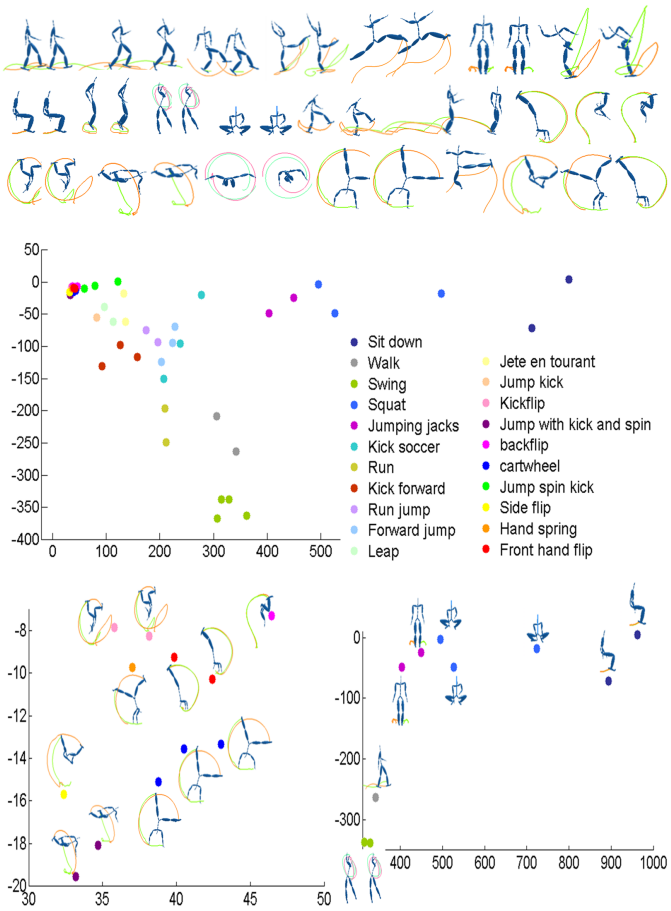

Fig. 13. Motion encoding result. A database containing 21 classes of motions is tested (top). The motions are displayed by colors (blue: cartwheel; red: front hand flip; pink: kickflip; yellow: side flip and so on) and plotted on a 2D frame (bottom).

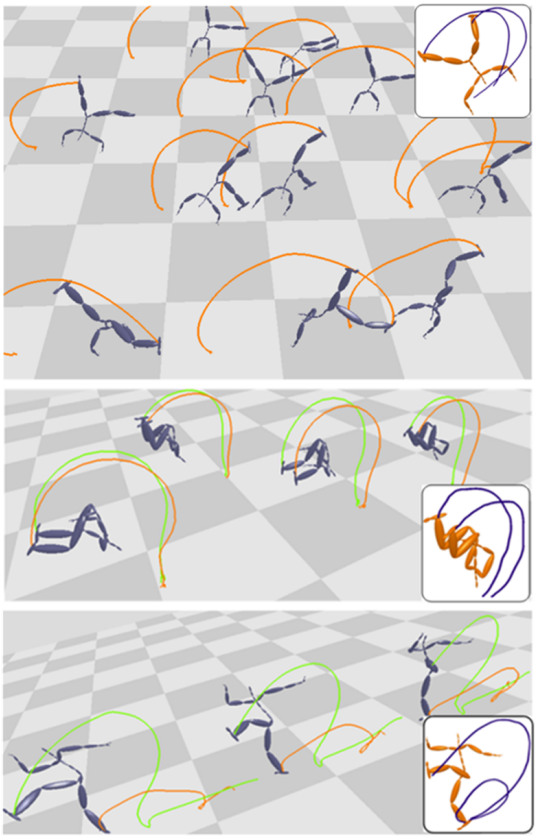

Fig. 16. The retrieval results.

Fig. 17. Top: The motions tested in the user study; Bottom: The motion lines sketched by users (light colors) and the corresponding motion trajectories (dark colors).

desired motions, how long it takes, and if the proposed system is easy to use. The user response time indicates that the users can draw motion lines and extract the desired motions in 3-10 seconds in simple cases (such as simple kicking, jumping, and punch) and in about 30 seconds for complex actions (such as spinning, cartwheel, and hand spring). The results also show that the users with drawing experience or ones with prior sketching experience yield better results than no experience. Figure 18 shows the user satisfaction surveys. The surveys indicate that 62% users agree that the proposed system is easy to use, but 10% users think the proposed system is hard to use, and 89% users would like to use the proposed system if the data is very huge. These surveys show that the proposed system can be a useful tool to input motion query if the users are familiar with it.

Application. The user working with our system can compose complex motions from a sequence of desired actions as shown in Figure 19. In this case, the system uses an additional constraint to describe the common poses between the joined actions, and smoothly concatenates these extracted motions using motion blending techniques [38].

9 Conclusions, Limitations and Future Work

We have presented a novel motion retrieval approach which uses a hand-drawn sketch as query input. With a novel sketching based interface, our system demonstrates a simple and novel motion-encoding approach. Users sketch some motion strokes on a selected character in a single view. The SHs encoding is highly suitable for this scenario: it removes the

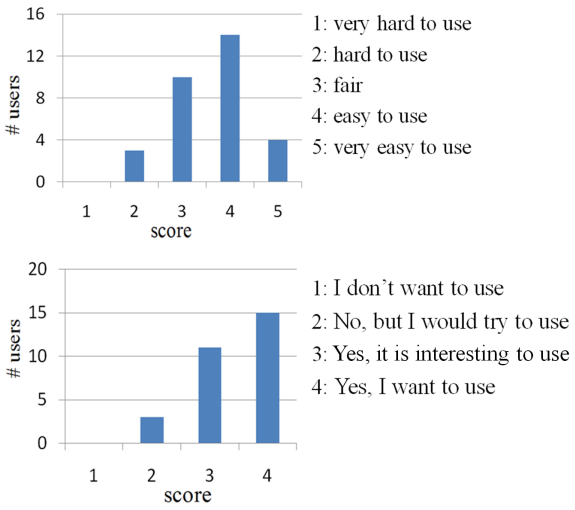

Fig. 18. User satisfaction survey. Top: The survey question: please rate the usefulness of the system. Score 5 indicates positive result. Bottom: The survey question: do you want to use this system to retrieve motions if the data is very huge? Score 4 indicates positive result.

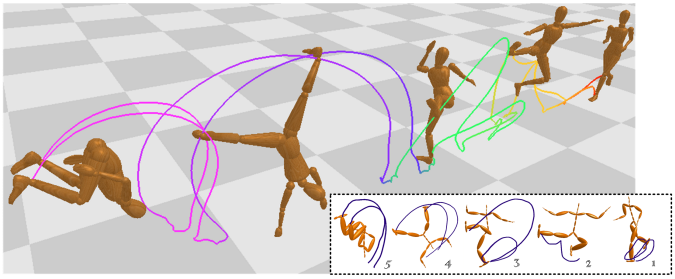

Fig. 19. A 3D animation generated from hand-drawn comic strip. Top: The 3D animation generated by concatenating the extracted motions. Bottom: the hand-drawn queries for motion retrieval.

need of time-alignment motion analysis and provides a compact encoding reducing the search time and repository space. The experimental results show that our approach can be used in real-time scenarios and is useful in various applications of motion creation. Certain limitations still exist in our approach. Not all 3D motion curves such as complex motions of hip-hop dancing can be easily or correctly described from a 2D drawing. In such case, a query of real motion clip is required. Moreover, our system requires users to select their desired poses and camera views from a set of predefined key-poses and their default camera viewpoints. Generally, a good starting pose and a good camera view are necessary by our system. If a neutral character pose or a poor camera viewpoint is selected, it may require sketching a more complex motion lines, and thus, significantly decrease the accuracy of retrieval. Fortunately, inferring 3D character poses from 2D sketched figures [29] or automatically extracting key poses and camera views [2] are possible. These approaches can be potentially applied to our system. In the near future, we plan to develop an approach to infer 3D character pose from 2D sketched figure, which can avoid the selection of character pose and camera view. Besides, the generated character pose can be used in the motion retrieval as suggested by [39]. Moreover, we also plan to expand the various types of sketch strokes which will allow more expressive details of the query.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments. We are also grateful to Yan-Bo Zeng for his coding at early stage of this project. The motion data used in this paper were partially collected from CMU Graphics Lab Motion Capture Database (http://mocap.cs.cmu.edu/). This work was supported in part by the National Science Council (contracts NSC-97-2628-E-006-125-MY3, NSC-98-2221-E-006-179 and NSC-99-2221-E-006-066-MY3), Taiwan.

References

[1] G. Baciu and B. K. C. Iu, "Motion retargeting in the presence of topological variations," *Journal of Visualization and Computer Animation*, vol. 17, no. 1, pp. 41–57, 2006.

[2] J. Assa, D. Cohen-Or, I.-C. Yeh, and T.-Y. Lee, "Motion overview of human actions," *ACM Transactions on Graphics (SIGGRAPH Asia 2008 issue)*, vol. 27, no. 5, pp. 115:1–115:10, 2008.

[3] K. Pullen and C. Bregler, "Motion capture assisted animation: texturing and synthesis," *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 501–508, 2002.

[4] Y. Li, Y. L. T. Wang, and H. yeung Shum, "Motion texture: a two-level statistical model for character motion synthesis," *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 465–472, 2002.

[5] L. Kovar, M. Gleicher, and F. Pighin, "Motion graphs," *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 473–482, 2002.

[6] J. Lee, J. K. Hodgins, J. Chai, N. S. Pollard, and P. S. A. Reitsma, "Interactive control of avatars animated with human motion data," *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 491–500, 2002.

[7] M. Thorne and D. Burke, "Motion doodles: an interface for sketching character motion," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 424–431, 2004.

[8] R. Heck and M. Gleicher, "Parametric motion graphs," *I3D '07: Proceedings of the 2007 symposium on Interactive 3D graphics and games*, pp. 129–136, 2007.

[9] Y.-Y. Tsai, W.-C. Lin, K.-B. Cheng, J. Lee, and T.-Y. Lee, "Real-time physics-based 3d biped character animation using an inverted pendulum model," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 2, pp. 325–337, 2010.

[10] C.-Y. Chiu, S.-P. Chao, M.-Y. Wu, S.-N. Yang, and H.-C. Lin, "Content-based retrieval for human motion data," *Journal of Visual Communication and Image Representation*, vol. 15, no. 3, pp. 446–466, 2004.

[11] K. Forbes and E. Fiume, "An efficient search algorithm for motion data using weighted pca," *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pp. 67–76, 2005.

[12] B. Demuth, T. Röder, M. Müller, and B. Eberhardt, "An information retrieval system for motion capture data," *Proc. 28th European Conference on Information Retrieval*, pp. 373–384, 2006.

[13] M. Müller, T. Röder, and M. Clausen, "Efficient content-based retrieval of motion capture data," *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 677–685, 2005.

[14] Z. Deng, Q. Gu, and Q. Li, "Perceptually consistent example-based human motion retrieval," *I3D '09: Proceedings of the 2009 symposium on Interactive 3D graphics and games*, pp. 191–198, 2009.

[15] W. Eisner, *Comics and Sequential Art*, 2008.

[16] L. Kovar and M. Gleicher, "Automated extraction and parameterization of motions in large data sets," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 559–568, 2004.

[17] E. Keogh, T. Palpanas, V. B. Zordan, D. Gunopulos, and M. Cardle, "Indexing large human-motion databases," *VLDB '04: Proceedings of the 30th international conference on Very large data bases*, pp. 780–791, 2004.

[18] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3d shape descriptors," *SGP '03: Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pp. 156–164, 2003.

[19] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, and D. Jacobs, "A search engine for 3d models," *ACM Transactions on Graphics*, vol. 22, no. 1, pp. 83–105, 2003.

[20] E. Keogh, "Exact indexing of dynamic time warping," *VLDB '02: Proceedings of the 28th international conference on Very Large Data Bases*, pp. 406–417, 2002.

[21] Y. Gao, L. Ma, Y. Chen, and J. Liu, "Content-based human motion retrieval with automatic transition," *Computer Graphics International*, pp. 360–371, 2006.

[22] Y. Gao, L. Ma, J. Liu, X. Wu, and Z. Chen, "An efficient algorithm for content-based human motion retrieval," *Lecture Notes in Computer Science*, pp. 970–979, 2006.

[23] Y. Lin, "Efficient human motion retrieval in large databases," *GRAPHITE '06: Proceedings of the 4th international conference on Computer graphics and interactive techniques in Australasia and Southeast Asia*, pp. 31–37, 2006.

[24] M. Müller and T. Röder, "Motion templates for automatic classification and retrieval of motion capture data," *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pp. 137–146, 2006.

[25] E. S. L. Ho and T. Komura, "Indexing and retrieving motions of characters in close contact," *IEEE Transactions on Computer Graphics and Visualization*, pp. 481–492, 2009.

[26] T. Igarashi, S. Matsuoka, and H. Tanaka, "Teddy: A sketching interface for 3d freeform design," *ACM SIGGRAPH '99*, pp. 409–416, 1999.

[27] O. Karpenko and J. Hughes, "Smoothsketch: 3d free-form shapes from complex sketches," *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 589–598, 2006.

[28] A. Nealen, T. Igarashi, O. Sorkine, and M. Alexa, "Fibermesh: designing freeform surfaces with 3d curves," *ACM Transactions on Graphics*, vol. 26, no. 3, p. 41, 2007.

[29] J. Davis, M. Agrawala, E. Chuang, Z. Popović, and D. Salesin, "A sketching interface for articulated figure animation," *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pp. 320–328, 2003.

[30] Q. L. Li, W. D. Geng, T. Yu, X. J. Shen, N. Lau, and G. Yu, "Motionmaster: authoring and choreographing kung-fu motions by sketch drawings," *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pp. 233–241, 2006.

[31] J. Barbič, A. Safonova, J.-Y. Pan, C. Faloutsos, J. K. Hodgins, and N. S. Pollard, "Segmenting motion capture data into distinct behaviors," *GI '04: Proceedings of Graphics Interface*, pp. 185–194, 2004.

[32] L. Shen and M. K. Chung, "Large-scale modeling of parametric surfaces using spherical harmonics," *3DPVT '06: Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission*, pp. 294–301, 2006.

[33] C. Brechbühler, G. Gerig, and O. Kübler, "Parametrization of closed surfaces for 3-d shape description," *Computer Vision and Image Understanding*, vol. 61, no. 2, pp. 154–170, 1995.

[34] J. de Souza's, "An illustrated review of how motion is represented in static instructional graphics," *STC's 55th Conference*, pp. 1–4, 2008.

[35] J.-Y. Kwon and I.-K. Lee, "Determination of camera parameters for character motions using motion area," *The Visual Computer*, vol. 24, no. 7, pp. 475–483, 2008.

[36] CMU 2003. Motion capture database., http://mocap.cs.cmu.edu/.

[37] F. Liu, Y. Zhuang, F. Wu, and Y. Pan, "3d motion retrieval with motion index tree," *Computer Vision and Image Understanding*, vol. 92, no. 2–3, pp. 265–284, 2003.

[38] L. Kovar and M. Gleicher, "Flexible automatic motion blending with registration curves," *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pp. 214–224, 2003.

[39] K. S. Sakamoto, Y. and T. Kaneko, "Motion map: Image-based retrieval and segmentation of motion data," *Symposium on Computer Animation (SCA2004)*, pp. 259–266, 2004.

Min-Wen Chao received the BS degree in Mathematics from the National Cheng-Kung University, Taiwan, in 2003 and the MS degree from the Department of Computer Science and Information Engineering, National Cheng Kuang University, Tainan, Taiwan, in 2005. She is currently working toward the PhD degree in the Department of Computer Science and Information Engineering, National Cheng-Kung University. Her research interests include computer graphics and Data hiding.

Chao-Hung Lin received his MS and PhD degree in computer engineering from National Cheng-Kung University, Taiwan in 1998 and 2004, respectively. He is currently an associate professor in the department of geomatics at National Cheng-Kung University in Tainan, Taiwan. He leads the Digital Geometry Laboratory, National Cheng-Kung University. His research interests include digital geometry processing, digital map generation, information visualization, and remote sensing. He is a member of IEEE and ACM.

Jackie Assa Jackie Assa received his BsC, MsC (Cum-Laude) and PhD in computer sciences from the Tel Aviv University in 1993, 1998 and 2010 respectively. He collaborated closely with researchers from various academic and industry research institutes on problems in the fields of image processing, 3D modeling, human animation, animation and video summarization and camera control.He now works as a consultant for leading companies, on problems in computer graphics, data visualization, video analysis and computer vision.

Tong-Yee Lee received the PhD degree in computer engineering from Washington State University, Pullman, in May 1995. He is currently a distinguished professor in the Department of Computer Science and Information Engineering, National Cheng-Kung University, Tainan, Taiwan, ROC. He leads the Computer Graphics Group, Visual System Laboratory, National Cheng-Kung University (http://graphics.csie.ncku.edu.tw/). His current research interests include computer graphics, nonphotorealistic rendering, medical visualization, virtual reality, and media resizing. He also serves on the editorial boards of the IEEE Transactions on Information Technology in Biomedicine, the Visual Computer and the Computers and Graphics Journal. He served as a member of the international program committees of several conferences including the IEEE Visualization, the Pacific Graphics, the IEEE Pacific Visualization Symposium, the IEEE Virtual Reality, the IEEE-EMBS International Conference on Information Technology and Applications in Biomedicine, and the International Conference on Artificial Reality and Telexistence. He is a senior member of the IEEE and the member of the ACM.