International Journal of Computational Vision and Biomechanics
Vol. 6 No. 1 (January-June, 2020)

Various Aspects of a Structured Motion Database for Efficient Human Motion Recognition

S. M. Ashik Eftakhar, Joo Kooi Tan, Hyoungseop Kim and Seiji Ishikawa

Department of Control Engineering, Kyushu Institute of Technology, Sensui-cho 1-1, Kitakyushu, Fukuoka 804-8550, Japan

Received: 23rd September 2019 Revised: 14th December 2019 Accepted: 25th December 2019

Human motion recognition is an emerging research field for the video-based applications capable of recognizing human motions or actions. The automaticity of such a system has vital importance for the implementation in practical scenarios. With the growth of enormous bulk motion dataset, it becomes an indispensable task to develop a motion database that can deal with large variability of motions efficiently. We have developed such a system based on the structured motion database concept. In order to gain a perspective on this issue, we have analyzed various aspects of the motion database with a view to establishing a standard recognition scheme. The conventional motion database structure is subjected to improvement by considering three aspects: directional organization, nearest neighbor searching problem resolution, and prior direction estimation. In order to investigate and analyze the effect of those aspects on motion recognition comprehensively, we have adopted two forms of motion representation, eigenspace-based motion compression, and aB-Tree structured database. Performance evaluation is performed by synthesized 3D human motions observed from eight camera viewpoints. The experimental results illustrate the suitability and compatibility of various aspects of the motion database.

Keywords: *Human Motion Recognition; Structured Motion database; Directional Organization; Nearest Neighbor Searching Problem; Clustering.*

1. INTRODUCTION

With the availability of huge number of captured videos consisting of various activities of a number of people, and their interactions, the importance of extracting semantic information has noticeably increased for different multimedia applications, for example, object detection, motion recognition, behavior understanding, behavior estimation, video conferencing, scene interpretation, and so on. Among these applications, the automatic representation and recognition of human movements in video sequences has its great influence on surveillance, supporting aged people in rehabilitation centers and clinics, sports analysis, guiding handicapped people by interpreting surrounding activities, and many other man-machine interaction systems. Moreover, the increased adaptability of current technology has made this challenging task to be solved with ease0and perfection. For example, with the development of digital libraries, the ability to automatically interpret video sequences will save much human effort in sorting and retrieving images or video sequences using content-based queries. Due to the underlying necessity of such kind of research, many researchers are rigorously involved in search of suitable techniques for interpreting and recognizing human motions. The literature on this problem of recognizing human motion from videos sequences is extensive [1]-[6]. Here, we focus on the methods addressing the specific problem of recognizing human motion from video-extracted image sequences without the use of markers, tracking devices, or special body suits. In general, such methods can be classified into two categories: multiple-view and single-view methods. Multiple-view methods address the motion recognition problem using image sequences obtained from multiple cameras placed at different spatial locations. The strength of these methods is their power to resolve ambiguous human motion patterns that may result from self-occlusion and viewpoint-driven appearance changes. However, multiple view approaches usually require the availability of synchronized camera systems and controlled camera environments. On the other hand, single-view methods rely only on the information provided by a single video camera ([4]-[10). Under the single-view assumption, human motion recognition becomes significantly more challenging, since the subject's detail cannot be exploited comprehensively. Therefore, feature extraction becomes extremely essential task to represent a motion. In this case, it is necessary to collect adequate useful features (e.g., contours, textures, skeletons, etc.) from the motion clips. The representative features extracted from human motions or activities is inherently both highly non-linear and high-dimensional. As a result, it is required to obtain a reduced dimensional spatial or spatio-temporal representation of the relevant (i.e., discriminative) motion features. The representation can be accomplished by making use of explicit measurements on the image to which a pre-determined model is fitted, or by

E-mail: ashik@ss10.cntl.kyutech.ac.jp
Phone & Fax: 81-93-8843183

accumulating prominent features from the large number of features (i.e., skeleton-based methods [11], part-segmentation-based methods [8] [9] [12], feature space reduction [13] [14]).

Standard techniques for the segmenting and extracting interest image portion include the ones based on Motion History Images (MHI) and its variants [4] [7] [15]-[17]. Motion history-based representations employing frame-wise preservation of motion history allow for simultaneous description of both the dynamics of the motion and the shape of objects. Alternatively, object's silhouette information alone can be used as an input for recognition systems. Wang and Suter [6] used silhouettes as the input to their recognition method. Besides, Elgammal and Lee [5] also used silhouettes without motion history. Furthermore, *a high accuracy and real-time* motion recognition approach was also proposed which considered multi-view motion representation and recognition in [18]. In this approach, the successive motion frames are transformed into a single eXclusive-OR (XOR) image for the task of storage and recognition. However, after generating the motion model or template, the maximum similarity between an unknown motion and each of the pre-learned motion is determined. Several distance measures are taken into account in terms of Mahalanobis distance [7], Hausdorff distance [4] [6] [15], Dynamic Time Warping and its variants [2] [19], and so on. However, with the enormous growth of motion archives, the demand for efficiently finding similar motions within a large motion database has been significantly increased. In most of the cases, motions are represented as multi-dimensional spatio-temporal data which are to be stored within the database. However, a motion recognition system was analyzed using B-Tree with exclusive-OR images in [18] that showed satisfactory results with a medium-sized motion dataset. Moreover, there are many variations for motion representations based on different applications. Therefore, considering the variability, there are many non-linear databases available for storing the motions, as a whole. Some examples of this kind of databases are: AVL Trees [20], B-Trees [21], R-Trees [22], R+ -Tree [23], PK-Tree [24], etc. Most of these structures deal with the storage of the multi-dimensional data within the database.

In this paper, we propose a 3-aspect structured database-based approach for efficient human motion recognition. Since the structured organization is a new invention for motion recognition, there exist several factors which influence the system's performance. Therefore, we analyze those aspects of the structured database for refining the efficiency of the overall system. We incorporate the direction-oriented motion capture of the subjects performing different motions or actions to make the system insensitive to orientation. Two different motion templates are adopted to analyze the performance by taking into account different aspects in terms of accuracy and speed. This paper is organized as follows: Section 2 describes the core operations for the development of the structured motion database. Section 3 briefly describes various aspects of the structured database, and analyzes the suitability of those aspects in recognition. Section 4 illustrates the experimental results and the performance evaluation with the modified structured database. Finally, Section 5 summarizes the compatibility and applicability of the structured motion database, including some future works.

2. DEVELOPMENT OF STRUCTURED MOTION DATABASE

A motion recognition system based on the structured motion database concept comprises several phases. The phases are functionally ordered as, *motion representation*, *motion compression within the feature space*, *generation of motion extracted data*, and *construction of structured motion database with the training motions*. The detail operations of these phases are described in the following.

2.1. Motion Representation

Motion representation, in this context, demonstrates the change in brightness values within the consecutive motion frames representing a motion. This sort of representation works on the successive motion frames extracted from each motion or action, captured mainly in two-dimensional form. It is usually a pixel-level transformation of human motion. Generally, a motion capture system captures the motion within itself and manipulates it to represent in a compact format to ease the high-level processing. According to this, there are several individual features or properties that can be extracted and tracked in the sequence of frames within a motion. We have adopted MHI (*Motion History Image* [7]) and XOR image (*Exclusive-OR image* [18]) representations for the current recognition system. We refer to these representations as either *motion image* or *feature image* (See Fig. 1).

2.1.1. Motion History Image

Motion History Image (MHI) is a frame-based temporal template for human motions. As the name implies, this form of motion representation keeps track of the motion history, i.e. representing *how* the motion is moving along a certain period of time. The more recently moving pixels are brighter than past moving pixels on the feature image. Let $H_\tau(x, y, t)$ be the pixel intensity function of the temporal history of motion at a particular point (x, y). The function is represented in a simple way in (1).

$$H_\tau(x,y,t) = \begin{cases} \tau & if \quad D(x,y,t) = 1 \\ \max(0, H(x,y,t-1)-1) & \text{Otherwise} \end{cases} \quad (1)$$

Here, $D(x, y, t)$ is a difference binary image constructed by successive frame difference. The function $H_\tau(x, y, t)$ returns a scalar value. According to the function, in the generated image the more recently moving pixels have higher values than the past moving pixels (See Fig. 1(b)). In (1), τ is taken as the temporal extent which is critical to define. But for the flexibility of the value of τ, it can be taken as the maximum gray level pixel value (255) or the maximum number of frames comprising the motion [25].

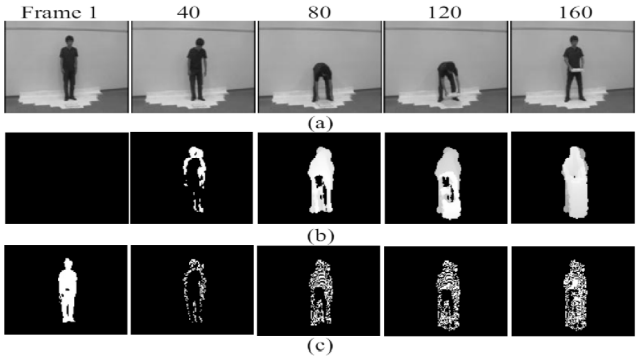

Figure 1: Generation of frame-by-frame motion representations in 'Carry-up' motion sequence (a) Original motion frame, (b) Corresponding MHIs, (c) Corresponding XOR images

2.1.2. Exclusive-OR Image

The exclusive-OR (XOR) operations are performed on each pixel between consecutive frames, and the cumulative XORed form of the frames is referred to as XOR image. Binarized form of each frame is obtained by using a fixed thresholding (e.g., *threshold value* = 20) function. Let *m*, *h*, *c* representing motions, persons, camera-directions, and *f*, *U* denoting *binarized motion frame* and *XOR image*, then (2) presents the complete XOR operation on the motion frames (See Fig. 1(c)).

$$\begin{aligned} &U_c^{m,h}(2) = f_c^{m,h}(1) \ \text{XOR} \ f_c^{m,h}(2), \\ &U_c^{m,h}(r) = U_c^{m,h}(r-1) \ \text{XOR} \ f_c^{m,h}(r), \ \text{where,} \ \ r = 3,4,\ldots,R \\ &U_c^{m,h} \equiv U_c^{m,h}(R). \end{aligned} \tag{2}$$

Here, the motion image at frame *r* for motion *m* of person *h* obtained from camera *c* is denoted by $U_c^{m,h}$. Hence, for *M* motions of *H* persons each having *R* frames from *C* camera directions generates *MHC* XOR images representing one feature image corresponding to *r* frames. This method is capable of extracting the moving portion in the scene using the logical manipulation. This is simple, effective, and fast generating motion representation method.

2.2. Motion Compression

A compressed feature space for the high-dimensional spatial representation of motions is characterized by a number of vectors with higher significance, where each vector is an effective form of characterization and similarity determination among a number of data. An eigenspace representation serves this purpose by projecting the motion data onto a high-dimensional feature space, and the computing the vectors with high variance. It is a modified form of Karhunen-Loeve Transform that is commonly used to derive relationship among different random variables. In practice, a large set of training motions is needed to be projected onto the eigenspace by finding featured eigenvectors. For each motion image (MHI or XOR image) $I_m(m = 1,2,\ldots,M)$, an image matrix is defined and the brightness is normalized to minimize the its effect. Thus a normalized image matrix $\mathbf{x}_1, \mathbf{x}_2,\ldots,\mathbf{x}_M$ is obtained. The average image $\mathbf{c}$ is subtracted from it to constitute a standard image matrix $\mathbf{X}$ in (3).

$$\mathbf{X} \triangleq (\mathbf{x}_1 - \mathbf{c}, \mathbf{x}_2 - \mathbf{c}, \ldots, \mathbf{x}_M - \mathbf{c}) \tag{3}$$

The image matrix $\mathbf{X}$ is $N \times M$, where *M* is the total number of motion images, and *N* is the total number of pixels in each image. To compute eigenvectors of the motion image set, a covariance matrix Q is defined as:

$$\mathbf{Q} \triangleq \mathbf{X}\mathbf{X}^{\mathrm{T}} \tag{4}$$

The eigenvalues of the covariance matrix is real and nonnegative. Among *N* eigenvectors, *k* most prominent eigenvectors ($\mathbf{e}_1, \mathbf{e}_2,\ldots, \mathbf{e}_k$) are chosen to create an eigenspace *ES* consisting of the training motions [26]. These *k* eigenvectors constitute the eigenspace which is an approximation to the complete eigenspace with *N* dimensions. The eigenspace is constructed by projecting the corresponding motions (i.e., motion images) onto the eigenspace using (5). This represents multi-dimensional points within the hyperspace which will be used for storage and similarity measurement in latter section.

$$\mathbf{g}_m = (\mathbf{e}_1, \mathbf{e}_2,\ldots, \mathbf{e}_k)^{\mathrm{T}} (\mathbf{x}_m - \mathbf{c}) \tag{5}$$

2.3. Generation of Motion Extracted Data

In order to extract suitable form of the motion data, the multi-dimensional information is to be converted into a form that fits with the motion database organization and that can be easily stored and retrieved. This compact form is termed as *index* and the process of generating an index is called *indexing*. As the size of the database is increasing day-by-day for the development of a robust system, the problem of indexing of motion data has attracted great interest in the database community. There have been extensive researches in the past years involving the task of indexing [27-31]. In this context, the dimension of an eigenspace is taken as an important cue for indexing by uniform space partitioning. Each eigen-axis $\mathbf{e}_k$ ($k = 1,2,\ldots, K$) is divided into *S* sections leading to S^k hypercubes with edge length *L* along each eigen-axis. Each hypercube is referred to as *bin* in spatial term and *index* in numeric term. Each bin or index is assigned a

digit from 0 to *S*-1 along each eigen-axis; each motion is assigned a *k*-digit number with each digit ranging between 0 and *S*-1 correspondingly. A bin encompasses one or more motion points within the space.

2.4. Construction of a Structured Motion Database

A motion database, solely, relies upon the organization or the storage within the computer memory. Unlike the typical database concepts, the most common database organization is linear – arranges data in the order of its input. There is no predefined rule for the organization. Therefore, at the time of query, it becomes exhaustive searching for the content within the database. In order to overcome this limitation of the query, many researchers have been involved in the development of a suitable motion database that is capable of successful and quick retrieval. However, it is obvious that due to the increased number of registration within the database, the maintenance of the database organization is becoming very tough to handle. A multi-way tree branching structure in the form of B-tree [21] database structure, is adopted in our research as the structured motion database. B-Tree database is advantageous for storing multi-dimensional data with average storage utilization, and effective retrieval.

Within the structured motion database approach, we construct B-Tree database with the indexes. These indexes are inserted within the tree structure using the B-Tree insertion algorithm. This is quite simple, and the database is also easily updatable. The detail of the index insertion and update algorithm is described in [26]. Thus, in the training phase, the motions are trained by generating the motion images representing the motion features (MHI or XOR image), constructing eigenspaces with the motions, indexing of the motions, and finally storing each motion corresponded by an index into the structured database. These phases represent the core operations for building the structured motion database.

3. VARIOUS ASPECTS OF STRUCTURED MOTION DATABASE

In order to adopt the recently introduced structured motion database for human motion recognition, various aspects of performance improvement is investigated and analyzed. In this paper, we shall explore three aspects of structured database: *directional organization*, *resolution of the nearest neighbor searching problem*, and *prior estimation of directions*. These aspects are briefly described in the following paragraphs.

3.1. Directional Organization of the Structured Database

For the motions with several orientations, these motions can be grouped into several motion sets based on the orientation. In [32], the directional scheme is adopted as an improved direction-oriented human motion recognition approach. The steps for constructing the directionally organized structured database are as follows:

(a) Capture the training motions having c (c >1; integer) orientations by maintaining motion synchronization.

(b) Create motion sets M_i (i = 1,2,…,c) based on the orientation.

(c) Construct eigenspaces ES_i corresponding to each motion set M_i using the scheme described in Section 2.

(d) Construct the structured B-tree sub-database BSB_i corresponding to each eigenspace ES_i taking the division parameter S.

(e) Combine all the sub-databases to develop directionally organized database.

(f) Construct a *global eigenspace* with all the directional training motions.

In recognition phase, an unknown motion, represented as a sequence of image frames, is processed for generating the motion representation. An MHI or XOR image is obtained from the motion, and projected onto each directional eigenspace. An index is generated corresponding to each directional eigenspace. For each camera orientation, the equal number of candidate motions is obtained by searching the corresponding B-Tree sub-database using the nearest neighboring point searching strategy [32]. These candidate motions are projected onto the global eigenspace as $\mathbf{g}_{m_r}$ (r = 1,2,.., D), where D is the number of camera orientations. The unknown motion is also projected as $\mathbf{g}_m$ within the global eigenspace. The most similar motion is obtained using *Euclidian distance function* in (6).

$$d_m = \min_r \left\| \mathbf{g}_{m_r} - \mathbf{g}_m \right\| \tag{6}$$

3.2. Resolution of the Nearest Neighbor Searching Problem

With the adoption of B-Tree structured database, the searching of the most similar motion fails when one motion is similar to several motions within the eigenspace. This misrecognition problem refers to *Nearest Neighbor Searching Problem*, or sometimes *boundary problem*. There are two cases for the occurrence of this problem.

(i) Motion points lying on the edge of a bin within the eigenspace imply the inaccurate selection of the bin, since other point in another bin may be the nearest one.

(ii) If the index does not seem to reside within the database, it is necessary to find the least different

index within it. But because of the high-dimensionality of the feature space, no standard algorithm exists (except linear searching) that can do it accurately. The approximation algorithm proposed in [32] using the decimal value corresponding to the *S*-nary number to match the nearest bin may lead to misclassification of motions.

In order to solve this problem, two sets of query space are maintained by shifting the space division, i.e., structurization, to a certain scale [33]. Thus two sets of query spaces are constructed, namely *original* (prior developed) query space set and *shifted* (left- or right-shifted) query space set. Two query space sets are maintained over the feature space parallely. Corresponding to two sets of query spaces, two parallel motion databases are developed which constitute the whole database system. The resolution to the boundary problem lies in the selection of an appropriate query space between the two. The scale of shifting can be chosen arbitrarily. Searching within the directional databases is accomplished using the recognition scheme mentioned in Section 3.1.

3.3. Prior Estimation of Directions

We introduce a direction-oriented motion recognition approach that primarily estimates the directional information, and then use this estimated information in similarity searching. This reduces the processing time of the system by excluding unnecessary searching for the most similar motions. The direction-wise motions are clustered within the feature space to make the direction estimation easier.

In order to estimate the possible orientations of an unregistered motion, the global eigenspace is exploited. For this purpose, the projected training motion points within the global eigenspace are clustered based on the orientation from which these are viewed. For *D* number of orientations, *D* clusters are constructed correspondingly within the space. We enclose these direction-wise motion points by hyperspheres within the space as *clusters*. Thus we obtain *D* hyperspheres in the space, either overlapping or non-overlapping. When an unknown motion comes, its position within the global eigenspace is computed and the clusters it belongs to are also computed and selected for possible estimation of directions. Afterwards, only the prior selected directional eigenspaces are searched. Thus we can reduce the unnecessary searching cost for motion retrieval. The steps required for the task of recognition are stated below.

- The possible directions of the feature image are extracted from the clusters within the global eigenspace.
- It is projected onto the selected directional eigenspaces.
- An index is generated for each of the selected directional eigenspace.
- The indexes are searched within the corresponding sub-databases to find the closest one within the registered motion points by searching the B-Trees. Among *D* number of directions, a subset *d* sub-databases corresponding to *d* directional eigenspaces are searched, and we obtain *d* number of candidate motions.
- The candidate motions are projected onto the global eigenspace and the most similar motion is obtained from the global eigenspace using (6).

4. EXPERIMENTAL RESULTS

4.1. Experimental Setup

The experiments are performed on an *Avatar dataset* with different synthesized human avatars performing ten different types of motions, namely *bend* (bending down), *carry* (carrying a box), *jump* (hopping in a place), *pjump* (jumping with two hands up and landing down), *pickup* (picking up something from the ground), *sitdown* (sitting down on a chair), *standup* (standing up from a chair), *stomachache* (touching stomach with pain and crouch), *walk* (walking motion), and *wave2* (waving two hands up in the air). The variation in motions is realized by subject's height and shape, speed of motion, and field of view. The scene is assumed to be backgroundless. Eight uncalibrated cameras are placed surrounding the avatar at 45 degrees apart, having 0-, 45-, 90-, 135-, 180-, 225-, 270- and 315-degree camera orientations. Human surface is perpendicular to the viewing plane, i.e., parallel to the camera direction at 0-degree camera view and the viewing angles are assumed to be in clockwise direction. Figure 2 illustrates different motions, and the corresponding MHIs and XOR images. The motion dataset consists of 800 motion videos divided into eight orientations. Among those, 560 motion data are considered as training set, and the rest as testing set. Thus each of the eight directional sub-databases consists of 70 motions corresponding to each orientation, whereas the global eigenspace consists of 560 motions corresponding to all the orientations. Likewise, the test set consists of 30 motions each (10 motions performed by 3 actors) for each orientation. Different aspects are adopted for experimentation to evaluate the performance of the system in terms of recognition accuracy and time requirement. The system is tested for MHI and XOR images to illustrate the performance regardless of the motion representation adopted.

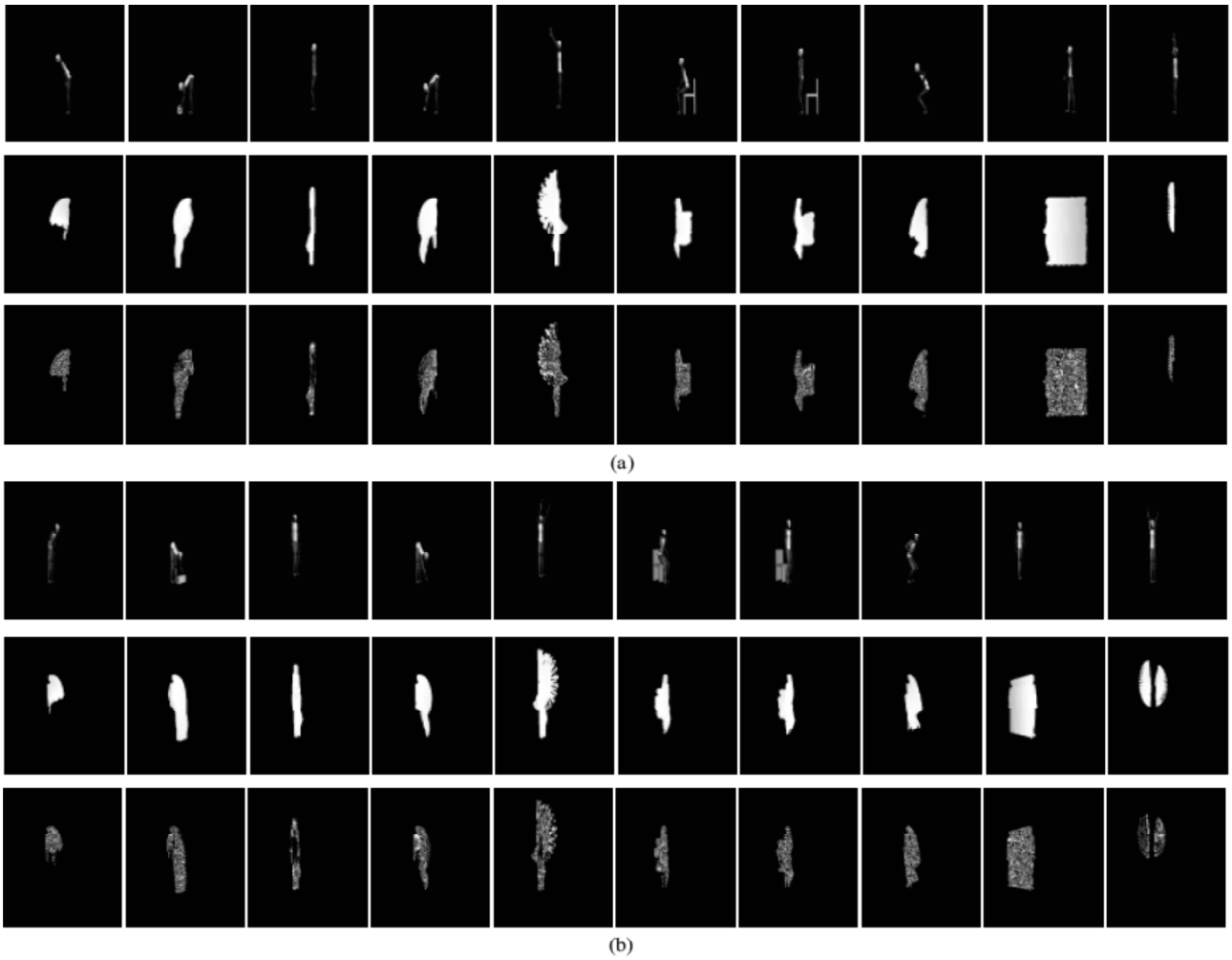

Figure 2: A single motion frame and corresponding MHIs and XOR images (in consecutive rows) for each of the ten motions from different camera orientations: (a) 0 degree, (b) 135 degree

4.2. Definition

4.2.1. Bin Length

A bin is defined as the hypercube within each directional eigenspace by dividing each eigen-axis. By transforming the extent of each axis to *unit* with *S* divisions along the axis, the length of each edge of a bin is defined as:

$$\text{Bin Length } (L) = \frac{\text{Extent of an eigen-axis}}{\text{Total number of divisions along the axis}} = \frac{1}{S}$$

The experimental results are computed by varying the division parameter *S* from 1 through 10 to note the performance variation, as well as, to choose the best one among them. We shall refer bin length as *L* in latter part of this paper.

4.2.2. Recall

Recall is defined as the percentage of successfully recognized motions with respect to the *ground truth* motions. It can be defined as:

$$\text{Recall } (R) = \frac{\text{Number of recognized motions}}{\text{Ground Truth}} \times 100$$

4.2.3. False Positive Rate (FPR)

False Positive Rate (FPR) is defined as the percentage of miss-recognized motions among the total number of motions which are recognized, either correctly or incorrectly. It can be defined as:

$$\text{False Positive Rate } (FPR) = \frac{\text{Number of miss-recognized motions}}{\text{Number of recognized motions} + \text{Number of miss-recognized motions}} \times 100$$

4.2.4. Precision

Precision is defined as the percentage of recognized motions among the total number of motions which are recognized, either correctly or incorrectly. It can be defined as:

$$\text{Precision } (P) = \frac{\text{Number of recognized motions}}{\text{Number of recognized motions} + \text{Number of miss-recognized motions}} \times 100$$

4.3. Analysis

In this section, we demonstrate the performance of different aspects of the structured motion database with comprehensive experimentation and analysis. Each aspect shows variation in performance from one another for the experimental data set. As we alluded various aspects of motion database in Section 3, we illustrate the performance comparison in terms of *recognition rate* and *searching time* in Fig. 3 and Fig. 4, respectively. We notice the significant increase in performance from *non-directional* to *problem resolution scheme* for both MHI and XOR image representations. The eigenspace with bin length 1 is referred to as *non-structured form* of motion database that employs exhaustive searching within the database for finding the candidate motions. However, having employed the structured motion database with MHI templates, the average recognition rate for non-directional, directional, problem resolution (scale of shifting is $L/2$), prior direction estimation and problem resolution with prior estimation schemes are 74%, 86%, 92%, 85% and 91%, respectively. Similarly, with XOR image, the average recognition rate for non-directional, directional, problem resolution (scale of shifting is $L/2$), prior direction estimation and problem resolution with prior estimation schemes are 64%, 77%, 85%, 77% and 85%, respectively. We notice significant improvement in recognition rates from non-directional to problem resolution. From Fig. 4, we notice the average searching time requirements for the above five schemes, in order, are 14.9 ms, 12 ms, 25.4 ms, 10.2 ms and 20.9 ms with MHI, and 13.8 ms, 14.4 ms, 32.5 ms, 13.6 ms and 31.7 ms with XOR, respectively. Analyzing the searching time, we find satisfactory improvement for the prior estimation of directions, since it diminishes unnecessary searching load. Experimentally, the reduction of searching cost with MHI in terms of the number of eigenspaces was found to be 266 eigenspaces corresponding to 240 test motions which remain unaccounted due to direction estimation. On an average, more than one eigenspace per motion is eliminated for redundancy at the time of searching for the candidate motions. In the case of XOR image, due to the scattered nature of the motion points within the space, the dimensionality of the feature space becomes high, along with the space being widely dispersed, leading to the less reduction of the searching spaces. Therefore, there is only slight reduction in time requirement with the prior estimation.

We have also tabulated the performance evaluation of various schemes, namely, *non-structured*, *basic structured* (or non-directional), *directional*, *prior direction estimation*, *problem resolution*, and *problem resolution with prior estimation*, with the maximum recognition rates achieved and the corresponding time requirement (See Table I). We find that 95% and 93%

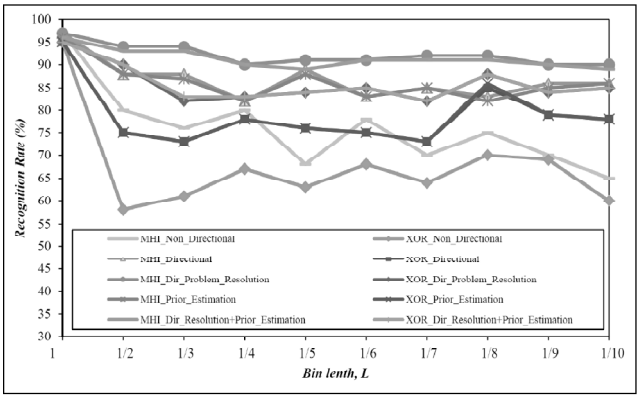

Figure 3: Illustration of recognition rate for various schemes with XOR and MHI

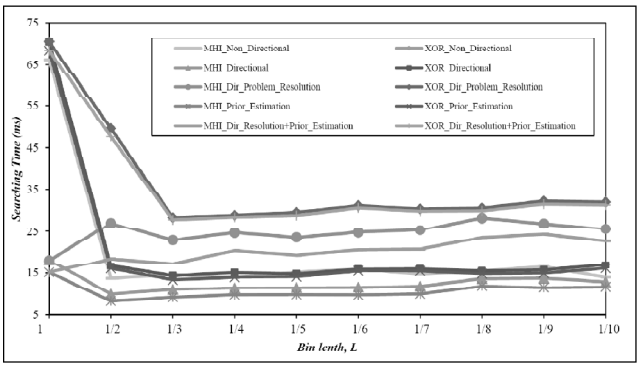

Figure 4: Illustration of time requirement for various schemes with XOR and MHI

recognition rate is obtained for the problem resolution scheme at the scale of shifting $2L$ (L=1/2) with MHI and XOR image, respectively. However, for the scheme having prior direction estimation and problem resolution together, though the recognition rate (94%) is slightly less than the scheme with problem resolution only, it shows shorter searching time which is much acceptable in this case. Therefore, the scheme with problem resolution with prior estimation of directions presents the best performance for our experimentation utilizing the three aspects of the structured database altogether.

However, we have also calculated the motion-wise *recall*, *FPR* and *precision* at scale of shifting $2L$ (L=1/2); we found that *standup* and *pickup* motions shows higher FPR and lower precision with MHI, and *stomachache* and *pickup* shows higher FPR and lower precision with XOR image. These measures are tabulated in Table II.

Moreover, we have also analyzed the effect of shifting parameter (i.e., scale of shifting) on the recognition rate by varying the parameter at $L/4$, $L/2$, L and $2L$. We found that the recognition rate is the maximum for both the MHI and XOR image at the scale of shifting $2L$. It is 95% for MHI and 93% for XOR image. Thus we can assume the shifting parameter empirically for a specific dataset. This effect is illustrated

Table I
Performance Evaluation for Various Recognition Scheme

Scheme	*MHI*		*XOR*	
	Recognition Rate (%)	*Searching Time (millisecond)*	*Recognition Rate (%)*	*Searching Time (millisecond)*
Non-structured	96	66.53	95	336.84
Basic structured	80	14.9	70	13.8
Directional	89	12	85	14.4
Prior estimation of directions	88	10.2	85	13.64
Directional with problem resolution	95	33.24	93	59.64
Directional with prior direction estimation and problem resolution	**94**	**22.85**	**93**	**57.28**

Table II
Motion-wise Performance Evaluation

Motion	*MHI*			*XOR*		
	Recall (%)	*Precision (%)*	*False Positive Rate (%)*	*Recall (%)*	*Precision (%)*	*False Positive Rate (%)*
bend	87.5	100	0	87.5	95.5	4.5
carry	83.33	95.2	4.8	83.33	87	13
jump	91.67	88	12	100	96	4
pickup	95.83	85.2	14.8	79.17	86.4	13.6
pjump	100	100	0	100	100	0
sitdown	87.5	100	0	83.33	87	13
standup	91.67	84.6	15.4	100	100	0
stomachache	100	100	0	100	82.8	17.2
walk	100	100	0	100	100	0
wave2	100	88.9	11.1	100	100	0

by bar-graph in Fig. 5. All the experiments were conducted on a CORE2DUO 2.93 GHz Processor 4GB RAM-computer.

5. CONCLUSIONS AND FUTURE WORK

We have presented three essential aspects of the recently developed structured motion database. In recent times, a bulk of motion/action datasets is available for action or behavior understanding and analysis. Due to the large amount of data, the structurization becomes an indispensable task. The structurization demands the database to be efficient with respect to recognition rate,

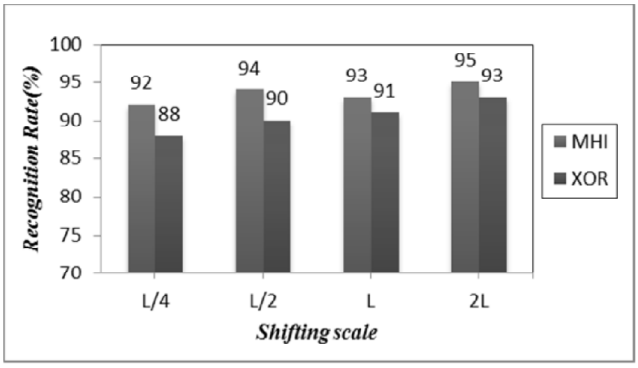

Figure 5: Effect of *scale of shifting* on recognition rate

time and space. We found that the aforesaid aspects jointly construct a database for the motion recognition that is suitable according to the above three criteria, i.e., recognition rate, time and space. Thus we have introduced a structured motion database based novel recognition technique for identifying and interpreting human motions and actions adopting the various aspects of the database. We propose a scheme where both the top-down and bottom-up strategies are followed one-after-another. We have estimated the directions for an unregistered motion by top-down manner, and we obtained the possible orientations of the motion that guides the searching algorithm. On the other hand, the motion is searched within the directional eigenspaces and candidate motions are obtained. These candidate motions are further projected onto the global eigenspace to confirm the category of the motion. This is accomplished by bottom-up manner. In the earlier approaches [32][33], motion recognition is accomplished in bottom-up manner only, whereas the newly introduced hybrid manner of problem solution proposed in this paper shortens the searching time by reducing potentially unnecessary searching cost within the feature spaces. The goodness of the system lies in the resolution of the similarity searching problem and the reduction of the searching complexity that makes it improved and non-redundant. The scheme of prior direction estimation with nearest neighbor search problem resolution has significantly reduced the searching time with high recognition rate - this proves its effectiveness in performance. We obtain 94% recognition rate at 22.85 ms with MHI, and 93% recognition rate at 57.28 ms with XOR image. Though both MHI and XOR image-based system shows similar recognition rate, the searching time varies due to the high dimensionality of the feature space with XOR image. However, we notice a slight decrease of less than one percent in the recognition rate with this scheme; this is due to the inclusion of 90 percent of the total number of direction-wise motion points for constructing each direction-wise cluster. Moreover, with the increase in registration within the database, the searching time tends to rise; but selective searching may lead to time-efficient human motion recognition.

Therefore, the online or real-time applications of the human motion recognition systems have much potential to be practically applicable in various real-life environments.

Although we have achieved satisfactory performance from our proposed system, there are, of course, some limitations in the current system. If there are more than one person within in the viewing area or one person partially occludes another, the system will surely fail to recognize. An alternative solution [34] might be incorporated to cope with the existence of multiple persons in a scene, whereas multi-view method can cope with the occlusion problem. So, it would be worthwhile to further investigate the system for more sophisticated motion recognition applications and with the multi-view method in order to improve the robustness of the system. Moreover, it would be more efficient to develop the system with real-life complex motion datasets. However, other motion representations [16] could also be adopted to test the system's performance. In practice, there is a great demand for an intelligent robot capable of human motion or action recognition with instant decision making in any security system, or in clinics or rehabilitation centers, or in surveillance system for tracking suspicious matter, etc. With the use of networks, it will become more reliable and robust.

ACKNOWLEDGMENT

This work was supported by (JSPS) KAKENHI (22510177) Grant-in-aid for Scientific Research (C), which is greatly acknowledged.

REFERENCES

[1] J. K. Aggarwal, Q. Cai, "Human motion analysis: a review", *J. Computer Vision Image Understanding*, **73**(3), 428-440, 1999.

[2] J. Blackburn, E. Ribeiro, "Human motion recognition using isomap and dynamic time warping", Human Motion Understanding, Modeling, Capture and Animation: LNCS Springer-Berlin: Heidelberg, pp. 285-298, 2007.

[3] D. M. Gavrila, "The visual analysis of human movement: a survey", *J. Computer Vision Image Understanding*, **73**(1), 82–98, 1999.

[4] O. Masoud, N. Papanikolopoulos, "A method for human action recognition", *J. Image Vision Computing*, **21**(8), 729–743, 2003.

[5] A. Elgammal, C. S. Lee, "Inferring 3D body pose from silhouettes using activity manifold learning", *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, **2**, 681–688, 2004.

[6] L. Wang, D. Suter, "Analyzing human movements from silhouettes using manifold learning", *Proc. of IEEE International Conference on Video and Signal Based Surveillance*, p. 7, 2006.

[7] A. F. Bobick, J. W. Davis, "The recognition of human movement using temporal templates", *IEEE Trans. Pattern Analysis and Machine Intelligence*, **23**(3), 257-267, 2001.

[8] C. Fanti, L. Zelnik-Manor, P. Perona, "Hybrid models for human motion recognition", *Proc. of Conference on Computer Vision and Pattern Recognition*. pp. 1166–1173, 2005.

[9] C. Heisele, B. Woehler, "Motion-based recognition of pedestrians", *Proc. of IEEE International Conference on Pattern Recognition*, pp. 1325-1330, 1998.

[10] S. X. Ju, M. J. Black, Y. Yacoob, "Cardboard people: a parameterized model of articulated motion", *Proc. of International Conference on Automatic Face and Gesture Recognition*, pp. 38-44, 1996.

[11] D. Gavrila, L. Davis, "3D model-based tracking of humans in action: a multi-view approach", *Proc. of Conference on Computer Vision and Pattern Recognition*, pp. 73-80, 1996.

[12] C. Bregler, J. Malik, "Learning appearance based models: Mixtures of second moment experts", Advances Neural Information Processing Systems 9, MIT Press, pp. 845-851, 1997.

[13] H. Murase, S. K. Nayar, "Visual learning and recognition of 3-D objects from appearance", *Computer Vision*, **14**(1), 5-24, 1995.

[14] H. Murase, R. Sakai, "Moving object recognition in eigenspace representation: gait analysis and lip reading", *Pattern Recognition Letters*, **17**, 155-162, 1996.

[15] M. Blank, L. Gorelick, E. Shechtman, M. Irani, R. Basri, "Actions as space-time shapes", *Proc. of IEEE International Conference on Computer Vision*, pp. 1395–1402, 2005.

[16] M. A. R. Ahad, T. Ogata, J. K. Tan, H. S. Kim, S. Ishikawa, "Motion History Image: Its Variants and Applications", *Machine Vision and Applications*, pp. 1-27, 2010.

[17] T. Ogata, J. K. Tan, S. Ishikawa, "High-speed Human Motion Recognition based on a motion history image and an eigenspace", *IEICE Transaction on Information and Systems*, **89**(1), 281-289, 2006.

[18] J. K. Tan, S. Ishikawa, "High accuracy and real time recognition of human activities", *Proc. of Annual Conference of IEEE Industrial Electronics Society*, pp. 2377-2382, 2007.

[19] D. J. Berndt, J. Clifford, "Finding patterns in time series: a dynamic programming approach", Advances in Knowledge Discovery and Data Mining, American Association for Artificial Intelligence, Menlo Park, CA, USA, pp. 229–248, 1996.

[20] C. Ellis, "Concurrent search and insertion in AVL trees", *IEEE Transaction on Computers*, Vol. C-29, No. 9, 811-817, 1980.

[21] R. Bayer, E. McCreight, "Organization and maintenance of large ordered indexes", *Acta Informatica*, **1**(3), 173-189, 1972.

[22] A. Guttman, "R-Trees: a dynamic structure for spatial searching", *Proc. of Annual Meeting, SIGMOD Conference*, pp. 47-57, 1984.

[23] T. Sellis, N. Roussopoulos, C. Faloutsos, "The R+ -tree: a dynamic index for multi-dimensional objects", Proc. of Very Large Data Bases (VLDB) Conference, pp. 507-518, 1987.

[24] W. Wang, J. Yang, R. Muntz, "PK-tree: a dynamic spatial index structure for large data sets", Technical Report no. 970039, Computer Science Department, University of California, Los Angeles, 1997.

[25] M. A. R. Ahad, T. Ogata, J. K. Tan, H. S. Kim, S. Ishikawa, "A Complex Motion Recognition Technique Employing Directional Motion Templates", *International Journal of Innovative Computing, Information and Control*, **4**(8), 1943-1954, 2008.

[26] S. M. A. Eftakhar, J. K. Tan, H. Kim, S. Ishikawa, "Human motion recognition employing large motion-database structure", *International Journal of Advanced Computer Engineering*, **2**(1), 17-23, 2009.

[27] E. Keogh, T. Palpanas, V. B. Zordan, D. Gunopulos, M. Cardle, "Indexing large human-motion databases", Proc. of Very Large Data Bases (VLDB) Conference, pp. 780-791, 2004.

[28] C. Li, B. Prabhakaran, "Indexing of motion capture data for efficient and fast similarity search", *Journal of Computers*, **1**(3), 35-42, 2006.

[29] M. A. Nascimento, E. Tousidou, V. Chitkara, Y. Manolopoulos, "Image Indexing and Retrieval using Signature Trees", *Data and Knowledge Engineering*, **43**(1), 57-77, 2002.

[30] F. Liu, Y. Zhang, F. Wu, Y. Pan, "3d motion retrieval with motion index tree", *Computer Vision and Image Understanding*, **92**(2-3), 265-284, 2003.

[31] P. Punitha, N. Onkarappa, D. S. Guru, "Indexing of document images based on triangular spatial relationships", *Proc. of International Conference on Computing: Theory and Applications*, pp. 533-537, 2007.

[32] S. M. A. Eftakhar, J. K. Tan, H. Kim, S. Ishikawa, "An effective directional motion database organization for human motion recognition", *International Journal of Innovative Computing, Information and Control*, in press.

[33] S. M. A. Eftakhar, J. K. Tan, H. Kim, S. Ishikawa, "Improvement of a structured motion database for high accuracy human motion recognition", *International Journal of Biomedical Soft Computing and Human Sciences*, in press.

[34] S. M. A. Eftakhar, J. K. Tan, H. Kim, S. Ishikawa, "Multiple Persons' Action Recognition by Fast Human Detection", *Proc. of SICE Annual Conference*, pp. 1639-1644, 2011.