ELSEVIER

Image and Vision Computing 18 (2000) 1067–1083

image AND vision COMPUTING

www.elsevier.com/locate/imavis

The model-based human body motion analysis system

I.-Cheng Chang, Chung-Lin Huang*

Electrical Engineering Department, National Tsing Hua University, Hsin Chu, Taiwan 30043, ROC

Received 5 January 1998; revised 3 April 2000; accepted 27 April 2000

Abstract

In this paper, we propose a model-based method to analyze the human walking motion. This system consists of three phases: the preprocessing phase, the model construction phase, and the motion analysis phase. In the experimental results, we show that our system not only analyzes the motion characteristics of the human body, but also recognizes the motion type of the input image sequences. Finally, the synthesized motion sequences are illustrated for verification. The major contributions of this research are: (1) developing a skeleton-based method to analyze the human motion; (2) using Hidden Markov Model (HMM) and posture patterns to describe the motion type. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Body signature; Posture graph; Posture transition path; Motion characteristics curves

1. Introduction

Human body motion analysis has been an interesting research for its various applications, such as athletic performance evaluation, medical diagnostics, virtual reality, and human–machine interface. In general, three aspects of research directions are considered in the analysis of human body motion: tracking and estimating motion parameters, analyzing of the human body structure, and recognizing of motion activities.

To track and estimate the motion parameters, most approaches focus on the motion estimation of the joints on the body segments between consecutive frames. The simplest representation is the Motion Light Display (MLD) which was originally considered by Johansson [1]. Similar to MLD, researchers put markers on some important locations of the human figure for motion analysis. The stick figure [2] that consists of line segments linked by joints can also be analyzed to provide the key to the motion estimation and recognition of the whole figure. The motion of the joints provides the key to the motion estimation and recognition of the whole figure of human body that are used to analyze the human motion and compute motion parameters. Campbell and Bobick [3] adopt the idea of the phase space to recognize the human body motion. They put 14 markers on the joints of human body, and analyze the data by tracking the movement of these markers. Webb and Aggarwal [4] started their study by making the fixed axis assumption: all movements consist of translations and rotations about a fixed-orientation for a short period of time. The jointed objects are recovered by analyzing the collection of rigid parts and unifying the structures. Chang and Huang [5] used a ribbon-based method to analyze the movement of human walking. They extract the extremities of the human body and calculate the angle variation on joints.

To recover the body structure, most researchers use models to fit to the input data for high-level motion interpretation. Akita [6] proposed an image sequence analysis method to solve three basic problems in human motion analysis: modeling, correspondence, and occlusion. The basic goal of his study is to find the outline of human body and recognize the body parts. Elliptical cylinders are used as volumetric models by Hogg [7] and Rohr [8] to model human motion. Each cylinder is described by three parameters: the length, the major and the minor axes. They used a three-dimensional (3D)-model to represent the human body and applied a Kalman filter to estimate the model parameters. Leung and Yang [9,10] applied a 2D ribbon model to recognize the poses of a human performing gymnastic movements. Their system includes two main processes: (1) extraction of the outline of a moving human body; and (2) interpretation of the outline and production of a labeled 2D human body stick figure for each frame. Chung and Ohnishi [11] proposed new 3D model-based motion analysis methods based on so-called *cue circles* (CCs) and *cue spheres* (CSs). Stereo matching for recovering the body model is carried out by finding pairs of CCs between the pair

* Corresponding author. Fax: +886-35715971.
E-mail address: clhuang@ee.nthu.edu.tw (C.-L. Huang).

PII: S0262-8856(00)00046-9

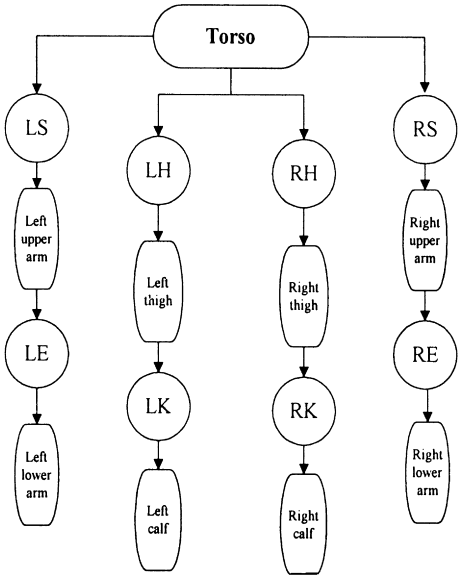

Fig. 1. The description of body structure.

of contour images under considerations: a CS is projected on the two image planes as its corresponding CCs.

To recognize the motion activities from image sequences, we need to successfully track the human motion through the image sequence and analyze the following features for motion type recognition: the outline of the moving body, the skeleton of the body, and the variant of the locations of joints. For human activity recognition, there are two efforts: template matching approach and state space approach. Bobick and Davis [12] compared the features extracted from the given image sequence to the pre-stored patterns during the recognition process. The disadvantage of template matching is its sensitivity to the variance of movement duration. To avoid this problem, state space approach defines each static posture as a state. These states are connected by certain probabilities. Any motion sequence is translated into a sequence of states. Yamato et al. [13] made the mesh features as feature vectors and then applied Hidden Markov Models (HMMs) to recognize the tennis motion. Li et al. [15] propose a model-based shape analysis method to estimate the posture parameters of moving human bodies in visual surveillance applications.

Two concerns are considered in the human motion analysis: the posture recognition and the estimation of motion parameters. In this paper, we propose a model-based method to analyze a real image sequence of a walker and then synthesize the results. Our system has three advantages: (1) we do not put any marker on the human body; (2) we do not use the sophisticated model matching between the 2D or 3D model and the input images; (3) we exploit the multi-state model to effectively recognize the posture; and (4) we generate the motion characteristics of the human body. In Section 2, we give an overview of our system. In Section 3 and Section 4, we discuss preprocessing phase and the model construction phase, respectively. The motion analysis phase is introduced in Section 5. Finally, we illustrate the experimental results in Section 6 and give the conclusion in Section 7.

2. System description

Human body motion is an articulated motion and several body models have been proposed to describe the human body [14]. A human body consists of several rigid ribbons connected by the joints and it can be described by a tree structure shown in Fig. 1. Four branches of the links are connected to the root node(torso) and the connected joint are LE(left elbow), RE(right elbow), LS(left shoulder), RS(right shoulder), LH(left hip), RH(right hip), LK(left knee), and RK(right knee). The extremities are connected at these joints and the angle variation of the eight joints can be used to characterize the human motion in the image sequence.

In each image frame, human body can be segmented from the background and reduced to a body skeleton called *body signature*. Similar to the fact that different handwritten characters have different configurations, each body pose has its own configuration. According to the similarity of the body signatures, the human motion sequence can be classified into several groups of similar postures. Each group is treated as a model state and a human motion can be described by a sequence of model states. We use *posture graph* to depict the inter-relationships among all the model states which is defined as $PG(ND,LK)$, where ND is a finite set of nodes and LK is a set of the directional links between every two nodes. The directional links are also called *posture links*. Each node may have a posture link pointing to itself or other nodes. Each node represents one model state and each link indicates that there is a transition between two model states.

After analyzing the input motion image sequence, we may convert the human motion image sequence to a model state sequence, which can be further described by tracing a path in the posture graph called the *posture transition path*. Analyzing different motion sequence, we may generate different posture transition paths for different postures. The posture transition path contains “stuttered” model states that can be replaced by one single model state. The posture transition graph can be simplified as the *posture path* that is used to identify the motion type.

The same posture path may be generated from different posture transition paths. For instance, both of the posture transition paths, such as $\{11122233\}$ and $\{11111233333\}$, generate the same posture path $\{1^*2^*3^*\}$. The notation “*” denotes a sequence of the same posture node. Two image sequences of the same posture with different moving speed will be described by two different posture transition paths, however, they can also be depicted by the same posture path. For a simple posture, the corresponding posture path may contain only one self-looping node. However, for a

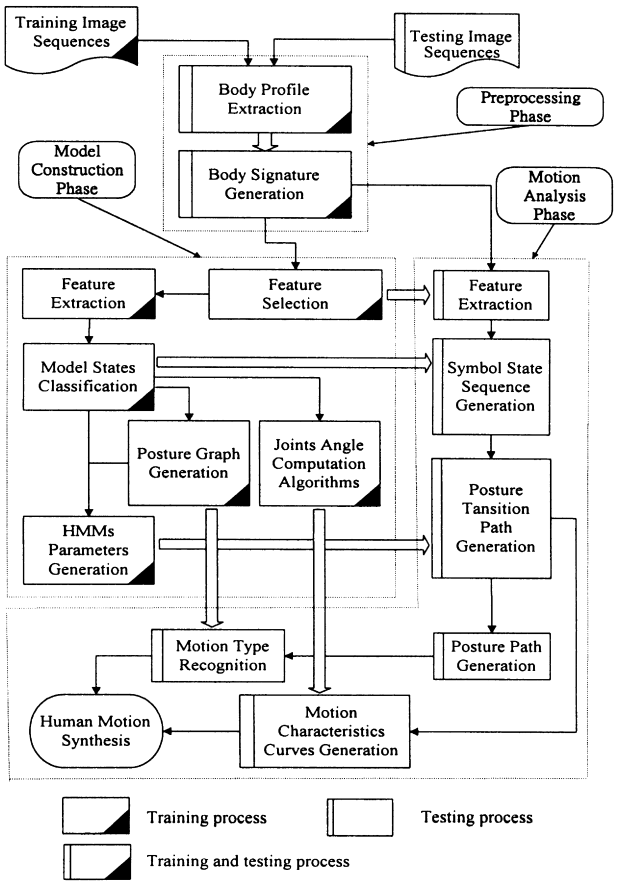

Fig. 2. The system diagram.

complex posture, the corresponding posture path will include many linkages among several different nodes.

This system (Fig. 2) consists of three main processing phases: preprocessing phase, model construction phase and motion analysis phase. Preprocessing phase extracts the human body from the image sequence and generates a sequence of body signatures for further processes. In the model construction phase, given a sequence of training feature vectors, we develop a HMM to model the temporal variation of body signatures of different human postures. In the motion analysis phase, we use the HMM to identify the body signature sequence in terms of a posture transition path. It can be further simplified as a posture path. Furthermore, the procedure for motion characteristics curves generation computes the motion parameters and produces the motion characteristic curves. Finally, the motion parameters are used to synthesize the motion sequence for verification.

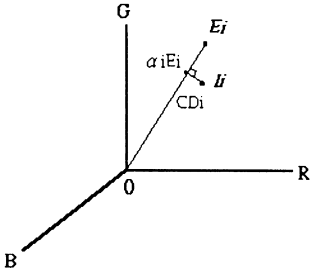

Fig. 3. The difference between I_i and E_i is decomposed into brightness (α_i) and chromaticity (CD_i) components.

3. Preprocessing phase

There are three stages in our preprocessing: (1) body silhouette extraction through background subtraction; (2) morphological filtering; and (3) thinning. Here, we assume that the background is stationary so that we can generate the background model by computing the model parameters over a number of static background frames. Here, we apply the background model proposed by Horprasert et al. [16] which can be used to cope with the local illumination change problems, such as shadows and highlights.

Here, we apply their color model to separate the brightness from the chromaticity component. Fig. 3 illustrates the proposed color model in 3D RGB space. Consider a pixel, **i**, in the image; let $E_i = [E_R(\mathbf{i}), E_G(\mathbf{i}), E_B(\mathbf{i})]$ represent the pixel's expected RGB color in the background model, and $I_i = [I_R(\mathbf{i}), I_G(\mathbf{i}), I_B(\mathbf{i})]$ denote the pixel's RGB color value in a current image that we want to subtract from background. We decompose the distortion of I_i from E_i into two components, *brightness distortion* and *color distortion*. The brightness distortion (α) is a scalar value that brings the observed color close to the expected chromaticity line. It is obtained by minimizing $\phi(\alpha_i) = (I_i - \alpha_i E_i)^2$. α_i represents the pixel's strength of brightness with respect to the expected value. The color distortion is defined as the orthogonal distance between the observed color and the expected chromaticity line. The color distortion of pixel **i** is given by $CD_i = \|I_i - \alpha_i E_i\|$.

The background subtraction consists of three procedures: (1) *background modeling* that constructs reference image representing the background; (2) *threshold selection* that determines appropriate threshold values; (3) *pixel classification* that classifies the pixels into background group and moving object group. After background subtraction and thresholding, we can get binary foreground image, however the image may be corrupted by negative noise inside the object and some isolated noise outside the object. The negative noise makes some small holes inside the object. So, we need to do the morphology filtering to remove the noise.

There are three steps in the morphology filtering: (1) Opening operation which deletes isolated noise and smooth the boundary of shape; (2) Closing operation which fills the holes and connects close regions; (3) Connected components extraction that detects the connected region and select the largest one. The morphological filtering consists of the following operations: (a) *Dilation*. With A and B as sets in Z^2 and Φ denoting the empty set, the *dilation* of A by B is defined as $A \oplus B = \{x|[(\hat{B})_x \cap A \neq \phi]\}$. (b) *Erosion*. For sets A and B in Z^2, the *erosion* of A by B is defined as

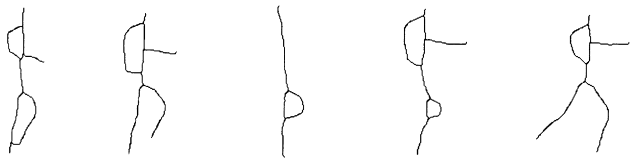

Fig. 4. The body signatures.

$A\Theta B = \{x|(B)_x \subseteq A\}$. (c) *Opening*. The opening of set A by B is defined as $A \circ B = (A\Theta B) \oplus B$. (d) *Closing*. The closing of set A by B is defined as $A \bullet B = (A \oplus B)\Theta B$. Finally, we use the thinning algorithm to generate the skeleton called body signature illustrated in Fig. 4.

4. Model construction phase

Here, we develop three procedures: feature selection and extraction, model state classification, and HMM parameters generation.

4.1. Feature vector selection and extraction

Body signature is an effective graphical qualitative representation for human motion analysis that is converted to a numeric quantitative description for motion analysis. Each body signature has it own configuration that can be characterized by *feature vector*. To select effective features, we have three feature selection criteria: (1) features should be preferably independent of rotation, translation and size; (2) features should be easily obtained; (3) features should be chosen so that they do not replicate each other. There are *topological feature vector* and *parametric feature vector*.

The topological features vector consists of the following components:

1. *Number of end-points* (N_{EP}). The end-points of the body signature are extracted to indicate the visibility of extremities such as head, hands, and feet. In general, there are three kinds of end-points in the body signatures: head end-point, hand end-points and ankle end-points. The possible number of end-points is shown in Table 1.

Table 1
The description of $N_{EP}(\mathbf{f})$

Number	$N_{EP}(\mathbf{f})$			Description
	Head	Arm	Leg	
2	1	0	1	Head and one leg are visible
3	1	1	1	Head, one leg and one arm are visible
	1	0	2	Head and two legs are visible
4	1	1	2	Head, two legs and one arm are visible
	1	2	1	Head, one leg and two arms are visible
5	1	2	2	Head, two legs and two arms are visible

Table 2
The description of $N_{UT}(\mathbf{f})$

Number	Description
0	Body stands still
1	One arm is visible
2	Two arms are visible
3	Two arms are visible and one forms a body signature loop

When a person stands still, the number of end-points is 2 that indicate the minimum number of visible extremities in the walking sequence. We can always find an end-point at the top of the body signature.

2. *Number of T-junctions* (N_T). In the body signature, T-junctions are found at the intersection of the arms and trunk, the legs and trunk, or two legs. We search the T-junctions by using a template matching (i.e. sixteen 3×3 templates) operation on the body signature. If the distribution of the masked points matches the pre-defined T-junction template, the location on the central position of the sliding window is recognized as a T-junction. In the upper part of body signature, T-junction is formed by the intersection of the skeletons of the arms and the torso. There are three different kinds of T-junctions in the lower portion. The number array N_T is composed of two components; the number of T-junctions in the upper body and the number of T-junctions in the lower body, i.e. N_{UT} and N_{LT}. Tables 2 and 3 illustrate the possible number of the T-junction in the upper body and lower body, respectively.
3. *Number of the body signature loops* (N_L). Analyzing body signatures, we may find one or two closed loops. However, when a person stands still or strides, we find no loop in the body signature. There are two kinds of body signature loops: (1) *arm loop*: it is formed by arm and torso, and it is located on the upper part of the body signature; (2) *leg loop*: it is formed by two intersecting legs, and it is located on the lower part of the body signature. Fig. 4 illustrates five examples of body signature loops. To analyze the structure of the body signature, we develop a *body signature loop searching algorithm* to find the body signature loops and then determine the number and the location of loops. Table 4 shows the possible number of body signature loops.

Table 3
The description of $N_{LT}(\mathbf{f})$

Number	Description
0	Body stand still
1	Two legs are separable
2	Two legs form a body signature loop

Table 4
The description of $N_L(\mathbf{f})$

Number	$N_L(\mathbf{f})$		Description
	Upper body	Lower body	
0	0	0	Human stands still or strides
1	1	0	One arm loop
	0	1	One leg loop
2	1	1	One arm loop and one leg loop

The parametric feature vector consists of the following components:

1. *Relative distance* (ρ_{RF}) *and angle* (θ_{RF}) *of ankle end-points*. These features can be identified when the two legs are segmented. The distance between two ankle end-points is normalized by dividing the height of human body. The relative distance and angle are defined as

$$\begin{cases} \rho_{\mathrm{RF}} = \frac{1}{H}\|\mathbf{LF} - \mathbf{EF}\|_2 \\ \theta_{\mathrm{RF}} = \arctan[(y_{lead\_f} - y_{lag\_f})/(x_{lead\_f} - x_{lag\_f})] \end{cases} \tag{2}$$

where $\mathbf{LF} = (x_{lag\_f}, y_{lag\_f})$ and $\mathbf{EF} = (x_{lead\_f}, y_{lead\_f})$ are the coordinates of two leg end-points, H is the height of the skeleton of human body, and $\|\cdot\|_2$ denotes the Euclidean norm.

2. *Relative distance* (ρ_{RA}) *and relative angle* (θ_{RA}) *of hand end-points*. If the number of hand end-points is two, then we may find the relative distance and the angle of the two hand end-points. Similar to Eq. (2), parameters ρ_{RA} and θ_{RA} are defined as

$$\begin{cases} \rho_{\mathrm{RA}} = \frac{1}{H}\|\mathbf{LA} - \mathbf{EA}\|_2 \\ \theta_{\mathrm{RA}} = \arctan[(y_{lead\_a} - y_{lag\_a})/(x_{lead\_a} - x_{lag\_a})] \end{cases} \tag{3}$$

where $\mathbf{LA} = (x_{lag\_a}, y_{lag\_a})$ and $\mathbf{EA} = (x_{lead\_a}, y_{lead\_a})$ are the coordinates of two hand end-points, H is the height of the skeleton of human body.

3. *Respective distance* (ρ_A) *and angle* (θ_A) *of hand end-points*. Since sometimes we cannot find two end points on the body signature, in addition to the relative features ρ_{RA} and θ_{RA}, we define the respective distance ρ_{A} and angle θ_{A} as

$$\begin{cases} \rho_{\mathrm{A}} = \frac{1}{H}\|\mathbf{A} - \mathbf{C}\|_2 \\ \theta_{\mathrm{A}} = \arctan[(y_a - y_c)/(x_a - x_c)] \end{cases} \tag{4}$$

where $\mathbf{A} = (x_a, y_a)$ is the coordinates of the hand end-points (either leading point or lagging point or both) and $\mathbf{C} = (x_c, y_c)$ is the coordinate of the head end-point.

4. *Distance* (ρ_T) *and angle* (θ_T) *of leg T-junction*. If the number of leg T-junction is one, then we may find the relative distance ρ_{T} and the angle θ_{T} of leg T-junction as

$$\begin{cases} \rho_T = \frac{1}{H}\|\mathbf{T} - \mathbf{C}\|_2 \\ \theta_T = \arctan[(y_T - y_c)/(x_T - x_c)] \end{cases} \tag{5}$$

where $\mathbf{T} = (x_T, y_T)$ is the coordinate of the leg T-junction and $\mathbf{C} = (x_c, y_c)$ is the coordinate of the end-point of human head, and H is the height of the skeleton of human body.

5. *Bias* (B). If the T-junction and two ankle end-points are found, then we may identify the bias, B, by calculating the central point of the triangle enclosed by the head point and two ankle end-points. B represents whether the walker is stretching out his leading leg or folding his two legs.

$$B = \begin{cases} -\omega_1 & \text{when the legs are folding.} \\ +\omega_2 & \text{when the legs are stretching.} \end{cases}$$

where $-1 \le \omega_1,\ \omega_2 \le 1$.

6. *Basis area* (Ω_b). If the T-junction and two ankle end-points are visible, we may calculate basis area, Ω_b. It is the normalized triangle area formed by T-junction(T) and two leg end-points which is defined as

$$\Omega_b = \frac{1}{2H^2}|\overrightarrow{V_{TE}} \times \overrightarrow{V_{TL}}| = \frac{1}{2H^2}\sqrt{|\overrightarrow{V_{TE}}|^2|\overrightarrow{V_{TL}}|^2 - (\overrightarrow{V_{TE}}\cdot\overrightarrow{V_{TL}})^2} \tag{6}$$

where E and L denote the leading ankle end-point and the lagging ankle end-point, respectively, $\overrightarrow{V_{TE}}$ is the vector from T-junction to the leading ankle end-point, and $\overrightarrow{V_{TL}}$ is the vector from T-junction to the lagging ankle end-point.

7. *Area* (Ω_L) *of the body signature loops*. The area of a body signature loop is obtained by counting the pixels inside the loop. Ω_L and the topological feature N_{L} are related.

The parametric feature vector has *variable length* since not every component is visible. The existence of the parameter feature component is determined by topological feature component. For instance, if the number of end-point is 2($N_{\mathrm{EP}} = 2$) then only one leg is visible, therefore, the relative distance and relative angle of hand end-points features are not identifiable. And if the number of T-junction is one ($N_{\mathrm{LT}} = 1$), then the distance and angle of leg T-junction can be obtained.

4.2. The classifiers

Based on the variable length feature vectors, we develop two classifiers. The first classifier, the *topological classifier*, categorizes the body signature into certain topological group; whereas the second classifier, the *parametric classifier*, determines to which symbol each body signature may be assigned. If the topological features extracted from the body signature are not complete, then the corresponding

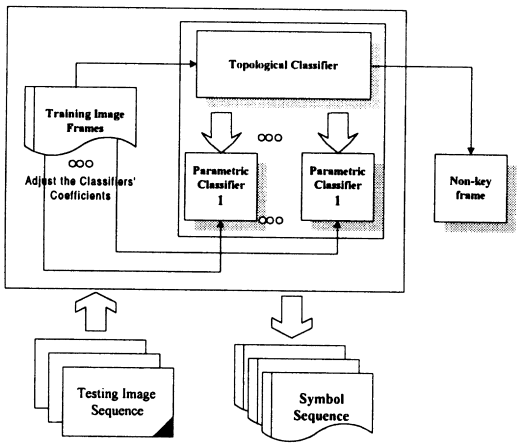

Fig. 5. The two-stage classifier.

image frame will not be treated as a key frame and it will not be processed further. The *parametric classifier* determines to which symbol the body signature belongs by investigating the corresponding parametric feature vector. Each parametric feature vector $\mathbf{f}_{p\times 1}$ will be assigned to a symbol $\mathbf{v}_m$ which is determined by

$$Q(\mathbf{f}) = \mathbf{v}_m \qquad \text{if } d(\mathbf{f},\mathbf{v}_m) \le d(\mathbf{f},\mathbf{v}_i) \tag{7}$$

where $d(\mathbf{f},\mathbf{v}) = [(f_1 - v_1)^{m_1-1}\ldots(f_p - v_p)^{m_p-1}]^{\mathrm{T}}\mathbf{w}[(f_1 - v_1)\ldots(f_p - v_p)]$, $m \neq i$, $1 \le i \le M$, $\mathbf{w}$ is a $p \times p$ diagonal matrix whose elements depict the weight of feature, and m_j is the power of the jth feature difference. The classification process is shown in Fig. 5.

4.3. Posture graph generation

The posture graph describes the relation among the model states and the *model state sequence* (MSS). Each posture node will register the relationship with its related posture nodes by using the following three tags: (1) *T-tag* registers the number of the motion type to which the posture node belongs. The number of the T-tags may be more than one since a posture node may describe several motion types; (2) *Pre-tag* registers the connective model states which are prior to it in the MSS; (3) *Pos-tag* registers the connective model states which are posterior to it in the MSS. After obtaining the model states of the training sequences, we use the *graph growing algorithm* to generate the posture graph (see Appendix).

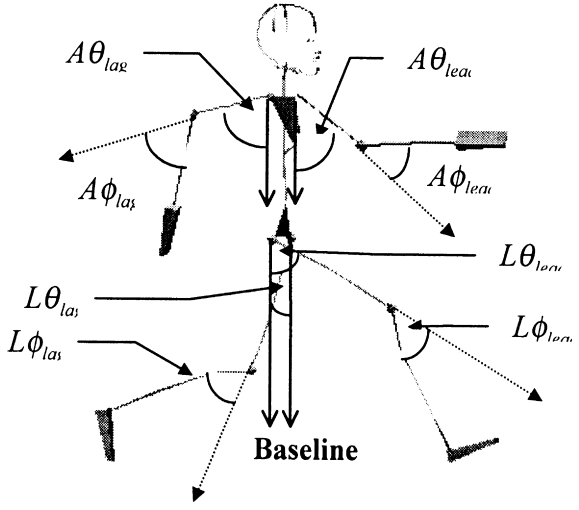

Fig. 6. The graphical demonstration of the joint angles.

4.4. The constraints on joint angle

To analyze the human motion, we find the angles of the eight important joints located on the shoulders, elbows, hips, and knees, respectively, and then we measure the angles associated with these joints. The human body motion can be analyzed by: (a) the speed of the human body motion, $v = f'(t)$, where $f(t)$ is the position of the human body; (b) the angular velocity, $(\theta'(t))$; and (c) the angular acceleration $(\theta''(t))$ of the joint angle, where $\theta(t)$ is the joint angle. Because each individual has his own posture and velocity, the motion characteristics provides an effective description of human motion.

Different walkers performing the same posture may generate similar body signatures. We classify the body signatures into leading arm/leg skeletons and lagging arm/leg skeletons. These joint angle parameters are: (1) arm joint angles ($A\theta_{lead}$, $A\phi_{lead}$, $A\theta_{lag}$, $A\phi_{lag}$); and (2) leg joint angles ($L\theta_{lead}$, $L\phi_{lead}$, $L\theta_{lag}$, $L\phi_{lag}$). The reference line of the human body is defined as the downward vertical line. The primary angle θ is the angle between the upper extremity skeleton and the reference line, and the second angle ϕ is the angle between the upper extremity skeleton and the lower extremity skeleton. Fig. 6 illustrates these joint angles. Based on the observation of the human motion of ordinary people, we have angular constraints on the hips and knees as: $(-\varepsilon \le A\theta_{lead} \le \pi, -\varepsilon \le A\phi_{lead} \le \pi), (-\pi \le A\theta_{log} \le +\varepsilon, -\varepsilon \le A\phi_{lag} \le \pi)$, $(-\varepsilon \le L\theta_{lead} \le \pi, -\pi \le L\phi_{lag} \le +\varepsilon)$, and $(-\pi \le L\theta_{lag} \le +\varepsilon, -\pi \le L\phi_{lag} \le +\varepsilon)$, where ε is a small disturbance.

4.5. HMMs parameters generation

Hidden Markov Model [20,21] has been successfully applied in speech recognition. Recently, it has been applied in image processing and computer vision such as handwritten character recognition and human body motion analysis. Hu et al. [17] developed a handwriting recognition method by using the 1st and the 2nd HMMs. Yamato [13] and Stoll [15] applied HMMs to recognize the human action in time-sequential images, and Starner [18] exploited the HMM technique for American Sign Language recognition. Based on HMM, Samaria [19] developed a method to identify the face expression. Here, we apply the HMMs to analyze the human motion. In the training stage, the first-order HMMs are used to determine the transitions among the model states. The advantage of using HMMs is that the models

can be exploited to investigate the time varying sequences of observations.

A HMM is a double stochastic process; one is not observable or hidden, and the other is observable. The output symbols of the observable process are considered and used to uncover the hidden stochastic process by a succession of computation. The HMM is defined by (S, Π, A, B, V) where (1) N is a number of model states of the posture graph; (2) M a number of observation symbols; (3) T a number of time units; (4) $S = \{S_i | 1 \le i \le N\}$: the set of model states in the posture graph; if the state is k at time t, it is denoted as $q_t = S_k$ where q_t denotes the state at time t; (5) $V = \{v_i | 1 \le i \le M\}$: the set of observation symbols; (6) $A = \{a_{ij} | 1 \le i, j \le N\}$: the state transition probability distribution, where $a_{ij} = P(q_j \text{ at } n+1 | q_i \text{ at } n)$; (7) $B = \{b_j(k) | 1 \le j \le N \text{ and } 1 \le k \le M\}$: the observation symbol probability distribution, where $b_j(k) = P(v_k \text{ at } t | q_t = S_j)$; (8) $\Pi = \{\pi_i | 1 \le i \le N\}$: the initial state probability distribution, where $\pi_i = P(q_1 = S_i)$.

In our system, the number of observation symbols is the same as the number of the model states, i.e. $N = M$. Initial state probability, state transition probability, and observed symbol probability are required for the implementation of HMMs. Here, we introduce the training process to generate these three probability distributions Π, A, and B.

1. *Initial state probability*. In HMMs, each model state has its own initial probability, $\Pi = \{\pi_i | 1 \le i \le N\}$, where the initial state probability π_i is computed by observing the probability that the motion starts from state i. Normally, we assume that the walking sequence starts from a certain state. However, if the input sequence is a segment of one complete walking sequence, the initial state can be any one of the model states. Here, we choose the average model as our initial state probability model, i.e.

$$\Pi = \left\{ \pi_i \middle| \text{where } \pi_i = \frac{1}{N} 1 \le i \le N \right\} \tag{8}$$

2. *Observation symbol probability*. In the HMM training process, with the feature vectors in the training sequence, we develop a classifier to categorize the training feature vectors into observation groups. For each observation group, there is a *representative vector* that is the centroid of the training feature vectors assigned to that designated group. For each input body signature, which may be assigned to state l, we select the best five observations by comparing the corresponding feature vector with a set of representative vectors. The probabilities of observations are determined by the similarity weighting between the feature vector ($\mathbf{A}_l$) and the representative vector ($\mathbf{B}_j$) is defined as

$$W_{l,j} = \sum_{i=1}^{5} D^*(A_l, B_i) / D^*(A_l, B_j), \qquad j = 1, 2, \ldots, 5 \tag{9}$$

where j indicates the specific one of the five selected observations for each input frame, the numerator $\Sigma_i D^*(\mathbf{A}_l, \mathbf{B}_i)$ is the normalization factor, and D^* is the Euclidean distance measure. A smaller distance $D^*(\boldsymbol{A}_l, \boldsymbol{B}_j)$ will generate a larger weight W_{ij}. For an observed body signature sequence, we find that in any time instance, it may be assigned to several possible states with different state probabilities $p(S_k)$. With the state initial probability π_i and the state transition probability a_{ij}, we can calculate the $p(q_j = S_k)$. For each body signature, we may have different observations with different observation probabilities (i.e. $p(v_j)$). In this training phase, given as many body signatures as possible, we calculate the $p(S_k)$ and $p(v_j)$. For each state S_k, we may calculate $p(S_k|v_j)$ by using Eq. (9) to generate W_{lj} and accumulating W_{lj} for observation v_j as

$$p(S_k | v_j) = \frac{\sum_{N_T} W_{k,j}}{\sum_{N_T} \sum_{l=1}^{N} W_{l,j}} \tag{10}$$

where the N is the number of states and the N_T is the number of training image frames. To obtain $p(v_j | q_t = S_k)$, we can use the following relationship: $p(v_j|S_k) = p(S_k|v_j)p(v_j)/p(S_k)$.

3. *State transition probability*. Since the motion speed and frame rate of the same type of motion may not be the same, different posture paths may be generated for the same type of motion. To simplify the training process, we use the following transition probability model:

$$a_{ij} = \begin{cases} a_{ii} & \text{when } j = i. \\ \varepsilon \to 0 & \text{when } j \ne i, \text{ and } S_j \text{ is not connected to } S_i. \\ \dfrac{1}{M_i}(1 - a_{ii} - n_i\varepsilon) & \text{when } j \ne i, \text{ and } S_j \text{ connected to } S_i. \end{cases} \tag{11}$$

where M_i and n_i represent the counts of the model states connected and disconnected to model state i. The intra-state transition probability a_{ii} is defined as:

$$a_{ii} = \frac{1}{U_i} \sum_{k=1}^{U_i} R_k \tag{12}$$

where U_i is the number of sequences containing model state i in the training set and $R_k = NI_k/NS_k$ (NI_k is the count of the transition of state i in the sequence k and NS_k is the count of the total transition of sequence k).

5. Motion analysis phase

In the motion analysis phase, we investigate the motion

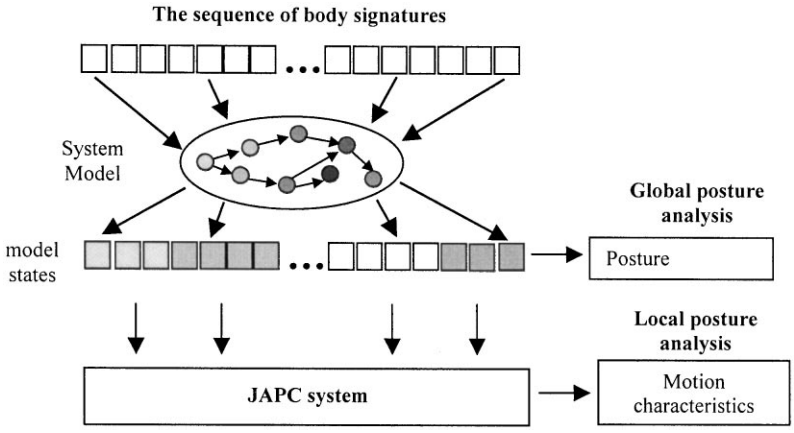

Fig. 7. The model state transitions and posture analysis.

sequence and describe the motion by using the pre-trained models generated in the model construction phase. The motion analysis phase contains two major analyses: *global posture analysis* and *local posture analysis*. Fig. 7 describes the relation between the two analyses.

5.1. Global posture analysis

The global posture analysis finds the inter-posture-node transition in the posture graph. Posture path tracking process determines the transition from one posture node to another. Different posture paths depict different types of human motion.

5.1.1. Generation of the posture transition path

For each input image sequence, we use the two-stage classifier to generate a set of possible symbol observation sequences that are mapped to different posture transition paths in the posture graph. First, we define a posterior probability as

$$\gamma_t(i) = P(q_t = i|O,\lambda) = \frac{P(O, q_t = i|\lambda)}{P(O|\lambda)} = \frac{P(O, q_t = i|\lambda)}{\sum_{i=1}^{N} P(O, q_t = i|\lambda)} \tag{13}$$

where $\gamma_t(i)$ is the probability of being in state i at time t, given the observation sequence O, and the model $\lambda = (\mathbf{A}, \mathbf{B}, \Pi)$. The individually most likely state q_t^* at time t can be determined by using $\gamma_t(i)$ in Eq. (13), that is

$$q_t^* = \arg \min_{1 \le i \le N} [\gamma_t(i)], \qquad 1 \le t \le T \tag{14}$$

To solve the state determination problem more efficiently and correctly, we apply the Viterbi algorithm [20,21] to determine the optimal model state sequence. The Viterbi algorithm is implemented based on dynamic programming methods, which is an effective approach to solve the transition of single state sequence.

Table 5
Arm posture patterns

Categories	# of visible arm skeletons	# of long segments	# of long bent segments	# of short segments	# of arm loops
1	2	2	0	0	0
2	2	0	2	0	0
3	2	0	0	2	0
4	2	1	0	1	0
5	2	0	1	0	1
6	2	0	0	1	1
7	1	0	0	1	0
8	1	0	1	0	0
9	1	1	0	0	0
10	0	0	0	0	0

Table 6
Leg posture patterns

Categories	# of visible leg skeletons	# of long segments	# of short segments	# of leg loops
1	2	2	0	0
2	2	0	2	0
3	2	0	0	1
4	1	1	0	0

5.1.2. Motion type recognition

The output of Viterbi algorithm is the posture transition path, for example, the posture transition path may be {*3 3 3 3 3 3 4 4 4 5 5 5 5 5 9 9 9 9 9 9*}, which can be converted to the posture path {$3^*4^*5^*9^*$}. To recognize the motion type of the posture path, we use the following steps:

1. Prepare a motion type array MTA[i] i = 1,…, MTC, where MTC is the number of different motion types.
2. For the optimal posture transition path, we check the T-tag of each posture node in the path. If the T-tag of the posture node is *i*, then we accumulate the counts of MTA[i], e.g. MTA[PN(k).T–tag[m]]. (The definition of PN(k) is mentioned in Appendix.)
3. The motion type is determined by arg $\max_{1\le i\le MTC}$ [*MTA*(*i*)].

Two different posture transition paths may be mapped to the same posture path. For example, two state sequences {*1 1 1 2 2 2 3 3 3 3 4 4 4*} and {*1 2 2 2 2 2 2 2 2 3 4*} are simplified as the same posture path {$1^*2^*3^*4^*$}. These two motion sequences indicate the same motion type but different motion speed. In global motion analysis, we can identify the motion type of the sequence, but not the detail motion information. The detail information of the motion can be obtained by using the local posture analysis.

5.2. Local posture analysis

The local posture analysis is developed to solve the problem of the intra-posture-node transition in the posture node. Based on the individual configuration of each model state, we may compute the angle variation of each joint of the body signature. By analyzing the body signatures in the training sequences, we have different posture patterns, which are categorized as arm posture patterns and leg posture patterns. They are illustrated in Tables 5 and 6. Different posture patterns indicate different motion characteristics that are to be analyzed.

The last item of the arm posture pattern (see Table 5) indicates that transition frame occurs (i.e. Section 5.2.2). The invisible portion of the joint angles of the upper arm can be easily interpolated by analyzing the angles of the pre- and post-frames. The second item of the leg posture patterns indicates that (1) the leg skeleton is bent between the hip point and T-junction, or (2) the leg skeleton is straight between the hip point and T-junction. The third item of the leg posture patterns indicates that the two legs intersect each other and form a leg loops. If the human body stands still (refer to the last item of leg posture pattern), we may assume that $L\theta_{lead} = L\phi_{lead} = L\theta_{lag} = L\phi_{lag} = 0$. Using the models and the corresponding posture patterns, we may develop a Joint Angle Parameters Computation (JAPC) to calculate the angle parameters. Fig. 8 illustrates the structure of JAPC that consists of a two-layer structure; the first layer is model state layer and the second is posture pattern layer.

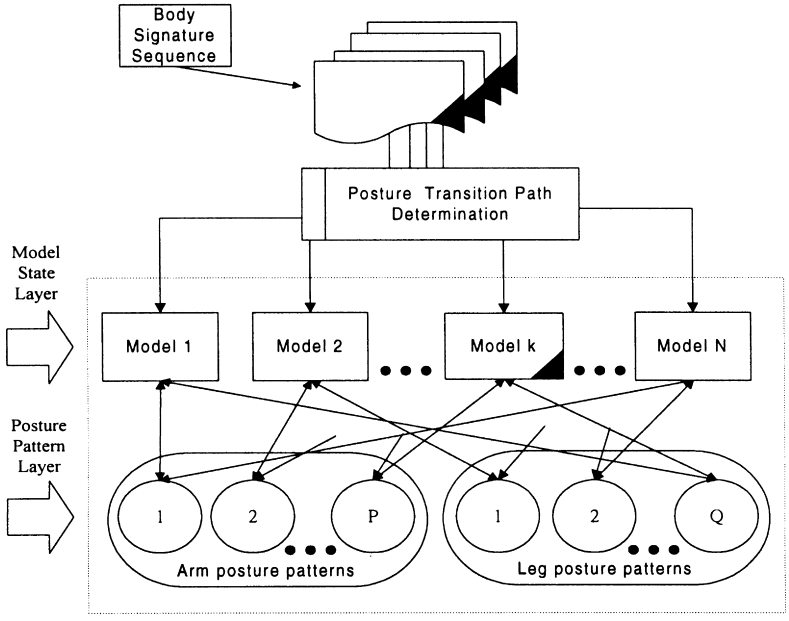

Fig. 8. The structure of JAPCS.

5.2.1. Calculation of the quantitative characteristics of posture patterns

There are some constraints associated with the meaningful body postures. To analyze the human walking more accurately, Lee and Chen [2] proposed fairly five general rules of the walking model: (1) *Both of the two arms cannot be in front of or behind the torso simultaneously. The same restriction also holds for the two legs*. (2) *The arm and the leg which are on the same side of the body cannot swing forward or backward at the same time*. (3) *When both the shoulder joint and the elbow joint of either arm swing, they must swing in a cooperative manner. The same rule holds for the hip joint and the knee joint of either leg*. (4) *The trajectory plane on which the arm or leg swings is generally parallel to the moving direction*. (5) *At any time instant of walking, there is at most one knee having a flexion angle. Moreover, when there is such a flexion in one leg, the other leg stands nearly vertically on the ground.*

Rule 5 is a stringent constraint, in general, it can be applied for an ordinary walking human motion. However, this rule does not fit when people are running or stalking because both legs may have flexion angle. Here, we modify rule 5 as *At the time instance when body signature has leg loop, the secondary angle of one leg is zero*. Beside the above rules, there are three assumptions associated with the body signatures.

*Assumption*1: The body signature has spurious skeletons in the neighborhood of the T-junctions.
Assumption 2: The head point of the body signature is assumed to the location of the shoulder.
Assumption 3: The location of the hip is about half of the body height $0.5H$ and derivation $\sigma = 0.05H$.

Having analyzed all the body signatures of the training sequences, we make the above assumptions of the upper part of the body signature (upper extremities) and lower part of the body signature (lower extremities). These general assumptions are applied for the body signatures of walkers, they can also be used to compute the similar body postures generated by other motion. To compute the quantitative characteristics of each arm/leg posture pattern, we develop a set of procedures called *operation elements* (OEs). With the above rules, the assumptions and the set of OEs, we can compute the motion characteristics for all the posture patterns. The OEs are introduced in the following:

OE 1: Estimate the hip point(H). The vertical location of the hip point is one half of the height of the human body from the head point.
OE 2: Estimate the elbow point(W) relative to the head point. The vertical location of W is 0.24 of the height of the human body from the head point.
OE 3: Estimate the knee point(K) relative to the hip point. The vertical location of K is 0.5 of the height of the human body from the hip point.
OE 4: Locate the knee point on the single long leg skeleton. Given T-junction (T) and ankle end-point (E), OE4 traces the curve linking with T and E by searching the location which has the largest distance with $\overline{\mathrm{TE}}$.
OE 5: Locate the knee point on the compound skeleton. Given the hip point(H), the leg T-junction(T), the leading end-point(E_1), and the lagging end-point(E_2), OE5 finds the knee point(K), the important points of the leading/lagging leg skeleton.
OE 6: Locate the elbow point and end-point in an arm loop. Given the relative area of an arm loop(Ω), the top T-junction(T_1) and the lowest T-junction(T_2) on the arm loop, OE6 finds the elbow point(W) and the arm end-point(E).
OE 7: Locate the elbow point on single long arm skeleton. Given T-junction(T) and hand end-point(E), OE7 traces the curve linking with T and E by searching the location which has the largest distance with $\overline{\mathrm{TE}}$.
OE 8: Locate the hip point. Given leg T-junction (T), OE8 checks if the location of the leg T-junction is within the neighborhood of central position of the height; if yes, then the leg T-junction is the hip point; else call OE 3.
OE 9: Locate the knee point in a leg loop. Given the hip point (H), the T-junctions(T_1 and T_2), the leading end-point(E_1), and the lagging end-point(E_2), OE9 finds the knee point. Note that E_2 may not exist.
OE 10: Find the primary and the secondary angles of a straight leg skeleton. Given hip point(H) and leg end-point(E), OE10 finds the primary and the secondary angles.
OE 11: Find the primary and secondary parameters of leg skeleton. Given the hip point(H), the knee point(K), the leg end-point(E), and the baseline $\vec{B}$, OE11 finds the primary and the secondary angle parameters of leg skeleton.
OE 12: Find the primary and the secondary parameters of arm skeleton. Given the head point(C), the elbow point (W), the end-point (E), and the baseline $\vec{B}$, OE12 finds the primary and the secondary angle parameters of arm skeleton.

5.2.2. Switching process

Walking motion is a cyclic motion, so the leading or lagging leg skeleton in each body signature does not always indicate the appearance of the same leg. For example, the appearance of the same leg in the image sequence may change its appearance from the leading leg skeleton to the lagging leg skeleton and vice versa. For example, the forepart of the curve of angle parameter $\theta_{leading}$ may belong to the right leg angle and then changes to the left leg angle.

To analyze the walking motion, we need to find out the frame in which the left/right leg changes its appearance. The

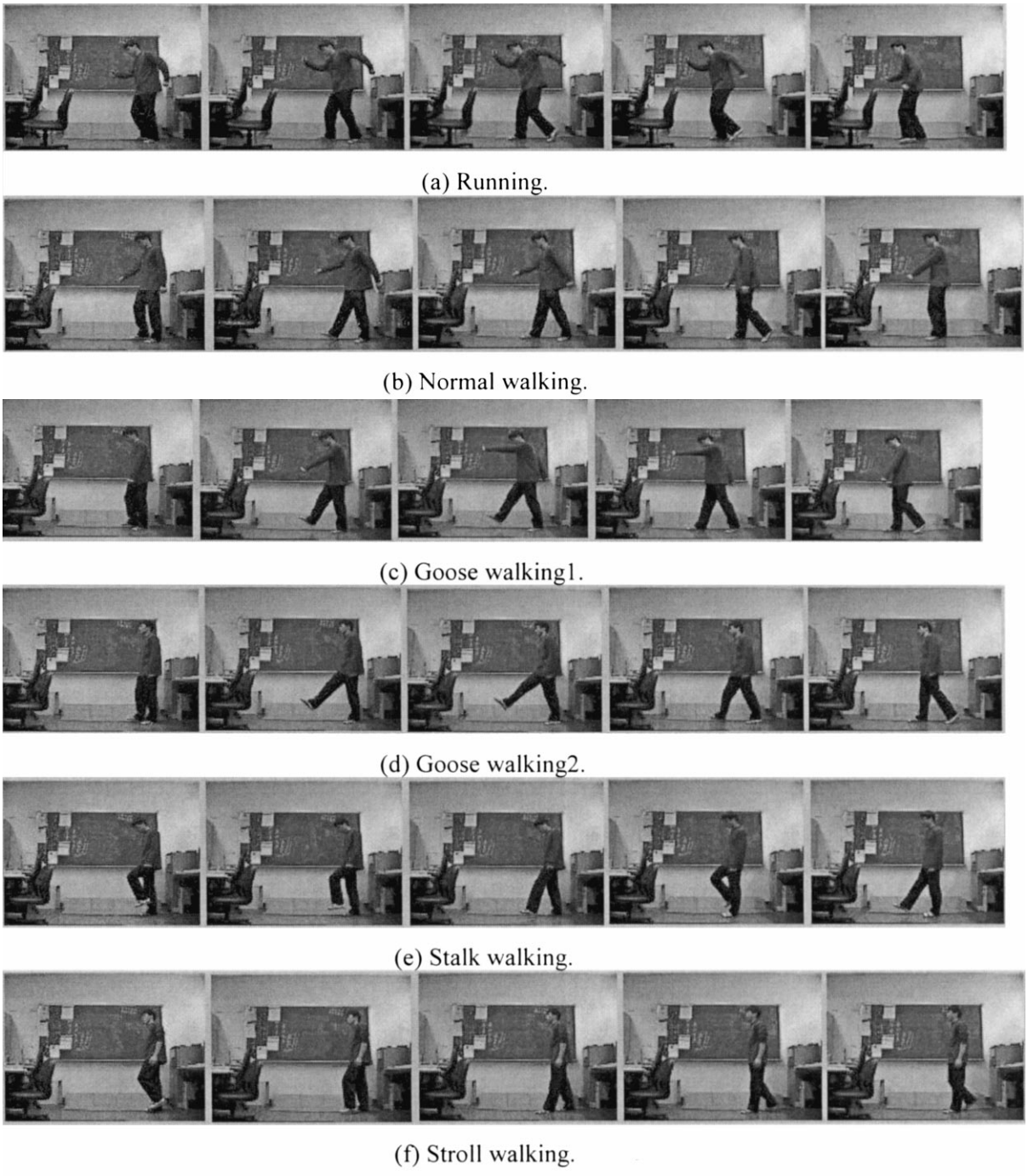

Fig. 9. Six postures.

specific frame is called the *switching frame*. The motion characteristics curves of the leading skeleton or lagging skeleton have to be switched for every half cycle to make a complete cycle of the motion parameter characteristics curve of the left/right extremity. We find the switching frame by observing the variation of the angle θ.

Assume the characteristics curves of the leading and lagging skeletons are denoted as $\theta_{leading}(f_n)$ and $\theta_{lagging}(f_n)$ where $1 \leq f_n \leq N$, N is the total frame number of the input sequence. The switching frame n_s is determined by

$$n_s = \arg\min_{f_n} \|\theta_{leading}(f_n) - \theta_{lagging}(f_n)\|. \quad (19)$$

In general, the values of $\theta_{leading}(f_n)$ and $\theta_{lagging}(f_n)$ are close to zero at the switching frame n_s.

1. Assemble the characteristics curve for the left skeleton as

$$\theta_{left}(f_n)|_{1\le f_n\le n_s} = \theta_{leading}(f_n)|_{1\le f_n\le n_s}\,\theta_{left}(f_n)|_{n_s<f_n\le N} = \theta_{lagging}(f_n)|_{n_s<f_n\le N} \quad (20)$$

2. Assemble the characteristics curve for the right skeleton as

$$\theta_{right}(f_n)|_{1\le f_n\le n_s} = \theta_{lagging}(f_n)|_{1\le f_n\le n_s}\,\theta_{right}(f_n)|_{n_s<f_n\le N} = \theta_{leading}(f_n)|_{n_s<f_n\le N} \quad (21)$$

However, not every input sequence will form a complete cycle, the switching process will not be used if the walking motion is bounded in a half cycle.

Since not every input frame is selected as a key frame, the characteristics curve generated from analyzing the key frames consists of several discontinuous curves. The unknown joint angles of the non-selected frames are located in the gaps between two discontinuous sections of the same characteristics curve. Since the motion of human walking is smooth, we use *linear interpolation* to predict the angle values of the joint of the moving extremities in the non-key frames. If there are consecutive non-key frames, we can estimate the joint angle of these frames by using the joint angle values of the adjacent pre- and post-key frames.

6. Experiments and discussion

In our experiments, the model dressed in black tight clothes performs different motion types, such as running, normal-walking, goose-walking I, goose-walking II, stalk-walking, stroll-walking (see Fig. 9). The most significant information of human motion is the movement of the lower extremities, i.e. the thigh and the calf. These walking sequences cover most of the human walking motions. We use a CCD camera with 256×256 resolution to capture the image sequence (with 30 frame/s). The input walking sequences are firstly converted to the body signatures sequences. Then, the system extracts the feature vectors of each body signature and generates the system parameters. In the training process, we classified the training sequences into 39 model states. Since the initial model state of the test sequences is not biased, each model state is assigned an equal initial probability. For visual verification, we use the software POSER(by Fractal Design Corporation) to simulate the experimental results.

The input walking sequence of the first experiment is shown in Fig. 10 and the corresponding body signatures are displayed in Fig. 11. The input image sequence has 21 image frames. After extracting the body signature of each image frame, the feature vector associated with each signature can be used to determine the state sequence. The length of the sequence is 21. The posture transition sequence (PTP) and the posture path (PP) are shown in Table 7. Since the posture transition path is $\{4^*5^*6^*\}$, the path indicates a normal posture. We use three individual motion characteristics generation procedures to analyze the body signatures of the three model states. The motion characteristics curves of the motion sequence are shown in Fig. 12. In the above figures, '*' denotes the angle value calculated from the image frame and the curves linking between '*'s are approximated by the method of "cubic interpolation". In Fig. 12, we find that the leading leg stretched to the largest angle and then returned to fold. The model bent his knee but toward and unbend when he folded his leading and lagging legs (see Fig. 12(b)). From these figures, we find that the entire motion is in the front interval of the complete walking cycle. We also know that the lagging leg does not bend since the angle is 0 in the walking sequence (see Fig. 12(d)). The synthesized results are illustrated in Fig. 13.

In the second experiment, we use a walking sequence that contains 28 image frames. Fig. 14 illustrates three image frames of the walking sequence and their corresponding body signatures are shown in Fig. 15. The posture transition path and the posture path of the input image sequence are displayed in Table 8. After the posture tracking process, the posture transition path is $\{16^*7^*17^*18^*19^*16^*7^*17^*\}$. Based on the posture path table, we can identify the image sequence as a subsection of the stalk walking. In order to get more detail description of the walking, we compute the joint angle. The angle maps describe the detail information about the movement of the lower extremities of the walker.

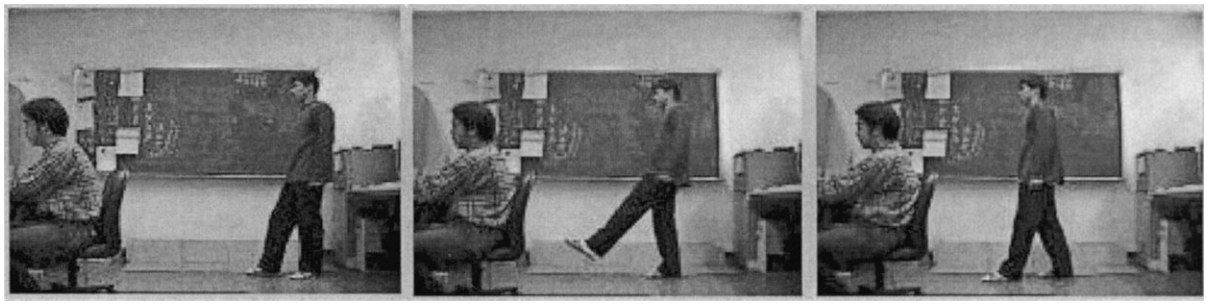

Fig. 10. The image frames of the walking sequence 1.

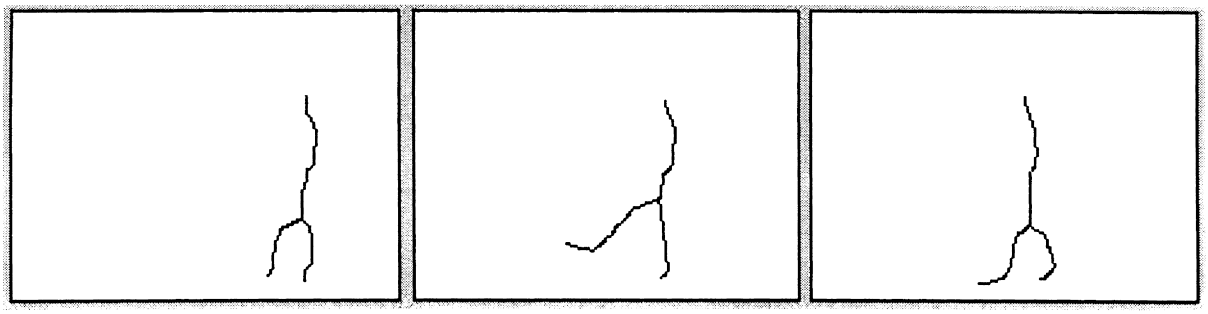

Fig. 11. The corresponding body signatures.

Fig. 16(a)–(d) shows the joint angles of the leading and lagging legs. We find that the angle values on hip vary above zero in Fig. 16(a) and below zero in Fig. 16(c). In Section 5.2.2, we discussed that there exist the switching frames when both the angles approach to zero. After using Eq. (19), we find that there are two switching frames (n_s) in the sequence, and they are the 8th frame and the 27th frame. We use Eqs. (20) and (21) to process Fig. 16(a)–(d). Fig. 16(e)–(h) show the results of which one leg is assigned to be *the first leg* and the other is *the second leg*. From Fig. 16(e), we know that the first leg is lifting up at about 40° in the beginning of the sequence and then lowering down to be a lagging leg. On the other hand, the second leg is the lagging leg in the beginning and lifts up to about 54°, and then lowers downward (see Fig. 16(f)). The first leg straightens itself in almost all the image sequence, but we can observe that it begins to bend from the 24th frame (see Fig. 16(g)), however, the second leg begins to straighten out from this frame (see Fig. 16(h)). We synthesize the experimental results of the motion sequence illustrated in Fig. 17.

Fig. 18 shows the image sequence of the third experiment and the corresponding body signatures are displayed in Fig. 19. The testing image sequence contains 30 image frames. Table 9 shows the posture transition path and the posture path of the sequence. The arm and leg motion characteristics curves are generated. The motion characteristics curves of the arm posture part are illustrated in Fig. 20(a)–(d). In Fig. 20(a), we find that the leading arm swings to the maximum angle at frame 21 and then swings backward. In the meanwhile, we also find that the lagging arm swings to the minimum angle at the same frame. Moreover, the angle variance of the elbow of the lagging arm is smaller than that of the leading arm. Fig. 20(e)–(h) show the motion characteristics curves of the leg posture part. In Fig. 20(h), we note that the knee of the lagging leg is straight at almost all the image frames. Fig. 21 shows the synthesized result of the experiment.

Table 7
Posture transition path (PTP) and posture path (PP) of the walking sequence 1

Time	1	2	3	4	5	6	7	8	9	10	11
PTP	4	4	4	4	5	5	5	5	5	5	5
PP	4*				5*						
Time	12	13	14	15	16	17	18	19	20	21	
PTP	5	5	5	5	6	6	6	6	6	6	
PP	5*(cont.)				6*						

Table 8
The posture transition path and posture path of the walking sequence 2

Time	1	2	3	4	5	6	7	8	9	10	11
PTP	16	16	16	16	16	16	7	7	17	17	18
PP	16*						7*		17*		18*
Time	12	13	14	15	16	17	18	19	20	21	22
PTP	18	19	19	19	19	16	16	16	16	16	16
PP	18*	19*				16*					
Time	23	24	25	26	27	28					
PTP	7	7	7	7	17	17					
PP	7*				17*						

7. Conclusions

In the paper, we propose a multi-state based approach to analyze the human walking sequences. We incorporate the concept of the HMM into our model state system to determine the posture transition path, which can be used to recognize the posture of the walker. Furthermore, we analyze the body signature of different postures to classify the posture patterns of the arms and legs. The proposed system not only recognizes the posture of the input sequence, but also provides the motion characteristics curves to describe the walking sequence. Besides, the motion analysis system can

Table 9
The posture transition path and posture path of the walking sequence 3

Time	1	2	3	4	5	6	7	8	9	10	11
PTP	21	22	22	22	22	22	23	23	23	23	23
PP	21*	22*					23*				
Time	12	13	14	15	16	17	18	19	20	21	22
PTP	23	24	24	24	24	24	24	24	24	25	25
PP	23*	24*								25*	
Time	23	24	25	26	27	28	29	30			
PTP	25	25	25	26	26	26	26	27			
PP	25*			26*				27*			

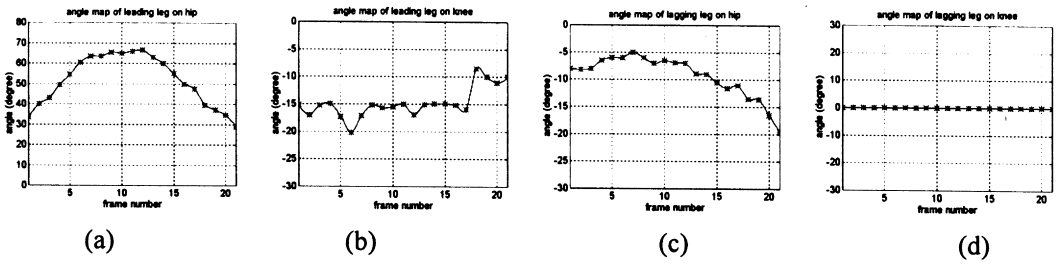

Fig. 12. The motion characteristics curves of: (a) the leading leg on hip; (b) the leading leg on knee; (c) the lagging leg on hip; and (d) the lagging leg on knee.

Fig. 13. The synthesized results of experiment 1.

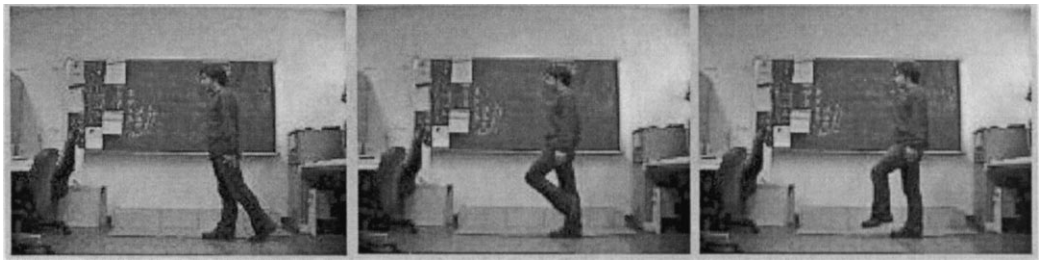

Fig. 14. The image frames of the walking sequence 2.

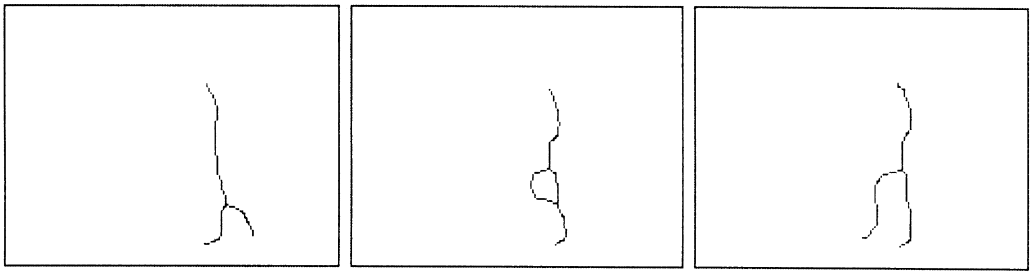

Fig. 15. The corresponding body signatures.

be expanded to other different motion types, e.g. ballet dancer, athlete, etc.

Appendix

A.1. Graph growing algorithm

Step. 1 Collecting all the MSSs of the training set.
MSS(i)[j] stores the model state of the jth element of the ith MSS and MT(i) is the motion type of the MSS(i).

Step. 2 Posture graph initialization
Denotation PN(k) is the kth posture node of the posture graph and there are three tags associated with posture node: T-tag, Pre-tag and Pos-tag.
We choose one MSS as the initial basement of the posture graph, e.g. MSS(i)$i = 1, \ldots, R_i$.
PN(1) ← MSS(i)[1]; //Root node assignment
PN(1).T-tag[1] ← MT(i);
PN(1).Pre-tag[1] ← NULL;
For k, $j = 2$ to R_i
PN(k) ← MSS(i)[j];

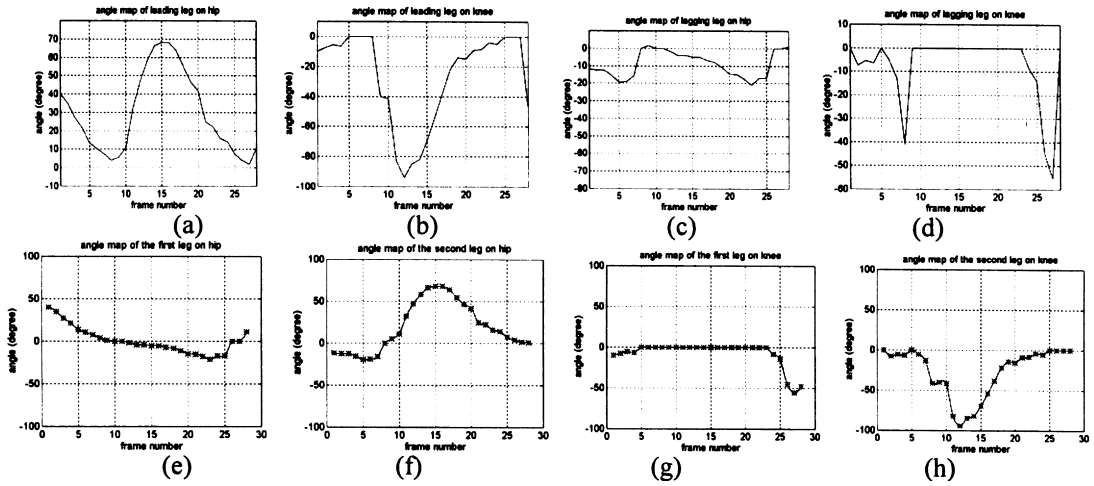

Fig. 16. The joint angle characteristics curve of: (a) leading leg on hip; (b) leading leg on knee; (c) lagging leg on hip; (d) lagging leg on knee; (e) hip of the first leg; (f) hip of the second leg; (g) knee of the first leg; and (h) knee of the second leg.

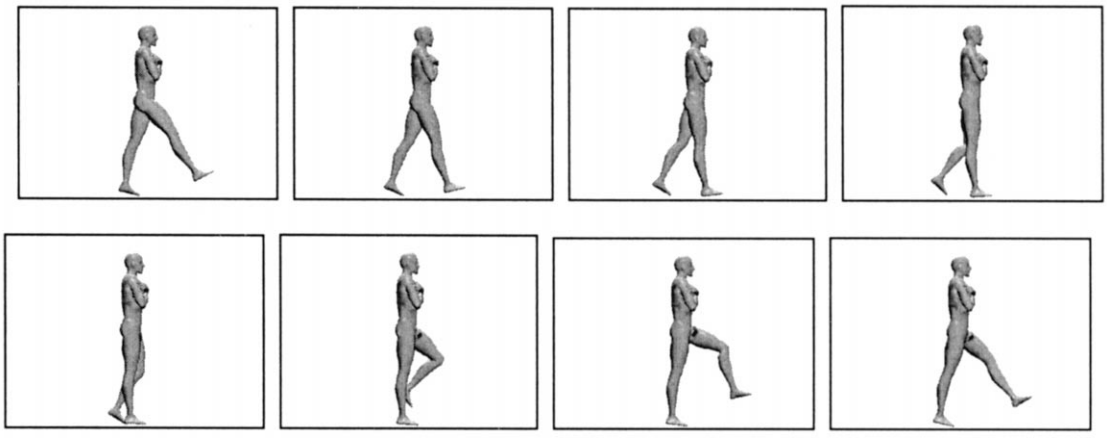

Fig. 17. The synthesized result of experiment 2.

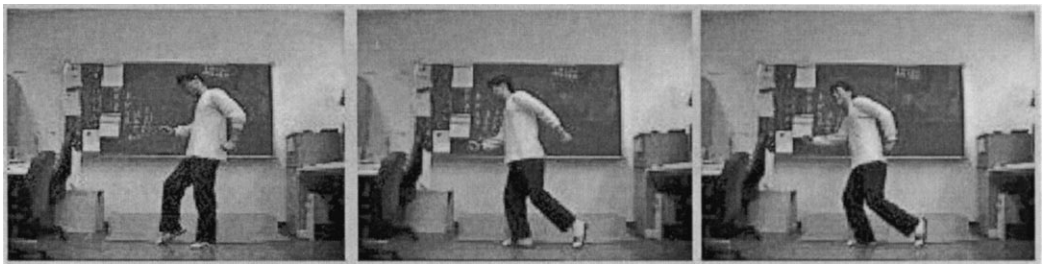

Fig. 18. The image frames of the walking sequence 3.

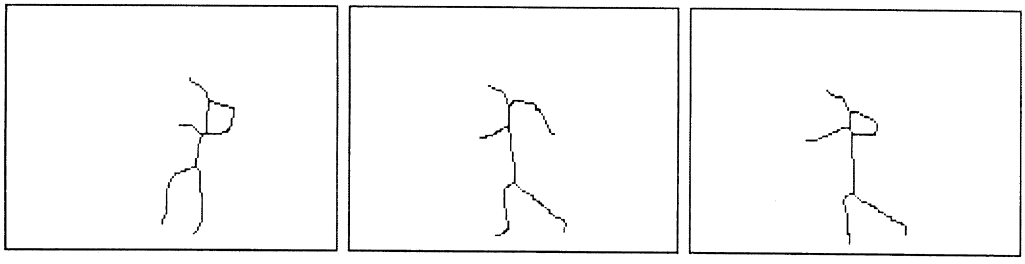

Fig. 19. The corresponding body signatures.

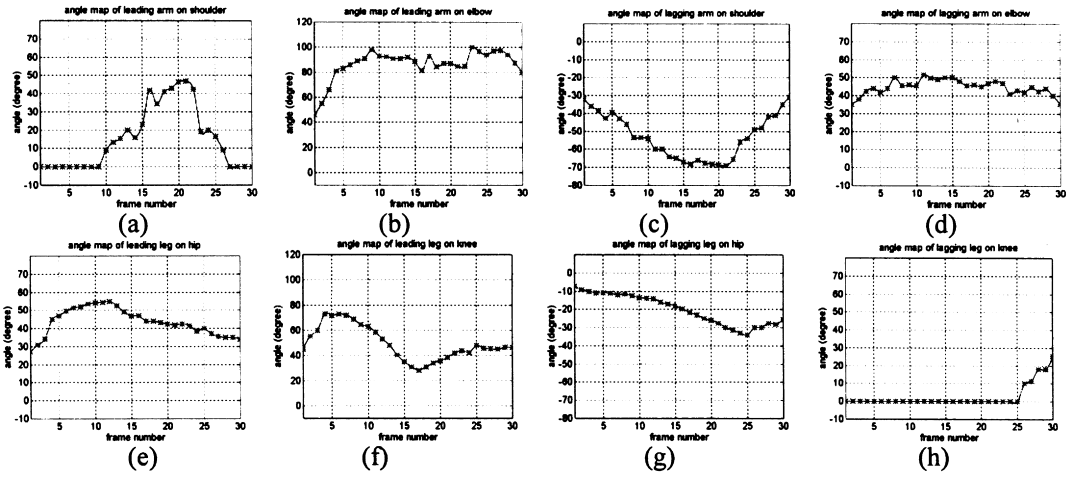

Fig. 20. The motion characteristics curves of: (a) the leading arm on shoulder; (b) the leading arm on elbow; (c) the lagging arm on shoulder; (d) the lagging arm on elbow; (e) the leading leg on hip; (f) the leading leg on knee; (g) the lagging leg on hip; and (h) the lagging leg on knee.

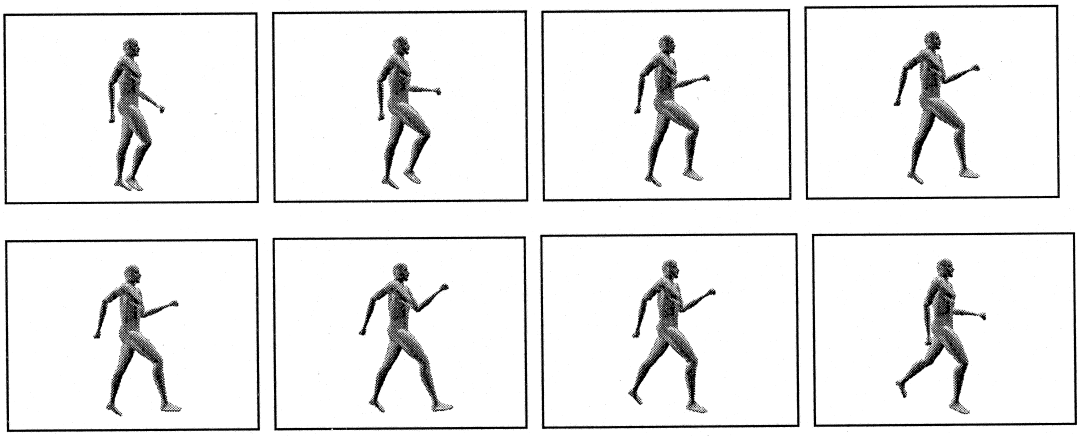

Fig. 21. The synthesized result of the experiment 3.

PN(k).T-tag[1] ← MT(i);
PN(k).Pre-tag[1] ← MSS(i)[j−1];
PN(k − 1).Pos-tag[1] ← MSS(i)[j];
End

Step. 3 Insert new posture node

If a MSS(m) of new motion type is input to the posture graph, the insertion algorithm is stated as followed. Though the motion type is new, the model state may have already existed in the posture graph. The insertion algorithm will be operated in two conditions: (1) the current model MSS(m)[n] is a new posture node state, and (2) the current model state MSS(m)[n] is an old posture node.

(1) MSS(m)[n] is a new posture node
PN(k) ← MSS(m)[n];
PN(k).T-tag[1] ← MT(m);
PN(k).Pre-tag[1] ← MSS(m)[n−1];
PN(PN(k).Pre-tag[1]).Pos-tag[0] ← MSS(m)[n];

(2) MSS(m)[n] is an old posture node
PN(k).T-tag [P + 1] ← MT(m); //P is the count of the motion types to which the posture node belongs.

References

[1] G. Johansson, Visual motion perception, Scientific American 232 (6) (1975) 76–88.
[2] H.-J. Lee, Z. Chen, Determination of 3D human body postures from a single view, Computer Vision, Graphics, and Image Processing 30 (1985) 148–168.
[3] L. Campbell, A. Bobick, Using phase space constraints to represent human body motion, International Workshop on Automatic Face- and Gesture-Recognition, Zurich, 1995.
[4] J.A. Webb, J.K. Aggarwal, Structure from motion of rigid and jointed objects, Artificial Intelligence (1982) 107–130.
[5] I.-C. Chang, C.-L. Huang, Ribbon-based motion analysis of human body movements, Proceedings of ICPR, 1996.
[6] K. Akita, Image sequence analysis of real world human motion, Pattern Recognition 17 (1) (1984) 73–83.
[7] D.C. Hogg, Model-based vision: a program to see a walking person, Image and Vision Computing 1 (1) (1983) 5–20.
[8] K. Rohr, Towards model-based recognition of human movements in image sequences, CVGIP: Image Understanding 59 (1) (1994) 94–115.
[9] M.K. Leung, Y.H. Yang, A region-based approach for human body motion analysis, Pattern Recognition 20 (3) (1987) 321–339.
[10] M.K. Leung, Y.H. Yang, First sight: a human body outline labeling

system, IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (4) (1995) 00.
[11] J.M. Chung, N. Ohnishi, Cue circle: image feature for measuring 3-d motion of articulated objects using sequential image pair, Int. Conf. on Automatic Face and Gesture Recognition, Nara, Japan, April 14–16, 1998.
[12] A.F. Bobick, J. Davis, Real-time recognition of activity using temporal template, IEEE Workshop on Application on Computer Vision, Sarasota, FL, pp. 39–42, 1996.
[13] J. Yamato, J. Ohya, K. Ishii, Recognizing human action in time-sequential images using Hidden Markov Model, Proc. IEEE Conf. on CVPR, pp. 379–385, 1992.
[14] N.I. Badler, S.W. Smoliar, Digital representations of human movement, Computing Survey 11 (1) (1979) 19–38.
[15] Y. Li, S. Ma, H. Lu, A multiScale morphological method for human posture recognition, Int. Conf. on Automatic Face and Gesture Recognition, Nara, Japan, April 14–16, 1998.
[16] T. Horprasert, D. Harwood, L.S. Davis, A robust background subtraction and shadow detection, Proceedings of the Fourth Asian Conference on Computer Vision, Taipei, Taiwan, pp. 983–988, January 2000.
[17] J. Hu, M.K. Brown, W. Turin, HMM based on-line handwriting recognition, IEEE Transactions on PAMI 18 (10) (1996).
[18] T. Starner, A. Pentland, Visual recognition of american sign language using Hidden Markov Models, Int. Workshop on Automatic Face- and Gesture-Recognition, Zurich, 1995.
[19] F. Samaria, S. Young, HMM-based architecture for face identification, Image and Vision Computing 12 (8) (1994).
[20] L.R. Rabiner, A tutorial on Hidden Markov Models and selected applications in speech recognition, Proc. of IEEE 72 (2) (1989).
[21] L.R. Rabiner, B.H. Juang, An introduction to Hidden Markov Model, IEEE ASSP Magazine, January 1986.