



# Flexible signature descriptions for adaptive motion trajectory representation, perception and recognition

Shandong Wu, Y.F. Li\*

<sup>a</sup>Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Kowloon, Hong Kong

## ARTICLE INFO

### Article history:

Received 28 January 2008

Received in revised form 14 May 2008

Accepted 20 June 2008

### Keywords:

Signature

Trajectory descriptor

Trajectory representation

Motion perception

Trajectory recognition

Robot vision

## ABSTRACT

Motion trajectory is a meaningful and informative clue in characterizing the motions of human, robots or moving objects. Hence, it is important to explore effective motion trajectory modeling. However, with the existing methods, a motion trajectory is used in its raw data form and effective trajectory description is lacking. In this paper, we propose a novel 3D motion trajectory signature descriptor and develop three signature descriptions for motion characterization. The flexible descriptions give the signature high functional adaptability to meet various application requirements in trajectory representation, perception and recognition. The full signature, optimized signature and cluster signature are firstly defined for trajectory representation. Then we explore the motion perception from a single signature, inter-signature matching and the generalization of a cluster signature. Furthermore, three solutions for signature recognition are investigated corresponding to different signature descriptions. The conducted experiments verified the signature's capabilities and flexibility. The signature's application to robot learning is also discussed.

© 2008 Elsevier Ltd. All rights reserved.

## 1. Introduction

Motion trajectory is a compact and robust clue for motion characterization and it has been extensively studied for describing actions, behaviors and activities in different applications. Calinon et al. [1] applied motion trajectory to model manipulation task for humanoid robot learning. Ude et al. [2] used motion trajectory to describe full-body movements in transferring human motion to a humanoid robot. Bennewitz et al. [3] investigated human motion pattern representation for robot learning using hidden Markov model (HMM), in which each motion pattern is represented by a cluster of motion trajectories. Common human motions were modeled by motion trajectories in many works [4–8] in which the human gestures, facial expressions and gaits were recognized. Even though the full body motion of human or robot is usually complicated due to the articulated structure, descriptive motion trajectories can still be extracted from the body parts of concern such as the head, hands or feet [8,9]. As the body parts that generate most of the motion trajectories are relatively small in relation to human body, a significant motion points (SMPs) method [10] was proposed towards reliable trajectory tracking for human motion recognition. Chen and Chang [11] explored motion trajectory based video object retrieval from

database, in which wavelet transform was employed to partition the motion trajectories according to different decomposition scales. In general, motion trajectory can play an important role in the characterization of diverse kinds of motions.

However, in most motion trajectory related work, a trajectory was often used directly in its raw data form and effective trajectory description is lacking. In fact, studying flexible trajectory representation is a key problem because it has direct and important influence on trajectory based motion recognition and analysis. Little work was conducted on trajectory representation. Rao et al. [12] presented a dynamic instant based method to characterize salient motion features along a trajectory, but it has difficulty in fully modeling a motion trajectory. Shim and Chang [13] built a moving object trajectory representation scheme based on moving distance and spatiotemporal relations, which is specifically for content-based object retrieval. In our view, to build effective motion trajectory representation, it will be useful to refer to the concept of 'shape descriptor' [14]. Effective motion trajectory descriptor will outperform the raw trajectory data.

In the existing work, some shape descriptors have been developed and used. However, most of them do not perform as well as expected in descriptive capabilities and the adaptability to satisfy different applications for motion characterization. Simple contour functions such as chain code [15], centroid-contour distance and R-S curve are mainly suitable for representation of simple shapes. The descriptors based on Fourier descriptor (FD) [16], wavelet coefficients [17,18] and curvature scale space (CSS) images [19] can represent

\* Corresponding author. Tel.: +852 27888410.

E-mail addresses: [s.d.wu@student.cityu.edu.hk](mailto:s.d.wu@student.cityu.edu.hk) (S.D. Wu), [meyfli@cityu.edu.hk](mailto:meyfli@cityu.edu.hk) (Y.F. Li).

shape in a coarse-to-fine or multi-resolution manner, in which only partial salient features such as the wavelet skeleton, the lower frequency information in FD and the curvature zero-crossing points in CSS are of concern for shape description. This explains why they are actually unable to represent shape uniquely. Also, it may be undesirable to ignore much amount of less-important information when the detailed features really matter. For example, in Fourier transform, it is difficult to perform local motion analysis in the frequency domain because the time information is lost. In CSS, the correspondence problem has to be regulated since a curve's length shrinks along with the Gaussian evolution. The algebraic curves and moment functions [20] suffer from occlusion as they make use of global features. The mathematical curves such as NURBS [21], B-spline [22] and Bezier curve [23] need a fitting process that inevitably causes inaccuracy in shape representation. In particular, the B-spline based method may result in recognition ambiguity as it is hard to compare B-spline parameters directly for recognition because a piece of curve is not uniquely described by a single set of control points [22]. More comparison about the shape descriptors can be found in the surveys in Refs. [24,25].

In this paper, we propose a novel 3D motion trajectory signature descriptor and investigate flexible signature descriptions for adaptive motion characterization. The signature admits three flexible descriptions; hence, it exhibits high functional adaptability. While the full signature is a complete description to the raw trajectory data, the optimized signature is the condensation of the full signature. In addition, a so-called cluster signature is also developed for more effective representation of motion patterns using mixture models. Based on the signature based motion trajectory representations, we further explore the motion characterization in two aspects: trajectory recognition and motion perception. The full signature is particularly good for visual motion perception and the optimized signature is advantageous for fast trajectory recognition. The cluster signature behaves well in both perception and recognition. Here the importance of motion perception is particularly emphasized because it is related much with high-level motion analysis and behavior understanding.

The remainder of this paper is organized as follows. Section 2 presents the definition and implementation of the three signature descriptions. In Section 3, the motion perception is elaborated based on the proposed trajectory signatures. Section 4 is dedicated to the signature recognition. Experiments proceed in Section 5. Section 6 discusses the signature's application to robot learning, followed by the conclusion.

## 2. Signature based trajectory representation

### 2.1. Motivation: flexible trajectory descriptor

Motion trajectories are involved in many different situations where various requirements on motion characterization exist. The basis to account for the different requirements lies in a flexible trajectory description. Raw trajectory data can be used directly. However, such a simple way of using the raw data is quite inflexible because it relies much on the absolute positions of the data points. Instead, building trajectory representations using some intrinsic properties would be a better way to capture the shape features of a trajectory and to avoid suffering from the constraints of absolute coordinates. Furthermore, developing dynamic and adjustable trajectory representation structures (for example, changeable in size) can benefit the adaptability of trajectory descriptions.

Trajectory based motion recognition is one of the most important objectives in motion characterization. The recognition mode, accuracy and efficiency are the three key factors related to the configurations of trajectory representation and recognition engine. Recognition modes can be categorized according to the initialization modes.

The recognition can start with an empty database or it necessitates a prior training database. While both high efficiency and accuracy are desired for a motion recognition system, they may be compromised considering practical situations. For instance, for the recognition out of a large scale database, the need of real time capability sometimes allows acceptable sacrifice in accuracy. Likewise, higher accuracy may also have to bear lower recognition efficiency to some extent. The key to adaptable recognition lies in the availability of flexible trajectory representation mechanisms and classifiers for configuring the appropriate solutions according to actual contexts.

Motion perception is an important function in motion characterization. We define motion perception as visualizing and perceiving the distinctive motion features according to certain rules or referring to a feature profile. Perception can help human or robots to visually and intuitively perceive the concerned features attached in a trajectory, and even to know more about the characteristics of the underlying motion pattern. This is useful for further behavior understanding and reasoning in motion analysis. However, this kind of research has attracted little attention in the past.

In view of the above analysis, building a flexible motion trajectory descriptor is needed to allow high functional adaptability. Basically, a descriptor's capabilities depend much on the kind of the feature we are interested in and the descriptive structure. In the following, we propose a novel signature descriptor and develop three flexible signature descriptions: full signature, optimized signature and cluster signature, which can be used adaptively for motion trajectory representation, perception and recognition.

### 2.2. Definition of the full signature

The full signature is a complete description to the entire raw trajectory data. Inspired by the 2D curve representation in differential geometry [26], we propose the following 3D trajectory signature definition.

**Definition 1.** For a free form motion trajectory  $\Gamma(t)$  parameterized by  $\Gamma(t) = \{X(t), Y(t), Z(t) | t \in [1, N]\}$ , where  $N$  is the trajectory length (frame number), its 3D Euclidean signature  $S$  is defined in terms of four differential invariants: curvature ( $\kappa$ ), torsion ( $\tau$ ) and their first order derivatives ( $\kappa_s$  and  $\tau_s$ ) with respect to the Euclidean arc-length parameter  $s$ , in the following form,

$$S = \{\kappa(t), \kappa_s(t), \tau(t), \tau_s(t) | t \in [1, N]\} \quad (1)$$

where

$$\kappa(t) = \|\dot{\Gamma}(t) \times \ddot{\Gamma}(t)\| / \|\dot{\Gamma}(t)\|^3 \quad (2)$$

$$\tau(t) = (\dot{\Gamma}(t) \times \ddot{\Gamma}(t)) \cdot \ddot{\Gamma}(t) / \|\dot{\Gamma}(t) \times \ddot{\Gamma}(t)\|^2 \quad (3)$$

Since  $s = \int_0^t \|\dot{\Gamma}(t)\| dt$ , we can derive that

$$\kappa_s(t) = \frac{d\kappa(t)}{ds} = \frac{d\kappa(t)}{dt} \cdot \frac{dt}{ds} = \frac{d\kappa(t)}{dt} \cdot \frac{1}{\|\dot{\Gamma}(t)\|} \quad (4)$$

$$\tau_s(t) = \frac{d\tau(t)}{ds} = \frac{d\tau(t)}{dt} \cdot \frac{dt}{ds} = \frac{d\tau(t)}{dt} \cdot \frac{1}{\|\dot{\Gamma}(t)\|} \quad (5)$$

Differential invariants are employed to construct the signature. They are typical local features that enable Eqs. (2)–(5) to be calculated locally. However, the signature components depend on high order derivatives that are sensitive to noise and round-off errors. To reduce this effect and to make the signature robust, we implement an approximate signature that avoids calculating the high order derivatives. To this end, the differential invariants in the signature are solved by the joint differential invariants which involve multiple neighbor points instead of a single point for the signature quaternion

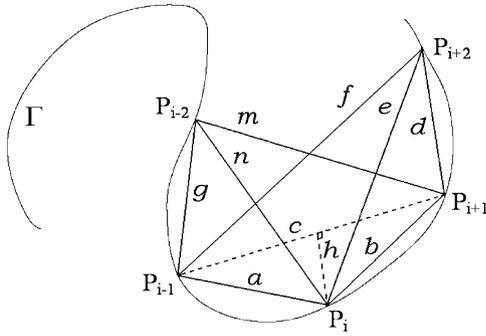


Fig. 1. Multiple neighbor trajectory points are involved for the signature approximation.

calculation. That is, the Euclidean signature is numerically approximated by using the joint Euclidean invariants (inter-point Euclidean distances).

As illustrated in Fig. 1, let  $P_{i-2}, P_{i-1}, P_i, P_{i+1}$  and  $P_{i+2}$  be five consecutively sampled points along a 3D discrete trajectory,  $\Gamma$ , in which the inter-point Euclidean distances are indicated by from  $a$  to  $n$ . Denote  $H^+(H^-)$  by the height of the tetrahedron with sides  $a, b, c, d, e, f$  ( $a, b, c, g, n, m$ ) with respect to point  $P_{i+2}(P_{i-2})$ . Then at  $P_i$  the approximate signature quaternion  $S^* = \{\kappa^*, \tau^*, \kappa_s^*, \tau_s^*\}$  are derived as follows by extending the derivations in Ref. [27].

Firstly, the curvature  $\kappa^*$  at point  $P_i$  can be approximated by the curvature of the circle passing through points  $P_{i-1}, P_i$  and  $P_{i+1}$ . It has been proved that the curvature of a circle is equal to the reciprocal of its radius. Denote  $\Delta_{abc}$  by the area of the triangle with sides  $a, b$  and  $c$ , and define  $\hat{s} = (a + b + c)/2$ , we can derive the following curvature approximation,

$$\kappa^*(P_i) = 4 \frac{\Delta_{abc}}{abc} = 4 \frac{\sqrt{\hat{s}(\hat{s}-a)(\hat{s}-b)(\hat{s}-c)}}{abc} \quad (6)$$

Based on the Taylor series expansion, the torsion is calculated by

$$\tau^*(P_i) = \frac{1}{2} \left( 6 \frac{H^+}{def \cdot \kappa^*(P_i)} + 6 \frac{H^-}{gnm \cdot \kappa^*(P_i)} \right) \quad (7)$$

Here  $H^+(H^-)$  can be derived via the calculation of the tetrahedron's (signed) volume, formulated as follows,

$$\frac{1}{3!} \begin{vmatrix} x_i & y_i & z_i & 1 \\ x_{i-1} & y_{i-1} & z_{i-1} & 1 \\ x_{i+1} & y_{i+1} & z_{i+1} & 1 \\ x_{i+2} & y_{i+2} & z_{i+2} & 1 \end{vmatrix} = V_{abcdef} = \frac{\Delta_{abc} \cdot H^+}{3} \quad (8)$$

where  $\{x_j, y_j, z_j\}_{j=i-1}^{i+2}$  denote the coordinates of vertex  $P_j$ .

Furthermore, two stable signature approximations for  $\kappa_s^*$  and  $\tau_s^*$  are obtained as follows:

$$\kappa_s^*(P_i) = 3 \frac{\kappa^*(P_{i+1}) - \kappa^*(P_{i-1})}{2a + 2b + d + g} \quad (9)$$

$$\tau_s^*(P_i) = 4 \frac{\tau^*(P_{i+1}) - \tau^*(P_{i-1}) + r(\tau^*(P_i)\kappa_s^*(P_i)/6\kappa^*(P_i))}{2a + 2b + 2d + h + g} \quad (10)$$

where  $r = 2a + 2b - 2d - 3h + g$ .

As the joint Euclidean invariants are local features, the approximate signature also admits the computational locality. We define Eqs. (6), (7), (9) and (10) as the full trajectory signature. Because the full signature is based on the features extracted from all the sampled trajectory points, the signature representation is a complete trajectory description. This means that the signature data is capable of capturing and preserving most motion properties (e.g., shape, speed, etc.) for motion perception. More importantly, rich descriptive invariants can be deduced from the signature due to the computational locality. The signature representation is not only invariant

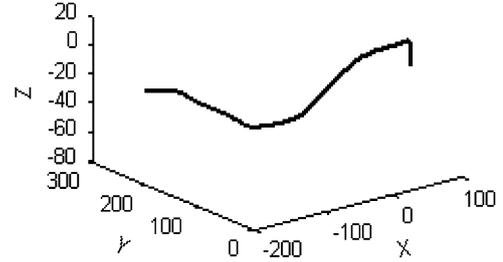


Fig. 2. A piece of 3D motion trajectory  $\Omega$ .

with respect to rigid transformation and viewpoint change, but also relatively invariant with respect to metric transformation and insensitive to occlusion. These invariants offer substantial advantages to enable the signature to perform better than the raw trajectory data.

To ensure the mathematical calculability of the signature, trajectory  $I(t)$  is assumed to be regular, say, for all  $t, \dot{I}(t) \neq \vec{0}$ . For the representation of irregular trajectories, the stationary points will be firstly detected by examining the condition of  $\dot{I}(t) = \vec{0}$  and subsequent to this, we have two options to generate the signature. One is to remove the stationary points to make the irregular trajectory be a regular one. This means leaving one and removing other repeatedly sampled points among a set of consecutive trajectory points. Meanwhile the positions and lengths of the removed points can be saved for later use in motion perception and reconstruction. In this way an individual signature can be calculated for the regulated trajectory. The other option is to segment the irregular trajectory into multiple regular segments according to the stationary points and then combine the signatures of all the segments to represent the entire trajectory. At this stage we adopt the first option, and the segmentation method will be studied in the next stage for joint description for temporally continuous motion trajectories.

Taking the trajectory  $\Omega$  shown in Fig. 2 as an example, the basic forms of its accurate and approximate signatures are illustrated in Figs. 3 and 4, respectively. We have two observations by comparing the signature curves. Firstly, the high signature shape similarity verifies that the approximation algorithms are reliable in preserving the signature features. Secondly, the approximate signature looks more smoothing and stable than the accurate signature, as the sawteeth effects and disturbing saltation in Fig. 3, which were caused by calculating the high order derivatives, are reduced. Hence, the approximate signature is more robust.

In addition, trajectory smoothing is also an effective way to enhance a signature's computational stability by reducing the noise and vibration in the trajectories. Meanwhile, note that trajectory shape may be affected by the smoothing too. Therefore, noise reduction and shape preservation have to be balanced. We design two trajectory smoothers. The first is a moving average filter with an interactively tunable parameter setting in which the span parameter can be determined with a tradeoff between smoothing effect and shape preservation. The other one is a wavelet smoother. To try to preserve the underlying trajectory shape, the wavelet decomposition level for approximate coefficient extraction can be tuned in correspondence with the noise strength. In normal situations when we do not know the noise strength in advance, in principle the decomposition level is restricted to a relatively low range (e.g., 1–3) to avoid over-influencing the trajectory shape. Fig. 5 demonstrates the smoothing for noisy trajectories by the two smoothers. It is observed that the smoothing effect is good with acceptable shape deviations.

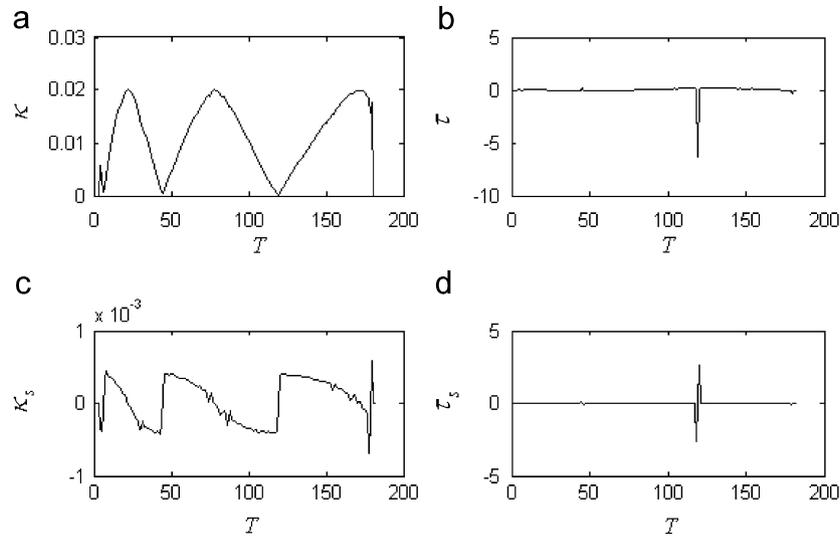


Fig. 3. The accurate signature of trajectory  $\Omega$  shown in Fig. 2. (a)  $\kappa$ , (b)  $\tau$ , (c)  $\kappa_s$  and (d)  $\tau_s$ .

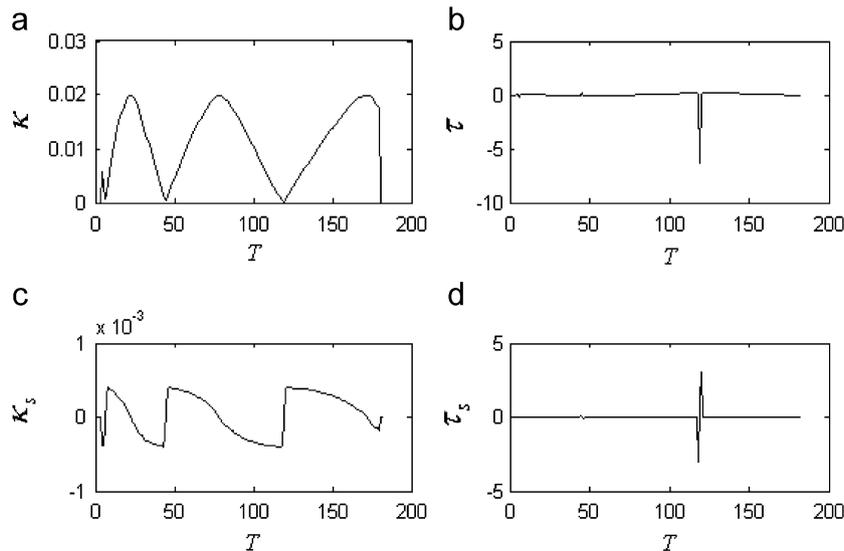


Fig. 4. The approximate version of the signature shown in Fig. 3.

### 2.3. Signature optimization by dimension reduction

The full signature has the same length as the motion trajectory. In case of a complex and longer trajectory, the signature data will be large, which may result in lower efficiency in recognition. In fact, the signature data are interrelated so that they can be optimized by reducing the redundancy to build a more compact signature description.

The optimized signature is achieved with a smaller signature data set but preserving most variance of the full signature. This can speed up trajectory recognition especially for large scale databases. However, it should be noted that most original motion features may be ruined along with the optimization. Hence, it may have difficulty in satisfying the requirements of precise motion perception. The above explains the pros and cons of the optimized signature. In our work, the linear principle component analysis (PCA) [28,29] is applied to the full signature to reduce the dimensions of the signature data.

The PCA transform, also known as Karhunen–Loeve transform, is an effective method for dimension reduction. Via the PCA transform, the interrelated original data are projected onto another feature space where they are small in dimension, uncorrelated and capable of preserving most variance of the original data. The PCA transform can be described by corresponding PCA coefficients. A key problem in the PCA is the choice of the number of principle components, which can be determined by an optimal threshold in practice.

Assume that the singular value decomposition (SVD) is expressed by  $X = UAV^T$ , the PCA transform can be represented by

$$F = U^T X \quad (11)$$

where  $X$  denotes the original data of dimension  $p$  and  $F$  is the projected data. Note that only the first  $m$  columns of  $U$  (principle components) are picked to represent the PCA coefficients. Compared with the dimension  $p$  of  $X$ ,  $F$  is just  $m$ -dimensional and  $m \ll p$ . Therefore, the dimension is much reduced while the data variance is mostly preserved through the PCA transform.

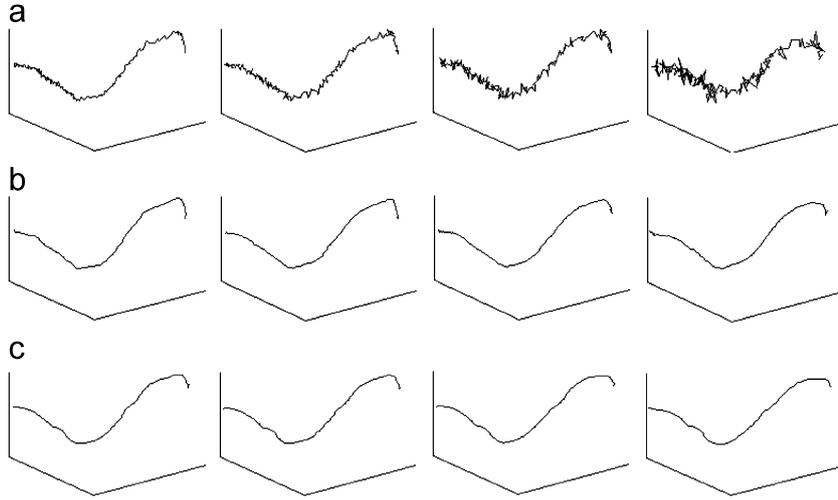


Fig. 5. Smoothing for noisy trajectories (a) by the moving filter (span = 11) (b) and the wavelet smoother (wavelet DB4 and the coefficients at level 2–5) (c).

Here the value of  $m$  is subject to the choice of the cumulative accuracy  $\varphi$ , which corresponds to the degree of data variance being preserved. This problem can be analyzed based on the sorted eigenvalues  $\lambda_i$  of the covariance matrix of  $X$ , formulated as follows:

$$\varphi = \left( \frac{\sum_{i=1}^m \lambda_i}{\sum_{i=1}^p \lambda_i} \right) \times 100\% \quad (12)$$

Hence,  $m$  can be determined by a cut-off of  $\varphi$  such as  $\varphi=95\%$ . Then the first  $m$  columns of the principle components are extracted as  $U_m$ , which is sufficient to guarantee that the expected data variance (in terms of  $\varphi$ ) is preserved in the PCA transform. In general, via the PCA transform ( $U_m$ ), the higher dimension  $X_p$  is converted into the lower dimension  $F_m$ , with a predefined accuracy expectancy indicated by  $\varphi$ .

The above method is carried out to project the full signatures,  $X_p$  onto the PCA space to obtain the optimized signatures,  $F_m$ . Note that the original data,  $X_p$  are composed of all the full signature samples where each sample is  $p$ -dimensional. As the full signature is quaternion-based (four signature components), each signature is rearranged as univariate data for the PCA transform by concatenating the four series of signature profiles in the form of  $X_p = [(k(t))_{t=1}^N (s(t))_{t=1}^N (\tau(t))_{t=1}^N (\tau_s(t))_{t=1}^N]$ , where  $p = 4N$ . The optimized signature,  $F_m$  is smaller in size. That is, more efficient trajectory representation is attained at little cost in the decrease of the description accuracy.

Here, an important problem about the signature length should be pointed out. In the full signature space, a signature retains the same length as the trajectory. Thus the full signatures might differ in length as motion trajectories normally have diverse lengths. However, the PCA transform requires all the input signatures to have the same length. Therefore, the input full signatures have to be pre-normalized to the same length for PCA transform. The problem is that in most cases the pre-normalization operation will affect the original motion properties in the full signature. For example, the speed profile may be changed by the pre-normalization. This means that it may not be reliable to do precise motion perception from the optimized signature. Yet, the optimized signature is useful in case a trajectory is preferably perceived as a whole that is insensitive to certain motion properties. That is, in that situation, it concerns more about the description for a continuous trajectory rather than a sequence of discretely sampled points.

#### 2.4. Cluster signature for motion class modeling

Both the full signature and optimized signature target the representation of a single motion trajectory. In fact, the description to a motion class is also important. Assume that a motion class is characterized by a cluster of similar trajectory signatures, we propose a so-called cluster signature to describe the motion class.

The cluster signature has two merits. Firstly, it is a model-based abstract representation to a motion class. Thus, a motion pattern can be described more efficiently in terms of signature model rather than the combination of multiple individual full or optimized signatures. Secondly, the cluster signature is not sensitive to trajectory length. Hence, it can be compatible with both the full signature and optimized signature. In our work, Gaussian mixture model (GMM) is used to learn a model for a motion class by the density estimation for a cluster of full or optimized signatures.

Following the principle of model-based method, we model each motion class by a probability distribution model. Assume that the number of motion classes is  $C$ , then via training,  $C$  models will be learned that are characterized by respective model parameters  $\{\Theta_i\}_{i=1}^C$ . Assume that  $X_i$  contains  $M$  signature samples (note that the samples do not necessitate the same length), say,  $X_i = \{X_{i,m}\}_{m=1}^M$ , which serve as the training samples to train an individual model  $\Theta_i$ . We firstly rearrange these samples in the form of  $X_i = [X_{i,1}, X_{i,2}, \dots, X_{i,m}, \dots, X_{i,M}]$ . Then the underlying probability density function (PDF) of  $X_i$  can be estimated by a mixture of Gaussian model in the following form:

$$P(X_i|\Theta_i) = \sum_{k=1}^K w_k N(X_i; \mu_k, \Sigma_k) \quad (13)$$

where  $K$  is the number of mixing Gaussian components,  $w_k$  is the mixing weights,  $w_k = P(k|\Theta_i)$  meeting  $\sum_{k=1}^K w_k = 1$ , and  $N(X_i; \mu_k, \Sigma_k)$  denotes the multivariate Gaussian function (mean  $\mu$  and covariance  $\Sigma$ )

$$f_{(\mu, \Sigma)}(X_i) = \frac{1}{\sqrt{2\pi}^d \sqrt{\det \Sigma}} \exp\left(-\frac{1}{2}(X_i - \mu)^T \Sigma^{-1}(X_i - \mu)\right) \quad (14)$$

It is worthy to emphasize the format of the signature  $X_{i,m}$  for the GMM modeling. Besides the signature data, the temporal index  $T$  of a motion trajectory is also incorporated to form an augmented

signature structure in the format of  $X_{i,m} = [T(t); k(t); k_s(t); \tau(t); \tau_s(t)]$  when the full signature is used. The reason for incorporating  $T$  is that it is capable of capturing the temporal characteristics of a motion. However, note that because the signatures may have different lengths,  $T$  may need a pre-normalization to have the signatures aligned reasonably in temporal correspondence. Denote the lengths of the  $M$  signatures by  $\{T_i\}_{i=1}^M$ , we define a bound of temporal range by  $T_{low}$  and  $T_{high}$  (usually  $T_{low} = 1$  and  $T_{high} = \max_{i \in [1, M]} \{T_i\}$ ), then the temporal index of each signature can be normalized to the range of  $[T_{low}, T_{high}]$ . Note that this pre-normalization just adjusts the temporal index for the purpose of reasonable temporal alignments among different signatures, in which the real signature data are not affected at all. This is different from the pre-normalization for PCA transform. With the augmented signature, the dimension  $d$  of  $f_{(\mu, \Sigma)}(X_i)$  will be 5 (1D temporal index plus 4D signature components) for the full signature and 2 (1D temporal index plus univariate signature data) for the optimized signature, respectively.

From the definition of GMM model, a motion class  $\Theta_i$  is characterized by the corresponding model parameter set  $\Theta_i = \{w_k, \mu_k, \Sigma_k\}_{k=1}^K$ . The key problem now is the model parameter estimation. Assume that the number of classes  $C$  is known *a priori* and the training samples are labeled knowing which sample belongs to which class, we combine the expectation–maximization (EM) algorithm [30] and the iterative pairwise replacement algorithm (IPRA) [31] to learn an optimal GMM model parameter set  $\Theta_i$ .

The EM algorithm is an iterative maximum likelihood estimation (MLE) algorithm. It transfers a single and difficult optimization problem into a sequence of smaller and simpler problems. The EM algorithm seeks to maximize the likelihood function by gradient descent (hill-climbing) technique. Here the likelihood function is defined by the log-likelihood  $L(\Theta_i) = \sum_{m=1}^M \log \sum_{k=1}^K w_k N(X_{i,m}; \mu_k, \Sigma_k)$ . The GMM parameter set  $\Theta_i^*$  is learned following the MLE principle  $\Theta_i^* = \arg \max L(\Theta_i)$ .

It should be noted that the results of EM algorithm depend much on the initial parameter setting because the EM algorithm is always monotonically convergent to find the local maximum. Therefore, the choice of the initial parameter values is critical. In practice, the GMM mixing number,  $K$  should be estimated at first, followed by the initialization of the respective GMM component parameters,  $\{w_k, \mu_k, \Sigma_k\}_{k=1}^K$ . Thus, setting a good initial value to  $K$  is the most important basis for successful model learning. Usually,  $K$  is just roughly fixed according to user's guess or determined by trial and error. Relying on user's guess is quite inflexible and inaccurate. The actual situation is that a user has to pay much effort to guess a good value to  $K$  for every GMM model. To improve on this and achieve automatic model learning, we propose to use the IPRA algorithm to iteratively refine the results of the EM algorithm to obtain optimal learning results. That is, our approach is based on the combination of the EM and IPRA algorithms.

The basic principle is as follows: firstly, we activate the EM algorithm by an arbitrary initialization of the mixing number  $K$ ; secondly, we apply the IPRA algorithm to further refine the resulting GMM model parameters  $\Theta_i$  from the EM algorithm, in which  $K$  is also adjusted correspondingly.

In detail,  $K$  is initially set to a relatively big number which can guarantee that the  $K$ -component GMM model is adequately capable of modeling the density. Since a bigger  $K$  leads to more accurate modeling, the more complex the signatures are, the bigger the  $K$  should be set. For example, we can set  $K = 50$  if we estimate that a 50-component GMM is sufficient to describe the signatures we face. Once  $K$  is initialized, the  $k$ -means method is used to estimate the initial GMM parameter values  $\Theta_i^{(0)} = \{w_k^{(0)}, \mu_k^{(0)}, \Sigma_k^{(0)}\}_{k=1}^K$ . With these initial parameter estimations, the EM algorithm iterates the E (expectation) step and M (maximization) step defined as

follows:

E step. calculate the mixing weights (probability  $P(k|X_{i,m}, \Theta_i)$ ) of each signature  $X_{i,m}$ ,  $m \in [1, M]$  that belongs to each component  $k \in [1, K]$ ,

$$P(k|X_{i,m}, \Theta_i^{(0)}) = \frac{P(k|\Theta_i^{(0)})P(X_{i,m}|k, \Theta_i^{(0)})}{P(X_{i,m}|\Theta_i^{(0)})} = \frac{w_k^{(0)}N(X_{i,m}; \mu_k^{(0)}, \Sigma_k^{(0)})}{\sum_{k=1}^K w_k^{(0)}N(X_{i,m}; \mu_k^{(0)}, \Sigma_k^{(0)})} \quad (15)$$

M step. update the weighted mean, covariance and the new mixing weights,

$$\mu_k^{(1)} = \frac{\sum_{m=1}^M P(k|X_{i,m}, \Theta_i^{(0)})X_{i,m}}{\sum_{m=1}^M P(k|X_{i,m}, \Theta_i^{(0)})} \quad (16)$$

$$\Sigma_k^{(1)} = \frac{\sum_{m=1}^M P(k|X_{i,m}, \Theta_i^{(0)})(X_{i,m} - \mu_k^{(1)})(X_{i,m} - \mu_k^{(1)})^T}{\sum_{m=1}^M P(k|X_{i,m}, \Theta_i^{(0)})} \quad (17)$$

$$w_k^{(1)} = \frac{1}{M} \sum_{m=1}^M P(k|X_{i,m}, \Theta_i^{(0)}) \quad (18)$$

The superscript (0) and (1) in the above formulae denote the iteration index. The E step and M step are repeated until the convergence condition is arrived. The convergence condition here is defined as a threshold for the function  $L(\Theta_i)$ , say,  $\frac{L^{(t+1)}(\Theta_i) - L^{(t)}(\Theta_i)}{L^{(t)}(\Theta_i)} < C_{stop}$  (e.g.,  $C_{stop} = 1e-7$ ).

Because  $K$  was initialized by an arbitrary big number, the convergence of the EM process does not guarantee that the resulting GMM model is an optimal learning result. For instance, certain mixing components may be very close or similar. Therefore, we subsequently apply the IPRA algorithm to further optimize the results of the EM algorithm. The core of the IPRA is, among the resulting mixing components  $\{w_k, \mu_k, \Sigma_k\}_{k=1}^K$  from the EM algorithm, to construct a minimum spanning tree connecting all the components, with all the inter-component similarities then measured and examined to iteratively merge the most similar pairs of the mixing components, along which the component parameters are updated correspondingly. After arriving at the predefined minimum similarity threshold, the IPRA process stops eventually giving rise to the optimal GMM model parameters (the mixing number  $K$  and the corresponding component parameters  $\{w_k, \mu_k, \Sigma_k\}_{k=1}^K$ ).

For the two mixing components represented by  $[w_1, N_1]$  and  $[w_2, N_2]$ , respectively, we define the similarity measure between them based on the Hellinger metric as follows,

$$H([w_1, N_1], [w_2, N_2]) = \left| \sqrt{w_1 w_2} \left( 1 - 2 \int \sqrt{N_1 N_2} dx \right) \right| \quad (19)$$

where

$$\int \sqrt{N_1 N_2} = (2\sqrt{2\pi})^d |\Sigma_1|^{1/4} |\Sigma_2|^{1/4} N(0; \mu_1 - \mu_2, 2\Sigma_1 + 2\Sigma_2) \quad (20)$$

If  $H([w_1, N_1], [w_2, N_2])$  is smaller than the predefined merging similarity threshold  $H_{merge}$ , then  $[w_1, N_1]$  and  $[w_2, N_2]$  will be merged, and the method-of-moments (MoM) algorithm is used to update the component parameters as follows,

$$w = w_1 + w_2 \quad (21)$$

$$\mu = \frac{w_1}{w} \mu_1 + \frac{w_2}{w} \mu_2 \quad (22)$$

$$\Sigma = \frac{w_1}{w} \Sigma_1 + \frac{w_2}{w} \Sigma_2 + \frac{w_1 w_2}{w^2} (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T \quad (23)$$

As stated above, applying the EM together with the IPRA algorithms, we can get all the optimal model parameters,  $\{\Theta_i\}_{i=1}^C$  describing the  $C$  motion classes, respectively. We define the GMM model

$\Theta_i = \{w_k, \mu_k, \Sigma_k\}_{k=1}^K$  as the cluster signature. The cluster signature is a probabilistic model and in essence it admits an abstract representation for a motion class.

### 3. Visual motion perception

As mentioned before, the visual motion perception correlates the visualization and perception with the concerned features for potential high-level motion analysis. We distinguish our perception concept with the traditional visual perception theory of machine vision (Gestalt theory or computational vision model). The perception here targets the motion properties of interest that are meaningful for motion analysis. Perceiving motions by a human or robot is important. For example, salient motion features always attract human's attention at the first peek. This explains why it is useful to extract salient features for motion perception. Another example is that sometimes it is desired to have an intuitive interface to 'know' and 'feel' the difference and consistency between two motions. Basically, motion perception cares about a motion instance in terms of distinctive and discriminative features. In this sense, we explore the visual motion perception using the three signature descriptions from the following perspectives: perception from a single signature, inter-signature matching and the cluster signature.

#### 3.1. Properties for motion perception

The above mentioned motion perception is achieved in terms of some properties related with a motion trajectory. In the following we enumerate several properties of interest in our work. Other properties can be added if necessary. In particular, we assume that the motion trajectory sampling rate is fixed in the vision system.

- (1) *Motion length.* Motion trajectories normally have different lengths. Trajectory length can characterize the duration and spatial range of a motion, which can be measured in terms of either the number of sampled points or the entire trajectory arc-length. The two measurements are different since the motion speed affects trajectory sampling.
- (2) *Occlusion.* Occlusion is common in trajectory acquisition (out of field of view or discontinuous tracking). Attributed to the signature's computational locality, occlusion only makes the signature shorter than the original, in length.
- (3) *Shape of motion.* Shape is one of the most important factors for motion pattern discrimination. It can be perceived easily from the curvature and torsion profiles of the trajectory signatures. Intuitively, curvature measures how far a trajectory is from being on a straight line and torsion measures how far it is from being in a plane.
- (4) *Motion speed.* Speed is a key motion feature. A user may perform the same motion with different speed profiles. A lower speed results in denser data sampled and a higher speed results in sparser data. Thus the speed feature can be perceived based on the sampled points' distribution along the trajectories. In fact, the motion trajectory speed profile is characterized by the signature components  $\kappa_s$  and  $\tau_s$  via measuring the traversing arc-length by  $s(t)$ .
- (5) *Group based transformations.* The same motion can be performed independent of absolute positions and sizes. For example, we usually need to deal with various motion instances which admit rigid (translation, rotation, or both) or metric transformations. Because of the signature's rigid and metric invariants, it is easy to perceive the correspondences among different trajectory instances according to an invariant signature description.
- (6) *Salient feature points/moments.* Among the profiles of the motion features captured by a trajectory signature, those salient and dis-

tinctive feature points/moments play important roles for motion analysis. Hence, it is meaningful to locate and extract them for motion perception (see Section 3.2.1 for details).

- (7) *Motion symmetry.* Motion symmetry is a special motion feature in certain motions such as martial arts, dancing or other tasks. Since the full signature can capture both the shape and arc-length features of a motion trajectory, it provides a simple way to detect and perceive motion symmetry (see Section 3.2.2 for details).

#### 3.2. Motion perception from a single signature

As mentioned before, because the optimized signature may ruin the motion properties, only the full signature with complete signature data is reliable for precise motion feature perception. In this section, we demonstrate the perception according to the salient features and motion symmetry from a single full signature.

##### 3.2.1. Salient feature based perception

Generally it is not necessary to take all details into account for motion perception. The salient features are particularly informative to attract human's attention. Also, extracting salient features can benefit motion segmentation and primitive based motion perception. This can usually be conducted by examining the feature of speed, acceleration or shape (curvature and torsion), etc. For example, the extremes (maximums or minimums) of these features usually admit salient information, and the corresponding points/moments can be located easily by measuring the full signature data.

Here we show a special case of salient feature based perception: zero speed perception. According to the signature description for the irregular trajectories in Section 2.2, the stationary point is actually the moment with zero speed. With the stationary points we can highlight the special points of zero speed along a trajectory, indicating meaningful pause/stop actions during a motion. Fig. 6 displays a piece of motion trajectory extracted from a dance fragment with two stationary points highlighted by circles, from which we can perceive the dance fragment more meaningfully and clearly. Since a pause usually consists of multiple continuous stationary points due to continuous trajectory tracking, measuring the positions and lengths of the stationary points is also useful for perceiving the characteristics of the pause.

##### 3.2.2. Perception based on motion symmetry

We define the motion symmetry as two adjacent trajectory fragments symmetrical with respect to a central symmetrical point which connects these two fragments. The signature can depict the motion symmetries existing in a trajectory. To analyze the symmetry, at least three points are needed. Fig. 7 shows a piece of symmetrical trajectory in 3D space, in which  $P^i$  is a central symmetrical point. Away from a central symmetrical point in opposite directions, if a pair of neighbor points (e.g.,  $P^{i-1}$  and  $P^{i+1}$ ) are symmetrical with respect to the central symmetrical point (e.g.,  $P^i$ ), these two points are called neighbor symmetrical points. For the motion symmetry description, we have the following motion symmetry theorem.

**Theorem 1.** For a symmetrical 3D motion trajectory, a central symmetrical point, e.g.,  $P^i$ , is the point whose curvature and torsion are equal to zero, say,  $\kappa^i = 0$  and  $\tau^i = 0$ . For a pair of neighbor symmetrical points, e.g.,  $P^{i-1}$  and  $P^{i+1}$ , the following relations hold,  $\kappa^{i-1} = -\kappa^{i+1}$ ,  $\kappa_s^{i-1} = -\kappa_s^{i+1}$ ,  $\tau^{i-1} = -\tau^{i+1}$  and  $\tau_s^{i-1} = -\tau_s^{i+1}$ .

**Proof.** Obviously, the central symmetrical point  $P^i$  must be an inflection point, therefore,  $\kappa^i = 0$  and  $\tau^i = 0$ , and it means that  $\kappa^{i-1}$  ( $\tau^{i-1}$ ) has an opposite sign with  $\kappa^{i+1}$  ( $\tau^{i+1}$ ). For the neighbor symmetrical points, they must have the same shape, which is fully described by the curvature and torsion. Hence, we have  $\kappa^{i-1} = -\kappa^{i+1}$

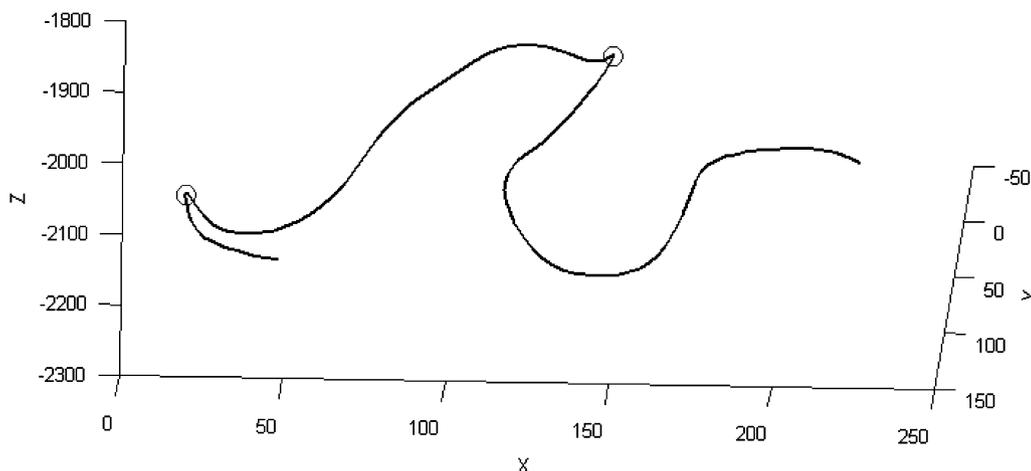


Fig. 6. A trajectory from a dance fragment along which the zero speed moments are highlighted by circles for motion perception.

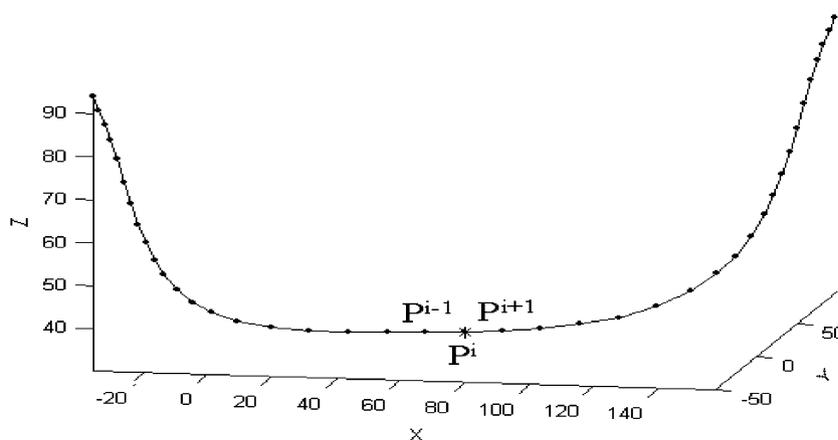


Fig. 7. A piece of symmetrical trajectory (note that it is symmetrical in 3D space although it is not intuitively observed).

and  $\tau^{i-1} = -\tau^{i+1}$ . Furthermore, the symmetrical points must also have the same arc-length with respect to  $P^i$ , say,  $s^{i-1,i} = s^{i+1,i}$ . Consequently, it can be inferred that  $\kappa_s^{i-1} = -\kappa_s^{i+1}$  and  $\tau_s^{i-1} = -\tau_s^{i+1}$ .  $\square$

According to Theorem 1, the motion symmetry in a motion trajectory can be detected by examining the relevant conditions from the full signature. The central symmetrical points should be located first. Then we need to examine whether the neighbor point pairs are symmetrical or not. As it is hard to find absolute symmetry in practical motion trajectories, it would be reasonable to replace the equalization relations in Theorem 1 by approximate equalization.

For better perception to the motion symmetry from the signature profiles, we can coordinate the full signature in terms of a curvature sub-signature ( $\kappa$  vs.  $\kappa_s$ ) and a torsion sub-signature ( $\tau$  vs.  $\tau_s$ ). In this manner, Fig. 8 shows the two sub-signatures of the symmetrical trajectory shown in Fig. 7. It can be observed that a part of symmetrical trajectory is described in the sub-signatures by a symmetrical polygon. Each central symmetrical point (e.g.,  $P^i$ ) leads to a polygon formed. As there are possibly multiple central symmetrical points along a trajectory, the same number of symmetrical polygons will be formed accordingly. With respect to a central symmetrical point, the number of the neighbor symmetrical points (e.g.,  $P^{i-1}$  and  $P^{i+1}$ ) is interrelated with the number of the vertices of the corresponding polygon. The symmetry of two neighbor symmetrical points is embodied by two vertices and two symmetrical sides formed in the

symmetrical polygon. In addition, the length of a polygon side indicates the distinctiveness degree of the symmetrical points. For example, a longer side implies that the related points are more distinctive. This is useful for extracting distinctive symmetrical points for salient feature based motion perception.

### 3.3. Perception by matching two full signatures

In the following, we present a method for matching two full signatures in a nonlinear manner, through which the similarity of the two signatures can be measured. More importantly, the trajectory properties can be highlighted according to the inter-signature matching map for intuitive motion perception.

Inter-signature matching has specific advantage for motion perception. Based on the analysis in Section 3.1, we know that two full signatures may be different in signature length and sampled points' distribution, which may result from the variations in motion speed, sampling rate, occlusion or users' inconsistency in motion repeats. On one hand, this explains why two full signatures cannot be matched in direct point-wise manner. On the other hand, a reasonable matching of two signatures can help to perceive the difference and consistency of the motion properties involved. Therefore, the inter-signature matching is meaningful for intuitive perception of motion characteristics. This can further benefit two typical cases. One, in view of the motion inconsistency existing among multiple

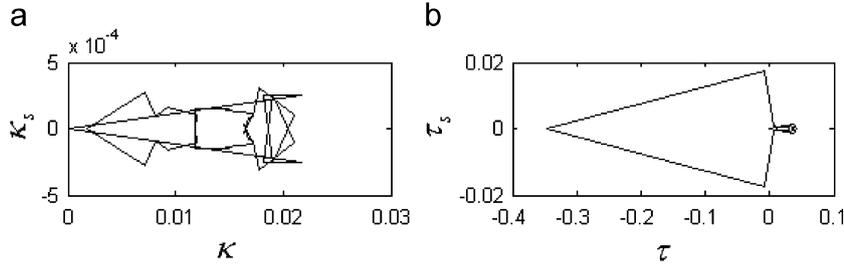


Fig. 8. The curvature sub-signature ( $\kappa$  vs.  $\kappa_s$ ) (a) and torsion sub-signature ( $\tau$  vs.  $\tau_s$ ) (b) for motion symmetry perception.

instances of a user or robot (in fact human cannot perform a motion in exactly the same way each time), the matching of two instances can help to identify the personal motion characteristics of a user. The other case is that when a signature is the reference and the other is a trial, the matching between them can help to find out the problems of the trial more clearly for the improvement in the next trial.

To match two full signatures, the metric must be able to account for the inconsistency in trajectory length and points' distribution. This is actually analogous to the elastic time series/sequences comparison [32]. In our view, an appropriate matching and the similarity measurement lie in finding the best alignment of the element pairs along two signatures. Following this, the nonlinear matching method of dynamic time warping (DTW) is employed to match two arbitrary full signatures.

Dynamic programming is used in the DTW to calculate the minimum cost of the best alignment between two signatures. For two signatures  $S^{*i}$  and  $S^{*j}$  with respective length  $P$  and  $Q$ , let  $S^{*i} = \{[\kappa^{*i}, \kappa_s^{*i}, \tau^{*i}, \tau_s^{*i}]_p | p \in [1, P]\}$  and  $S^{*j} = \{[\kappa^{*j}, \kappa_s^{*j}, \tau^{*j}, \tau_s^{*j}]_q | q \in [1, Q]\}$  represent the sequences of the signature quaternion. The cost function  $d(p, q)$  reflecting the similarity between  $S^{*ip}$  and  $S^{*jq}$  is defined using the Euclidean distance, formulated as follows:

$$d(p, q) = \Delta S^{*ipjq} = \frac{\Delta \kappa^{*ipjq} \cdot \Delta \tau^{*ipjq}}{\sqrt{(S^{*ip})^2} \cdot \sqrt{(S^{*jq})^2}} \quad (24)$$

where

$$\Delta \kappa^{*ipjq} = \|(\kappa^{*ip}, \kappa_s^{*ip}) - (\kappa^{*jq}, \kappa_s^{*jq})\| \quad (25)$$

$$\Delta \tau^{*ipjq} = \|(\tau^{*ip}, \tau_s^{*ip}) - (\tau^{*jq}, \tau_s^{*jq})\| \quad (26)$$

$$(S^{*ip})^2 = (\kappa^{*ip})^2 + (\kappa_s^{*ip})^2 + (\tau^{*ip})^2 + (\tau_s^{*ip})^2 \quad (27)$$

$$(S^{*jq})^2 = (\kappa^{*jq})^2 + (\kappa_s^{*jq})^2 + (\tau^{*jq})^2 + (\tau_s^{*jq})^2 \quad (28)$$

The accumulative minimum cost of aligning up to  $S^{*ip}$  and  $S^{*jq}$  is represented by  $u(p, q)$ , which is determined by the minimum cost among its three neighbors' plus the cost of itself, as expressed by

$$u(p, q) = \min(u(p-1, q-1), u(p, q-1), u(p-1, q)) + d(p, q) \quad (29)$$

Following the above and working from  $u(1, 1)$  to  $u(P, Q)$ , the best alignment of the two signatures is found while the DTW distance between  $S^{*i}$  and  $S^{*j}$  is recorded at  $u(P, Q)$ . In the DTW nonlinear matching, the difference in trajectory length is no longer a problem, and in particular, the one-to-many point correspondence makes the trajectories with different points' distributions aligned well. Benefiting from the DTW matching result, retrieving the indexes of the points from the best inter-signature matching map, the corresponding points between the two trajectories can be aligned by the nonlinear warping paths. Subsequently, the inter-trajectory matching and the warping paths can be visualized to provide an intuitive interface to do motion perception, say, to observe the difference and consistency of the motion properties between two trajectories.

Moreover, it should be noted that the DTW distance at  $u(P, Q)$  can serve as a quantitative similarity measure between the signatures  $S^{*i}$  and  $S^{*j}$ . This also means that DTW distance is a similarity perception for two trajectories. In general, through the DTW inter-signature matching, we can not only get an overall similarity measurement, but also can perceive two trajectories intuitively by visualizing the dynamic warping paths. In this sense, the DTW based method generally behaves better than the proposed matching method in Ref. [33] which is applicable for still curve matching.

#### 3.4. Perception based on cluster signature

An abstract cluster signature is developed in Section 2.4 using the GMM to describe a motion class. Through visualizing the GMM modeling result, say, all the mixing Gaussian components, we can also get an interface to perceive the possible varying range of a cluster signature. According to Eq. (14), a Gaussian model can be characterized by the parameters of the mean and covariance, where the former indicates the center of the modeled data and the latter implies the data's variance. The two parameters can be visualized to perceive a Gaussian model intuitively. Since each individual Gaussian model can cover a part of signature data, the visualization of a GMM model (all mixing Gaussian models) can give an intuition about the possible variance of an entire cluster signature.

Not limited to the perception based on the GMM visualization, we also extend the perception by instantiating a cluster signature from an abstract GMM model. This is equivalent to producing a generalized signature from the signature cluster for a motion class. The generalized signature can be viewed as a canonical signature instance of a motion class. By comparing the generalized signature with all the instances in the original signature cluster, we can perceive the generalization extent, goodness and some specific motion properties of the generalized signature. In practice, the comparison can be carried out using the DTW matching in which the generalized signature serves as the reference. This means that the DTW process can also benefit from the signature generalization when a canonical signature is required to do DTW matching.

To instantiate a generalized signature instance from an abstract cluster signature, we use the Gaussian mixture regression (GMR) method [31]. The regression problem is to reconstruct a general signal form from a set of observations. Arrange the variables in an observation with two sub-sets in the form of  $O = \{X, Y\}$ , where the  $p$ -dimensional  $X$  represents the predictor variable set and the  $q$ -dimensional  $Y$  represents the response variable set. If the joint density of the observation  $O$  falls in the GMM, then when partitioning the joint density by  $f_{X,Y} = f_{Y|X}f_X$ , both  $f_{Y|X}$  and  $f_X$  also have the GMM distribution. This is the basis to derive the GMR aiming at estimating the conditional expectation of  $Y$  given  $X$ . For the signature based regression, the observations are a cluster of signature instances of a motion class, and we know that the signature cluster does fall in the GMM from Section 2.4. For an augmented signature  $O_a = \{T, S\}$

where  $S$  denotes the signature data and  $T$  denotes the temporal index, we follow the GMR to get a generalized signature by estimating the conditional expectation value of  $S$  given  $T$ .

Using the method stated in Section 2.4, a cluster signature is firstly obtained by modeling a cluster of signature samples  $O_a = \{T, S\}$  by the GMM distribution,

$$f_{T,S}(t, s) = \sum_{k=1}^K w_k N(t, s; \mu_k, \Sigma_k) \quad (30)$$

For each learned Gaussian component  $N(T, S; \mu_k, \Sigma_k)$ , rearrange the parameter items  $\mu_k$  and  $\Sigma_k$  in terms of the predictor  $T$  and response  $S$  as follows:

$$\mu_k = \begin{bmatrix} \mu_{kT} \\ \mu_{kS} \end{bmatrix}, \quad \Sigma_k = \begin{bmatrix} \Sigma_{kTT} & \Sigma_{kTS} \\ \Sigma_{kST} & \Sigma_{kSS} \end{bmatrix} \quad (31)$$

Then each Gaussian component is partitioned and the GMM density function can be transformed into the following form:

$$f_{T,S}(t, s) = \sum_{k=1}^K w_k N(s|t; \hat{\mu}_k(t), \hat{\Sigma}_k) N(t; \mu_{kT}, \Sigma_{kTT}) \quad (32)$$

where

$$\hat{\mu}_k(t) = \mu_{kS} + \Sigma_{kST} \Sigma_{kTT}^{-1} (t - \mu_{kT}) \quad (33)$$

$$\hat{\Sigma}_k = \Sigma_{kSS} - \Sigma_{kST} \Sigma_{kTT}^{-1} \Sigma_{kTS} \quad (34)$$

The conditional PDF of  $S|T$  can be formulated by

$$f_{S|T}(s|t) = \sum_{k=1}^K V_k(t) N(s; \hat{\mu}_k(t), \hat{\Sigma}_k) \quad (35)$$

where the mixing weight  $V_k$  is calculated as follows based on the marginal density of  $T$ :

$$V_k(t) = \frac{w_k N(t; \mu_{kT}, \Sigma_{kTT})}{\sum_{k=1}^K w_k N(t; \mu_{kT}, \Sigma_{kTT})} \quad (36)$$

Thus given  $T$ , we can estimate the conditional expectation and covariance of  $S$  as follows:

$$\hat{\mu}(t) = E(S|T=t) = \sum_{k=1}^K V_k(t) \hat{\mu}_k(t) \quad (37)$$

$$\hat{\Sigma}(t) = \text{Cov}(S|T=t) = \sum_{k=1}^K V_k(t) \hat{\Sigma}_k \quad (38)$$

That is, given a predictor vector  $\hat{T}$  (a set of temporal indexes), via evaluating  $\hat{\mu}(t)$  at each temporal index  $t \in \hat{T}$ , a generalized signature  $\hat{S} = \{\hat{\mu}(t)|t \in \hat{T}\}$  can be produced. At the same time, the covariance matrix  $\hat{\Sigma}(t)$  indicates the generalization extent around  $\hat{\mu}(t)$  at each point  $t$ . It is worthy to point out that in the GMR,  $\hat{T}$  can differ from  $T$  in length and the temporal interval, and the resulting signature  $\hat{S}$  will have the same length with  $\hat{T}$ . This offers the feasibility to produce diverse generalized signatures with different lengths and points' distribution by setting different predictors.

Note that since the cluster signature can be developed from both the full signature and optimized signature, the GMR is consequently applicable for both these two kinds of signatures. In particular, if the GMR is conducted using the optimized signature, the generalized signature can still be projected back to the original full signature space making use of the linear transformation property of GMM: if an optimized signature  $\hat{S} \sim N(\mu, \Sigma)$ , we have the corresponding full signature  $U^T \hat{S} \sim N(U^T \mu, U^T \Sigma U)$ , where  $U^T$  is the PCA transform matrix in Eq. (11).

## 4. Signature recognition

Effective trajectory recognition relies on both efficient recognition engine and flexible signature descriptions. In this section, we study adaptive signature recognition in terms of the three signature descriptions by developing three corresponding recognition engines.

### 4.1. DTW matching of full signatures

As mentioned in Section 3.3, the DTW matching of two full signatures gives rise to a DTW distance that can serve as a signature similarity measure, which therefore can be adopted to develop a trajectory classifier. However, note that the DTW based classifier may suffer from two problems. One is the lower efficiency. As the DTW based nonlinear paths warping is time-consuming for longer motion trajectories, the DTW based trajectory classification from larger trajectory databases will become slower accordingly. Therefore, the DTW recognition engine plus the full signature may have difficulty in satisfying the objective of fast trajectory recognition.

The other problem is the recognition rate. As the DTW algorithm always matches all of the points in signatures, the recognition may become ambiguous when occlusion is severe. A modified matching method called minimal variance matching (MVM) [34] can offer higher recognition rate specifically for occluded trajectories. But the MVM method imposes a length constraint to the query and reference. Hence, its application scope is narrower than the DTW method.

Accordingly, it is observed that the DTW based method is particularly effective and suitable for small-scale applications that concern both trajectory recognition and motion perception. For larger databases, the DTW based motion perception is more advantageous than performing DTW recognition.

### 4.2. Fast classification using optimized signatures

As mentioned in Section 2.3, the optimized signature is more compact as the dimension reduction is applied to the full signature. Since all the optimized signatures have the same length after the PCA transform, the nonlinear inter-signature matching is no longer necessary. Instead, two optimized signatures can be matched straightforwardly in point-wise manner. Based on that, we develop a faster trajectory classifier using the optimized signature.

For two optimized signatures  $S^{*i}$  and  $S^{*j}$  with the same length  $N$ , we use the Mahalanobis distance defined as follows to measure the similarity of  $S^{*i}$  and  $S^{*j}$ ,

$$D(S^{*i}, S^{*j}) = \frac{1}{N} \sum_{n=1}^N \sqrt{(S^{*in} - S^{*jn}) \Sigma^{-1} (S^{*in} - S^{*jn})^T} \quad (39)$$

The optimized signature is much shorter than the full signature in length. In addition, the time-consuming DTW matching is replaced by the direct point-wise comparison. Hence, the above formulated signature recognition is more efficient than the DTW based classifier. However, we should note that because the pre-normalization operation to the full signatures may affect certain motion properties, the optimized signature based recognition actually ignores partial original motion features in the trajectories. This is allowable for some applications, but in some cases it may be undesirable.

### 4.3. Bayesian signature recognition

The previous two subsections deal with direct signature matching for trajectory recognition. Although the recognition may not perform as well as expected in efficiency or in the ability in preserving the original motion features, they have obvious advantage in recognition mode. That is, the recognition can start from an empty database

without the need for a prior training process. However, that could also become disadvantageous as it is not able to make use of the available knowledge, which can be derived provided that a trajectory database exists already. Basically, direct signature comparison is a kind of hard recognition. Thus it has difficulty to be further developed for some high-level functions such as uncertainties handling or predictive recognition. On the contrary, the probabilistic learning and recognition methods can make use of the prior knowledge and boost high-level functions.

Using the GMM based cluster signature, we here build a probabilistic signature recognition engine based on the Bayesian Theory. The probabilistic engine can have both higher recognition efficiency and higher accuracy. Only the cluster signatures need to be learned *a priori* from training samples. The Bayesian signature recognition is able to preserve original motion features if the cluster signature is based on the full signature. Therefore, it can satisfy the requirements better, with fast and accurate trajectory recognition even for larger scale databases.

Assume that the cluster signatures of all the motion classes have been learned via the signatures density estimation by GMM (Section 2.4), the query signature will be recognized by the Bayes' decision rule. Based on the  $C$  GMM models  $\{\Theta_i\}_{i=1}^C$ , for a query trajectory signature  $X_q$ , the logarithmic form of the Bayes' Theorem is adopted (it is more efficient in computation than the original form),

$$\log P(\Theta_i|X_q) = \log P(X_q|\Theta_i) + \log P(\Theta_i) - \log P(X_q) \quad (40)$$

The posterior probability  $\log P(\Theta_i|X_q)$  is measured for signature recognition based on the maximum a posterior (MAP) criterion. This means that  $X_q$  is classified into the class  $\Theta_{MAP}$  by examining  $X_q$  as an observation sequence to each GMM model in  $\{\Theta_i\}_{i=1}^C$

$$\Theta_{MAP} = \arg \max_{\Theta} [\log P(\Theta_i|X_q)] \quad (41)$$

$P(X_q|\Theta_i)$  is calculated based on Eq. (13). The prior probability  $P(\Theta_i)$  can be derived from the initial knowledge about the occurrence frequency of the samples. For example, it can be set to  $1/C$  when all the motion classes are equi-probable. The marginal probability  $P(X_q)$  can be calculated by  $P(X_q) = \sum_{i=1}^C P(X_q|\Theta_i)P(\Theta_i)$ .

## 5. Experiments

The objective of the experiments is to conduct descriptive motion trajectory perception and recognition using the proposed signature descriptions. Motion perception is mainly demonstrated by the DTW based nonlinear matching of full signatures and the GMR based cluster signatures. The trajectory recognition performance is tested based on a large trajectory database by developing three signature recognition solutions.

### 5.1. Sign motion trajectory acquisition

Sign language is a special communication modal for human interaction. While human perform a sign by hand, the underlying spatiotemporal motion trajectory is extracted discretely as the representation of the sign word. A binocular vision system with two TM-765 cameras is employed for trajectory tracking. The raw 3D trajectory is calculated by stereo algorithm and smoothed using the moving average filter. The object of interest is tracked using the CAMShift (continuously adaptable mean-shift) algorithm [35]. Note that the tracking is simplified by tracking a distinctive rigid object held by hand instead of direct tracking to the hand. This makes the tracking easier and more stable. The issue of pursuing robust human hand or body parts tracking is out of the scope of this paper. Fig. 9 shows the system setup and several stereo trajectory tracking snapshots of a sign demonstration.

At this stage, only the sign performed by a single hand is used and each sign admits a regular trajectory. Ten sign classes (words) are defined in the experiment, which are roughly close in trajectory shape to the characters of '0' to '9', respectively. Different users are asked to perform the sign words in 3D space and repeat multiple times to get diverse instances. Fig. 10 shows the extracted 3D motion trajectories from a set of sign instances.

### 5.2. DTW based sign trajectory perception

The first experiment demonstrates the free form sign language perception from the full signature by applying the DTW based signature matching. Figs. 11–18 visualize 8 representative cases of the nonlinear inter-trajectory paths warping resulting from the DTW inter-signature matching maps, from which we can get an intuitive perception to some of the trajectory properties listed in Section 3.1.

First, it is observed that the inter-trajectory alignments are quite reasonable, which verified that the calculated DTW distances are reliable quantities to characterize the inter-trajectory similarities. Second, the DTW warping paths reflect the motion specifics between trajectories. For example, we can get an immediate intuition to the motion length in terms of arc-length without counting the sampled points. In fact, only the number of sampled points may not be adequate to characterize the arc-length due to the possible difference in motion speeds.

The instance of I-1/I-2 is an occlusion example compared with R-1/R-2. Occlusion is embodied typically by a one-to-many paths emission where 'one' means an end point of the occluded trajectory and 'many' corresponds to the occluded part of the full trajectory.

From the matching of I-3 and R-3, the speed variability of I-3 can be easily perceived. Although the motion arc-length is close, the data sampling of I-3 is much denser than R-3, which shows that the speed profile of I-3 is much slower than R-3. Furthermore, comparing the respective density of the warping paths of the two trajectories, we can even estimate an approximate scaling relation between the two speed profiles. The pair of I-4 and R-4 is another example of speed perception.

The matching between I-5/I-6/I-7 and R-5/R-6/R-7 illustrates the perception to the correspondences between different trajectory instances due to the signature's invariants. We can find that I-5 is a variation of R-5 after a translation action, and the perfect point-wise matching between them verified the signature's translation invariant. Likewise, I-6 is a rotated variation of R-6 and the matching demonstrates rotation invariant. The overall shapes of I-7 and R-7 are quite similar but the sizes are obviously different, from which we can see that I-7 is either a smaller-size variation or an equal-size variation produced with a longer viewing distance between the demonstrator and visual sensor. Using a normalized cost function for the DTW algorithm by replacing  $\kappa_s$  and  $\tau_s$  with  $(|\kappa_s|/\kappa_s)\sqrt{|\kappa_s|}$  and  $(|\tau_s|/\tau_s)\sqrt{|\tau_s|}$ , respectively, in Eqs. (25)–(28), we can get a metric signature invariant, as verified in the point-wise matching between I-7 and R-7. However, we should notice that for real data of practical instances, it is rather hard to get absolute translation, rotation or metric transformations between two sign instances. For example, the matching between I-5'/I-6'/I-7' and R-5/R-6/R-7 shows the invariants for approximate transformation relations.

From the matching between I-8 and R-8 as well as I-8' and R-8, we can perceive the characteristics of two sign instances which belong to the same sign class. I-8 and I-8' are performed by two different users, but we get reasonable matching results with R-8. We can compare the difference on the shape feature, spatiotemporal shift and speed profile from the matching results. Similarly, the matching between I-9 with R-9 as well as I-9' and R-9 is another example illustrating the perception of various instances produced by different users.



Fig. 9. Vision system setup and snapshots of the stereo trajectory tracking.

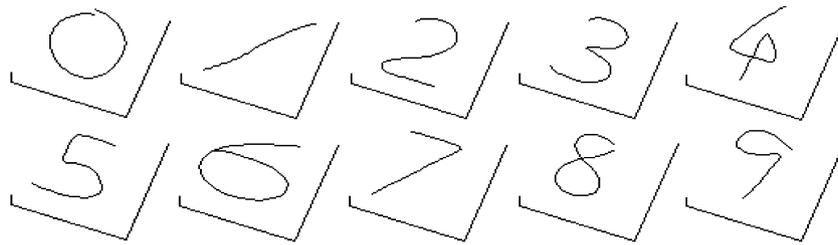


Fig. 10. Motion trajectories extracted from a set of sign instances.

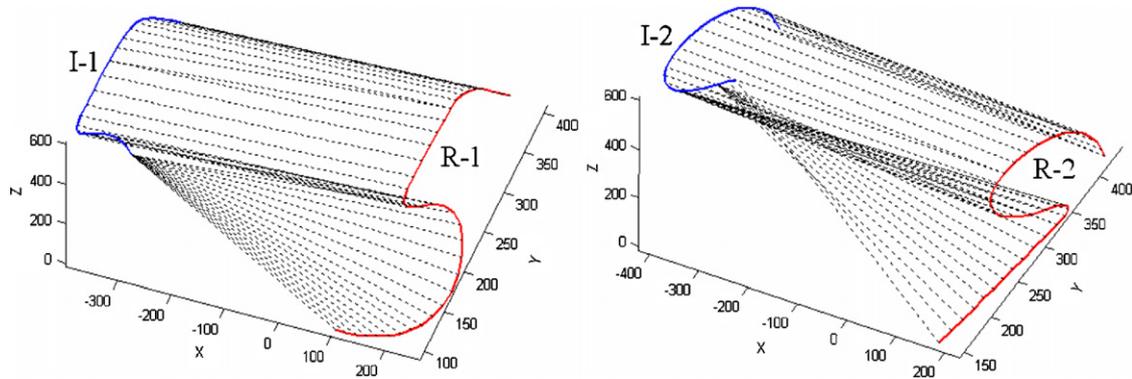


Fig. 11. Perception of occlusion.

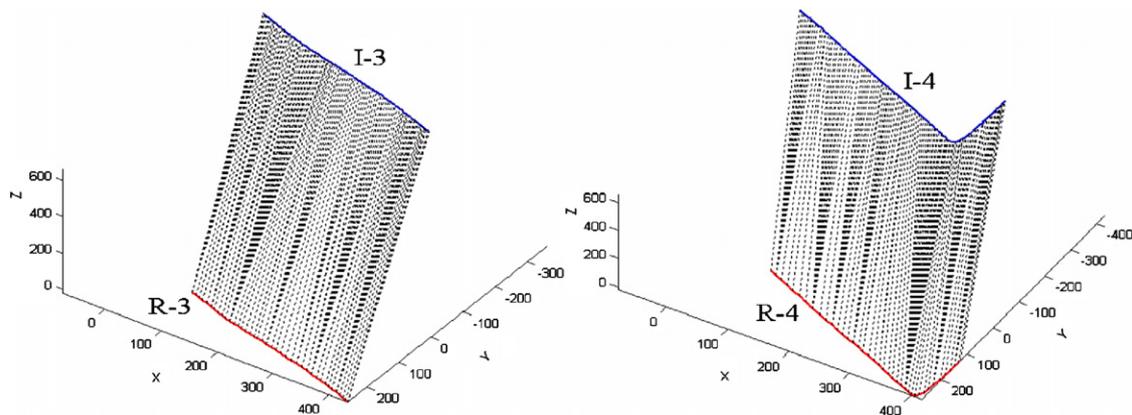


Fig. 12. Perception of speed variability.

The results of the DTW based motion perception may be useful to reveal some specific characteristics of a motion, even to identify a specific demonstrator. This is based on the fact that each different

demonstrator may have specifically its own characteristics embodied in the conducted motion trajectories. For example, given more matching samples like those in Fig. 16, we may feel that the user of

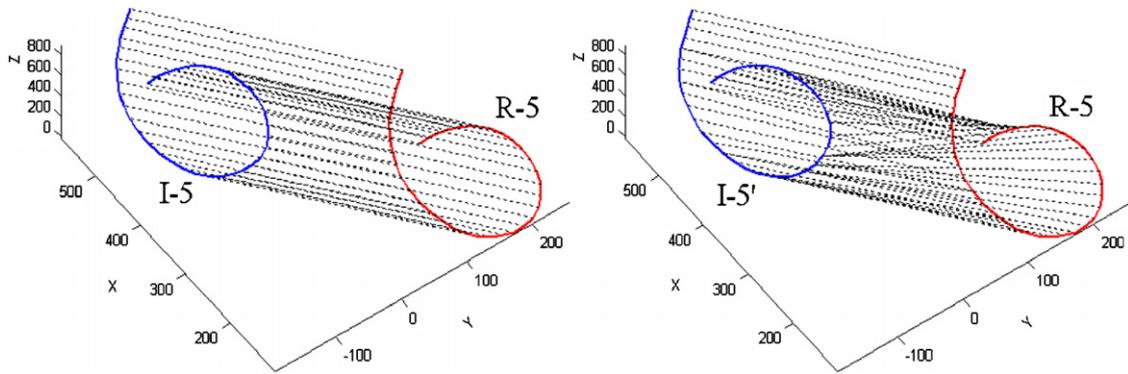


Fig. 13. Perception of translation invariant.

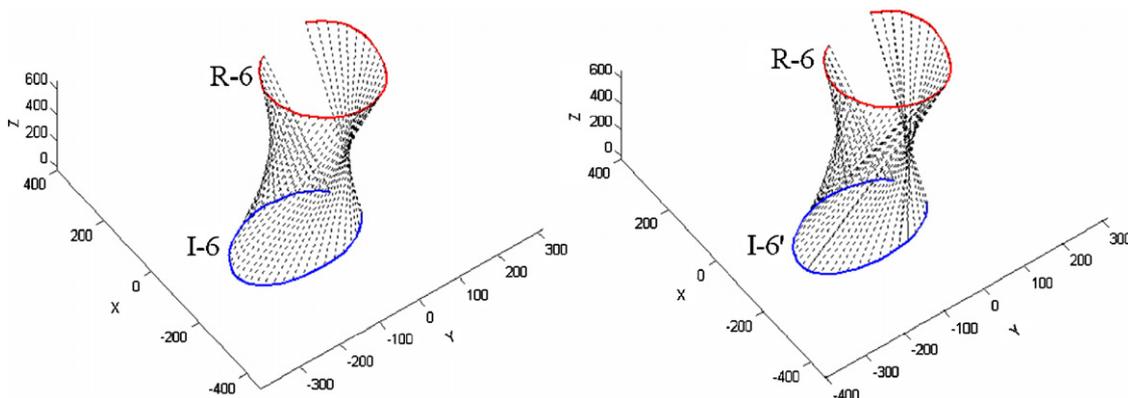


Fig. 14. Perception of rotation invariant.

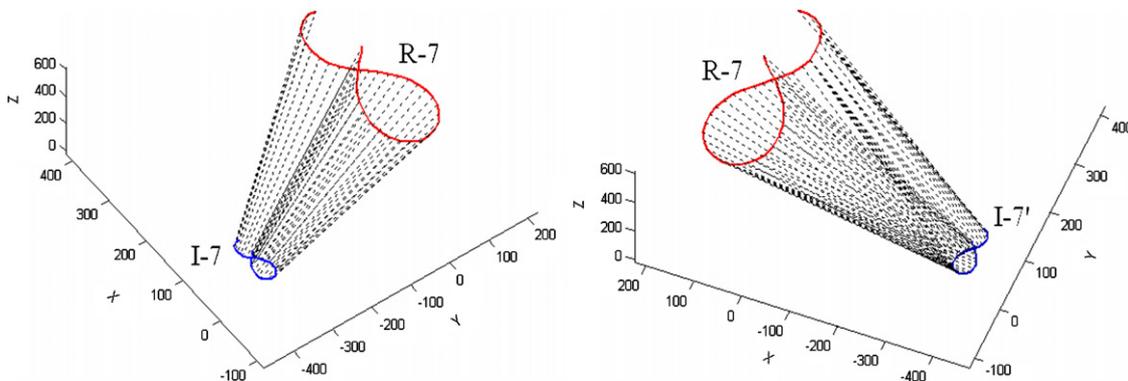


Fig. 15. Perception of metric (relative) invariant.

I-8 and I-9 probably is different from that of I-8' and I-9'. Of course it should be noted that, not all the personal motion characteristics can be visualized or observed. The DTW matching does offer a way to enhance the perception for user identification. For example, this is feasible to distinguish the demonstrators in case of limited number of users.

The perception to the matching between I-10/I-11 and R-10/R-11 can help people to easily discern the two instances that are from two completely different sign classes.

As mentioned before, the DTW method can be used to match a reference and a trial. This actually can help to perceive how two trajectories differ in certain properties and then provide advices to the supervised trajectory repeats/learning. For example,

assume that I-12/I-13 is a trial supposed to follow the reference of R-12/R-13, the matching results indicate that the user needs to improve the trial in the next try by paying more attention (say, to move faster) to the part of P-12 and P-13, where the denser data indicate that the speed is lower than that of the reference.

### 5.3. GMR based sign trajectory perception

The second experiment shows the GMM based cluster signature and then the GMR based sign signature perception. As shown in Fig. 19, three signature instances of the sign word '5' are

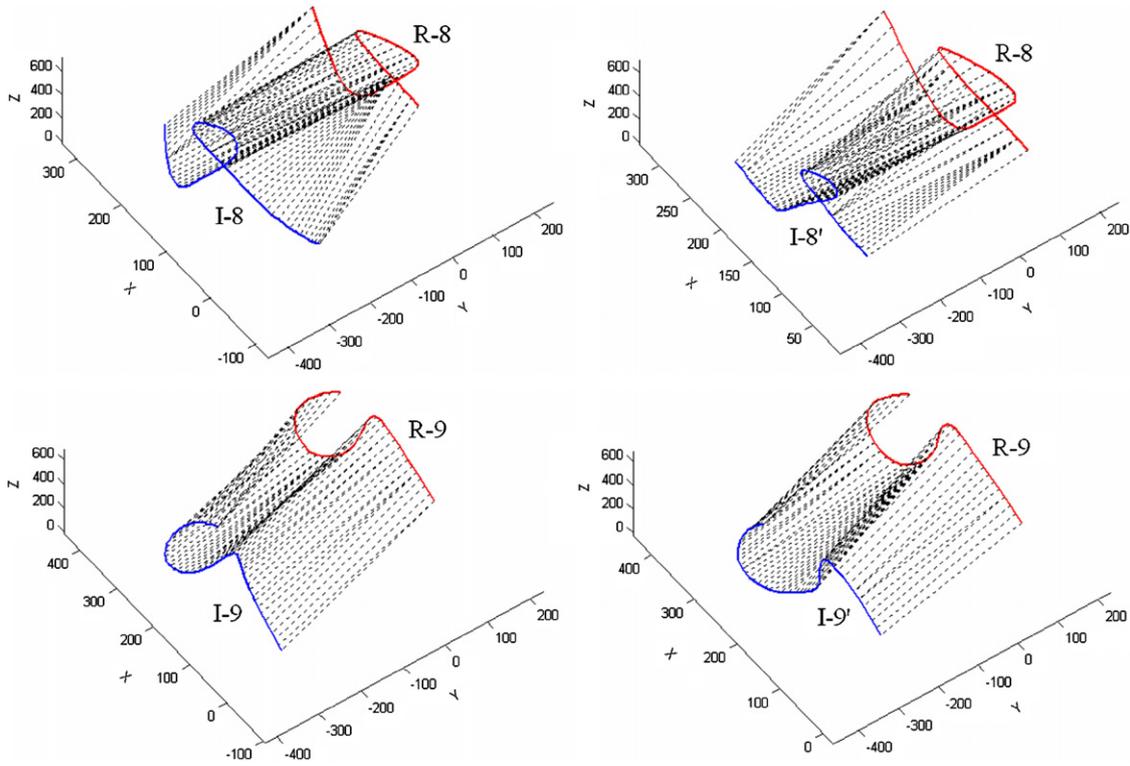


Fig. 16. Perception of various instances performed by different users.

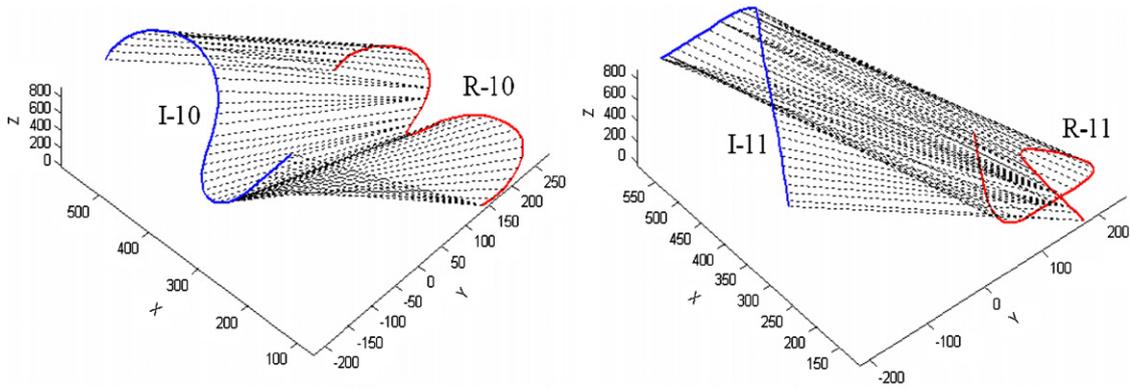


Fig. 17. Perception of completely different sign classes (words).

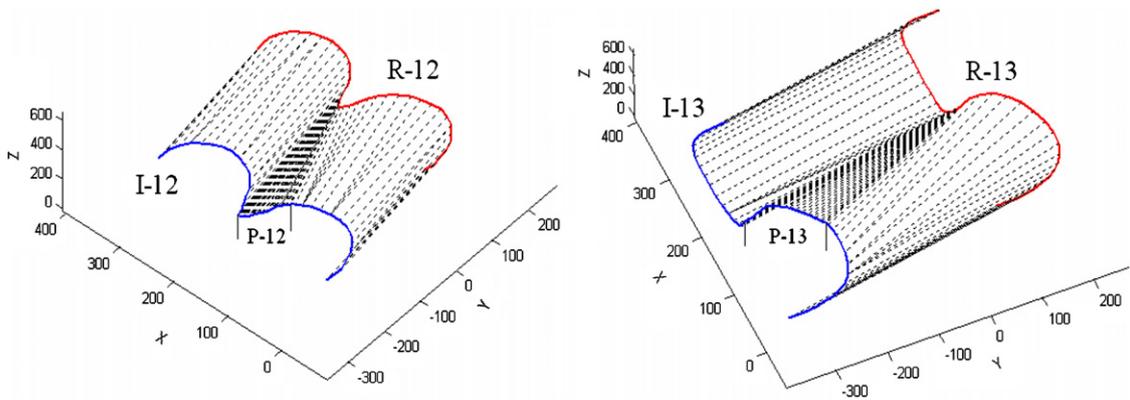
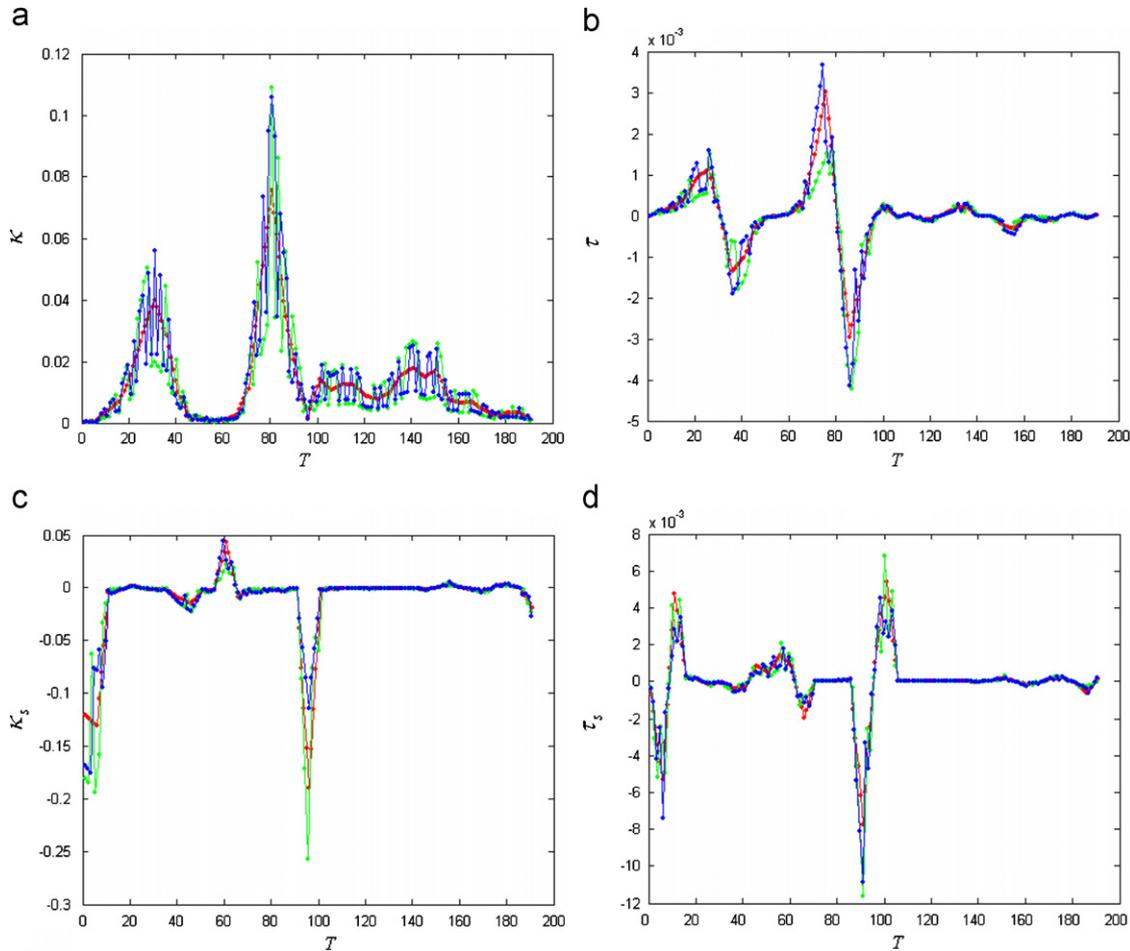


Fig. 18. From perception to advise the sign repeating/learning.



**Fig. 19.** The four signature profiles of the three signature instances plotted in red (length 191), green (length 127) and blue (length 153), respectively. (a)  $\kappa$ , (b)  $\tau$ , (c)  $\kappa_s$  and (d)  $\tau_s$ .

clustered using the full signature with different lengths. They have 191, 127 and 153 sampling points, respectively. As described in Section 2.4, their temporal indexes are firstly pre-normalized to the range of [1,191]. Then apply the EM and IPRA algorithms to the three signature instances with initial parameters of mixing Gaussian number  $K = 30$ . The stopping threshold of the EM algorithm is set to  $C_{stop} = 1e-9$  and the IPRA merging similarity threshold is set to  $H_{merge} = 0.001$ . After the EM and IPRA procedures stop, an optimal GMM model is finally learned that actually consists of 20 mixing Gaussian components. This GMM model is the cluster signature obtained from the three signature instances for the sign class '5'.

As shown in Fig. 20, the 20-component GMM is displayed in terms of the four individual signature profiles. Each mixing Gaussian model is represented by an ellipse in which the star symbol indicates the data center and the covered field of the ellipse indicates the varying range of the corresponding partial signature data being modeled. From the visualized GMM model, we get an intuitive perception to the entire cluster signature's varying range that is characterized by a sequence of ellipses (Gaussian models).

Next, we illustrate instantiating generalized cluster signatures from the GMM model by configuring different predictors to the GMR procedure. In Figs. 21 and 22, the two generalized instances of the cluster signature shown in Fig. 20 are produced by the GMR algorithm with respective predictor setting  $T = [1, 191]$  and  $[1, 63]$ . The generalized signatures are overlaid (in black) in the three signature

instances to compare its generalization degree. We can observe that the regressed signatures are good to serve as representative signatures for the sign word '5'. In fact, this kind of generalized signatures can be produced with free configurations of the temporal length and points' distribution for diverse descriptions and perceptions of a motion class.

As mentioned before, in the course of the GMR procedure, the variability of the mean of each point along the generalized signature is restricted to a generalization range that is controlled by the covariance matrix in Eq. (38). Based on that, the variability of the entire generalized signature can be perceived by visualizing the varying range around each point. As demonstrated in Fig. 23, the gray fields indicate the generalization range of the generalized signature in Fig. 22. This gives a clearer perception to the possible variability of each signature point, and then the entire generalized signature.

In addition, since the signature profiles of  $\kappa$ ,  $\tau$ ,  $\kappa_s$  and  $\tau_s$  can be visualized individually, the motion perception can actually be conducted based on an individual signature profile. For example, we can perceive a motion according to the entire curvature or torsion profile. For example, taking X axis as a reference direction, the curvature and torsion profiles describes the variations of motion direction towards Y and Z axis, respectively. This is useful for perceiving the shape features of a trajectory. In addition, it is also meaningful for the motion analysis based on the 2D projections of a 3D motion.

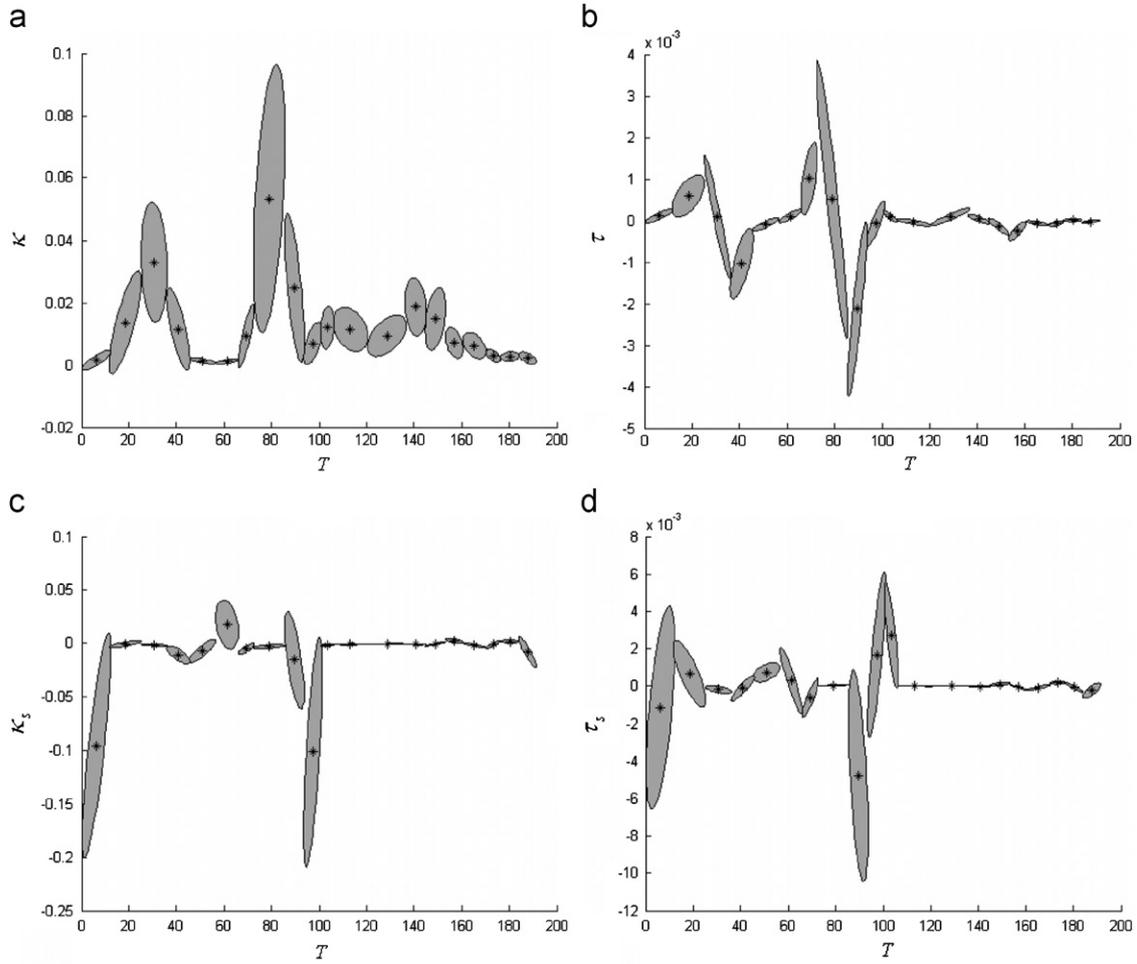


Fig. 20. The GMM based cluster signature obtained from the three signature instances shown in Fig. 19. (a)  $\kappa$ , (b)  $\tau$ , (c)  $\kappa_s$  and (d)  $\tau_s$ .

#### 5.4. Sign recognition test

In the third experiment, a larger trajectory dataset of UCI KDD high quality ASL [36] is used to test the trajectory recognition performance in terms of two important measurements: accuracy and efficiency. The ASL trajectory dataset consists of 95 sign classes (words), and 27 samples were captured for each sign. The position fields  $x$ ,  $y$  and  $z$  are extracted from the sign's feature sets to calculate trajectory signature. The length of the samples is unfixed (approximate length: 57 points). Two instances of the sign word 'hurry' and 'exit' are illustrated in Fig. 24. To reduce noise and vibration, the wavelet smoother is applied using wavelet DB5 and the third level coefficients.

According to the analysis in Section 4, we design three different signature recognition solutions: (1) full signature and DTW matching (Solution 1), (2) optimized signature and Mahalanobis distance (Solution 2), (3) cluster signature (full signature based GMM) and Bayesian recognition (Solution 3). In addition, we also compare the performances of the three solutions with an FD based solution. As reviewed in Section 1, the comparison results will be representative for a set of descriptors that use only partial features. The following Fourier transform is used to describe a 1D trajectory vector  $X(t)$ ,

$$X_f = \sum_{t=1}^M X(t) e^{-i2\pi(t-1)(f-1)/M}, \quad 1 \leq f \leq M \quad (42)$$

Applying the above transform to each dimension of a 3D motion trajectory, and setting  $\hat{f} = [0, 1, 2, 3, 4]$  as usual, we can get a Fourier feature vector  $F_l = (X_{\hat{f}}, Y_{\hat{f}}, Z_{\hat{f}})$  that describes the motion trajectory using partial coefficients. As  $F_l$  has fixed length, two Fourier representations  $F_l^i$  and  $F_l^j$  are compared directly using a metric based on the Euclidean distance  $D_F = 1/l \sum_{n=1}^l \|F_l^i - F_l^j\|$ . The FD and  $D_F$  constitute Solution 4 for the ASL trajectory recognition.

For Solution 1, of all the samples of 95 classes, half are used as training samples to learn 95 GMM models. A reference full signature is then generalized using the GMR (predictor length 50) for each class. The other half samples are input to do recognition using the DTW matching. For Solution 2, the optimized signature is condensed with 95% of the original data variance, and all samples are pre-normalized to fixed length 50 for PCA transform. Similar GMM and GMR steps are also applied to the optimized signatures to obtain the references. Solution 3 uses the 95 cluster signatures developed from the full signatures. For Solution 4, half samples of a class are used to average out a reference description, and then the other half are input to do recognition using the metric  $D_F$ .

The recognition experiment is repeated more than 50 times on a common PC (Pentium 4 CPU 3.00 GHz, 512 M RAM) by randomly picking up a number of different classes and samples, which gives rise to an average recognition performance as recorded in Tables 1 and 2. In particular, the average confusion matrices of the four solutions are generated to show the overall recognition effects of the

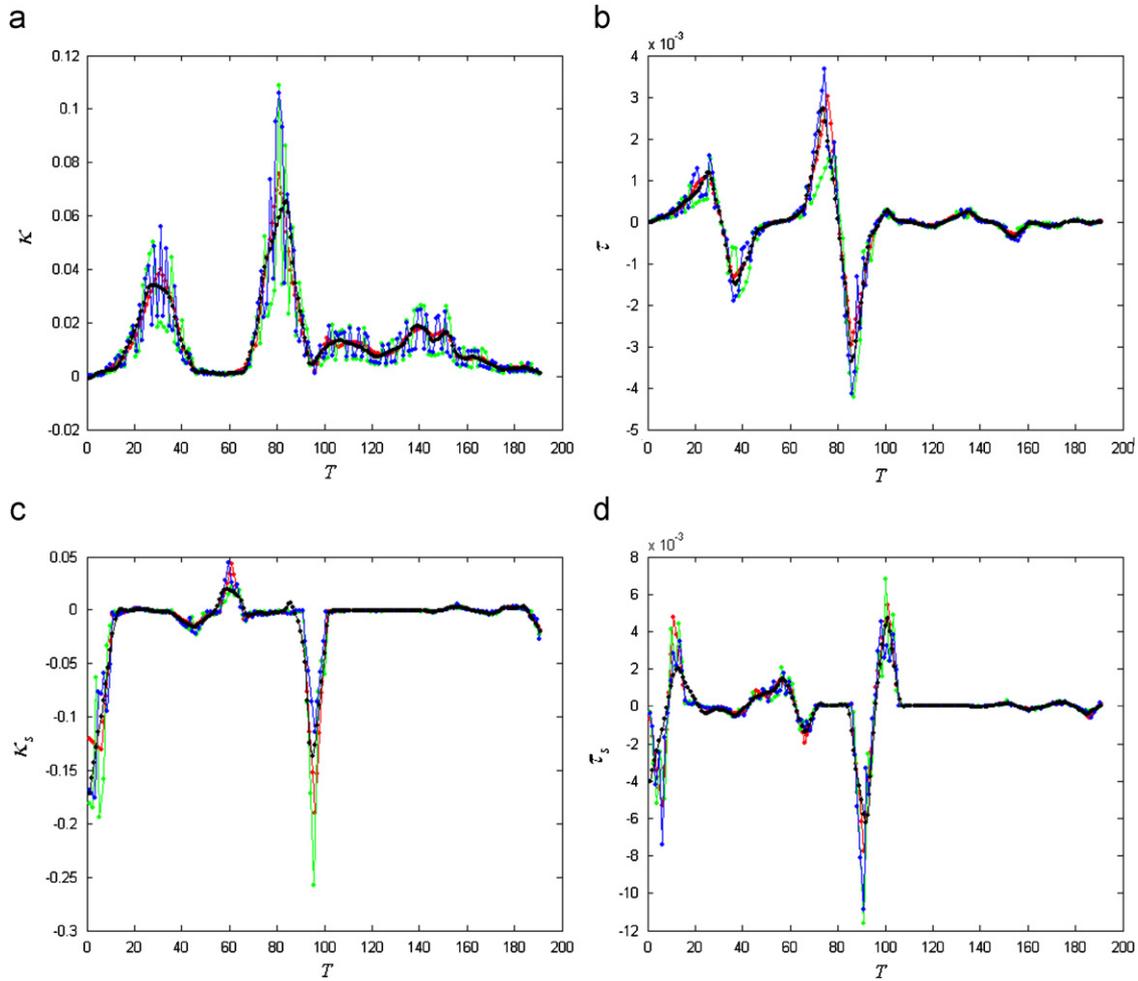


Fig. 21. Signature generalization (in black) from the cluster signature with temporal index  $T = [1, 191]$ . (a)  $\kappa$ , (b)  $\tau$ , (c)  $\kappa_s$  and (d)  $\tau_s$ .

four-class and eight-class recognition in Figs. 25 and 26, respectively. Note that the quantities of the recognition rates are scaled to intensity images, in which the scaling is with a smaller bound of  $[0, 0.2]$  for clearer display. We can find that Solutions 1 and 3 have higher recognition rate. Solutions 2 and 3 have higher efficiency (in terms of query time) suitable for potential real time trajectory retrieval. The query efficiency of Solution 1 decreases quickly as the class number increases. Solution 3 behaves generally better on both accuracy and efficiency. But it necessitates a prior training process. In contrast, Solutions 1 and 2 could start with an empty database. Meanwhile, it is observed that while the recognition efficiency of Solution 4 is comparable with Solutions 2 and 3, it has, obviously, lower recognition rates than all of the other three signature solutions. The above comparisons manifest that the proposed signature recognition solutions can have high adaptability satisfying different recognition requirements in accuracy, efficiency and necessity of training process.

## 6. Discussion and conclusion

In this section, we firstly discuss the potential application scenarios of the signature descriptor to robot learning and then conclude our work.

### 6.1. Application: enhancement to robot learning

A signature descriptor is designed for free form 3D motion trajectory characterization with wide potential applications where a

motion trajectory is concerned for representation, perception and recognition, for example, in human behavior recognition, robot learning, human–robot interaction and various motion analyses. Here we specifically emphasize the advantages of the proposed signature descriptions for enhancing robot learning by demonstration (LbD) [37,38]. Firstly, many kinds of human demonstrations can be described by the underlying motion trajectories such as the reported work in Refs. [1,2,21]. Because LbD demands systematic description to the demonstrations [39,40], the three flexible signature descriptions provide the possibility to meet different trajectory description requirements for diverse robot tasks. More importantly, the signature has an advantage in rich descriptive invariants. Note that the mentioned invariants hold in all the three signature descriptions, as the optimized signature and cluster signature are based on the full signature. The invariant signature representations actually admit generalized task representation to be learned by a robot. Hence, the signature outperforms the raw trajectory data.

Secondly, robot learning is not only supposed to acquire physical task description, but also expected to learn more about the characteristics and knowledge of the demonstrated tasks. The presented motion perception in this paper shows several aspects of the enhancements to LbD as analyzed in the following.

The perception from a single signature is helpful to enable a robot to know more task details, which could be the basis to perform

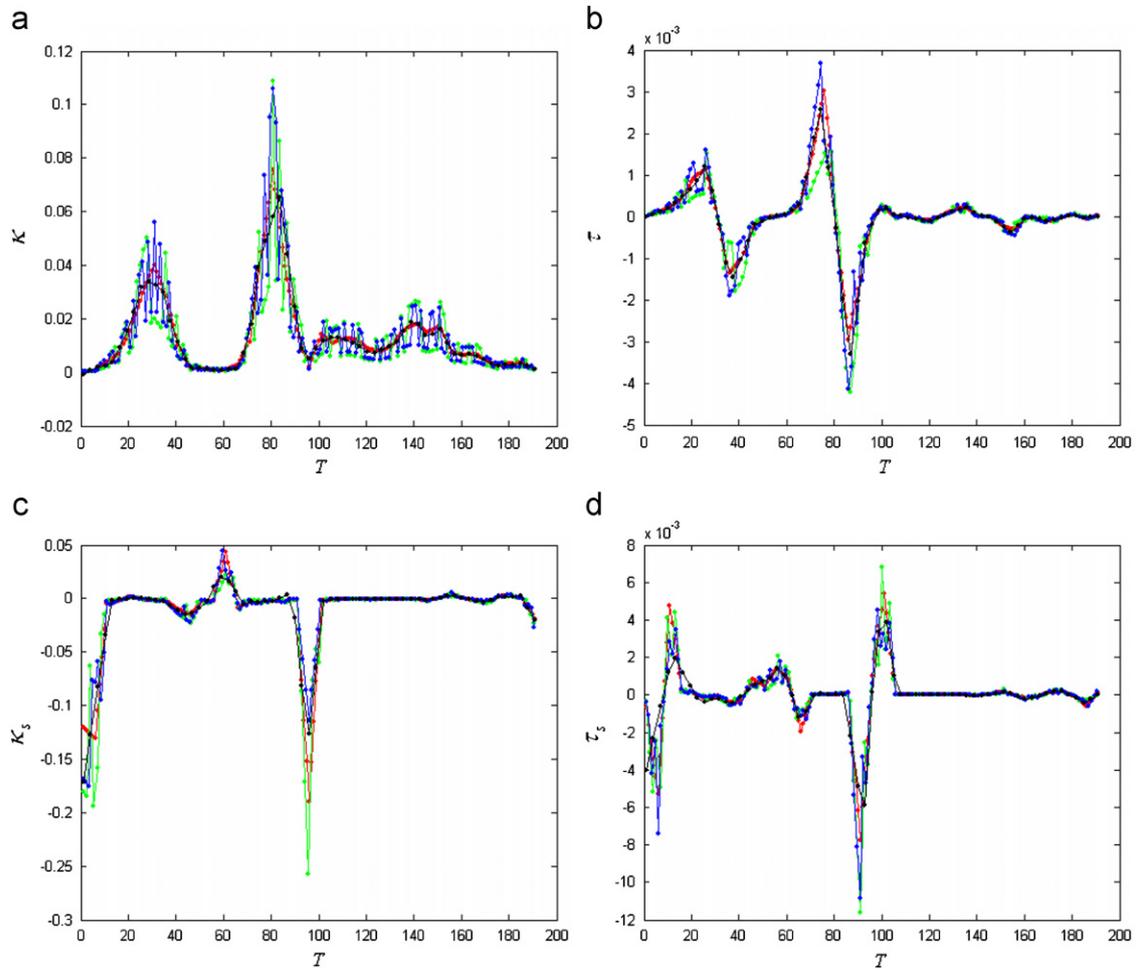


Fig. 22. Signature generalization (in black) from the cluster signature with temporal index  $T = [1, 63]$ . (a)  $\kappa$ . (b)  $\tau$ . (c)  $\kappa_s$  and (d)  $\tau_s$ .

simple task reasoning and symbolic interpretation to drive robot behaving more intelligently. The DTW matching gives an intuitive interface to indicate the aspects that a robot needs to improve in the course of LbD learning. In particular, the DTW method can serve as an appropriate metric to measure the quality quantitatively (DTW distance) and qualitatively (warping paths, refer to the example in Fig. 18) of a reproduced motion by robot. The GMR signature regression can output a generalized signature from a GMM model or a set of signature instances. In LbD, different demonstrators may produce diverse samples and then diverse signatures for the same task. The generalized signature obtained by the GMR can serve as a representative signature to be referred in motion reproduction by a robot. As shown in Section 5.3, it is flexible to generate various generalized signatures by customizing the predictors. This explains why the GMR can make a robot learn abstract task knowledge via the generalization of a cluster signature.

Thirdly, motion trajectory recognition is also helpful to enable a robot to ‘understand’ the meanings of the learned tasks. LbD should not be restricted to simple task reproduction. The recognition of learned demonstrations can give a robot higher ability and intelligence to enhance human–robot interactions during the LbD learning process.

Finally, the sign language oriented experiments show the characterization of sign motion trajectories. Sign language could be a kind of robot task to be learned via LbD. If a robot can recognize and perceive the sign language from human demonstrations, it will be

able to interact with human (or other robots) via the sign language friendlyly.

Here, we also point out that, the present work cannot serve as a full LbD framework yet at this stage. This is because in basic LbD procedures, a robot has to be able to reproduce the learned task. So far we have not reported the relevant work in reproducing trajectories from a learned signature. Developing trajectory instantiation algorithm from a given signature would be a key step to complete the LbD close-loop workflow. This is our ongoing study and we will report in the next step work.

## 6.2. Concluding remarks

To achieve adaptive motion characterization, motion trajectory is studied in this paper by building a flexible signature descriptor with its applications to motion perception and recognition. The proposed signature admits three different descriptions. The novelty of the work lies in the functional adaptability to meet diverse application requirements.

Each signature description has specific capability for trajectory representation. The full signature data are relatively redundant, but they provide an opportunity to perceive motion details. The optimized signature is more compact and its length is dynamically adjustable, only it cannot completely preserve the original motion features. The cluster signature is a model based description for a motion class, which is quite effective in describing a set of similar trajectory instances.

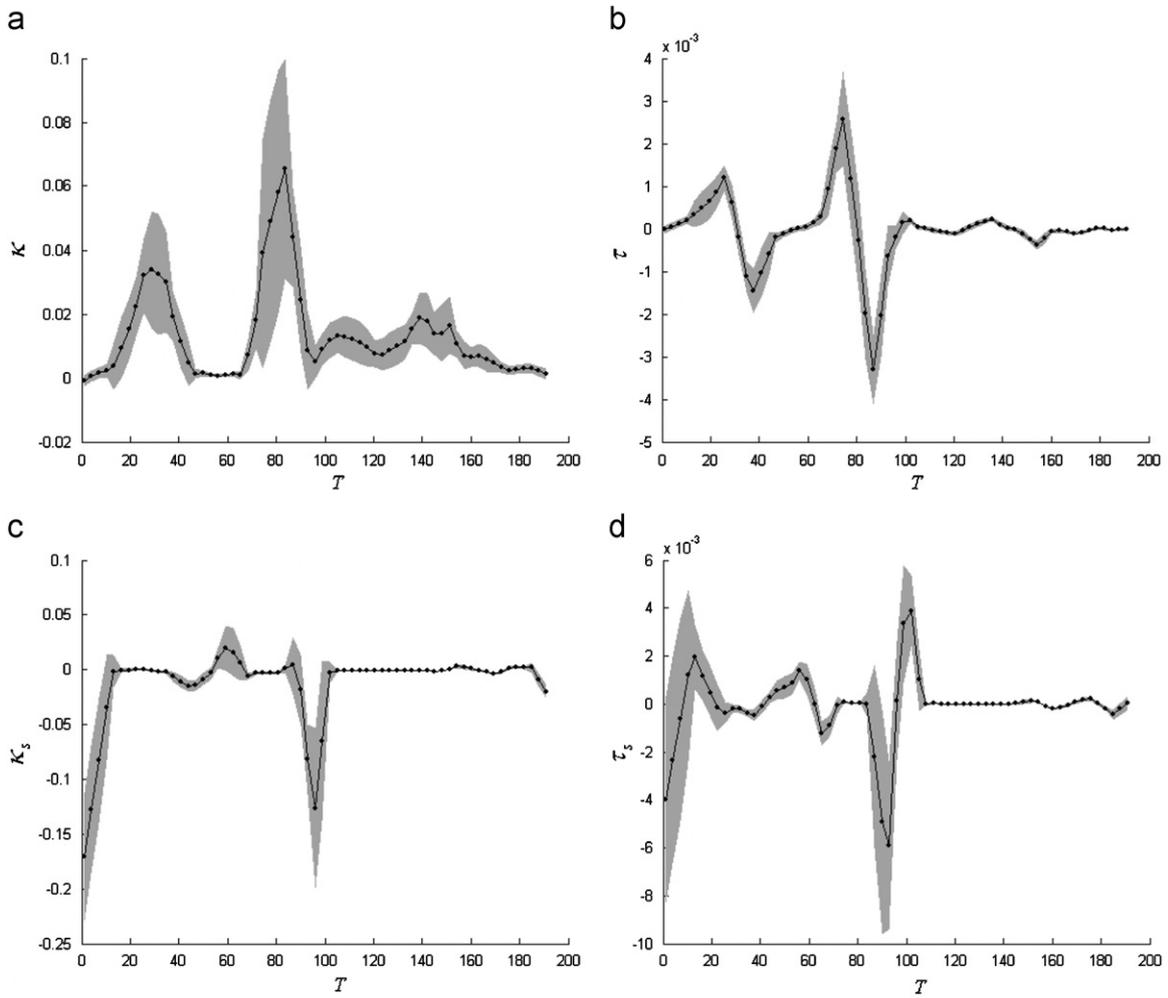


Fig. 23. Varying range (gray fields) perception to the generalized signature shown in Fig. 22. (a)  $\kappa$ , (b)  $\tau$ , (c)  $\kappa_s$  and (d)  $\tau_s$ .

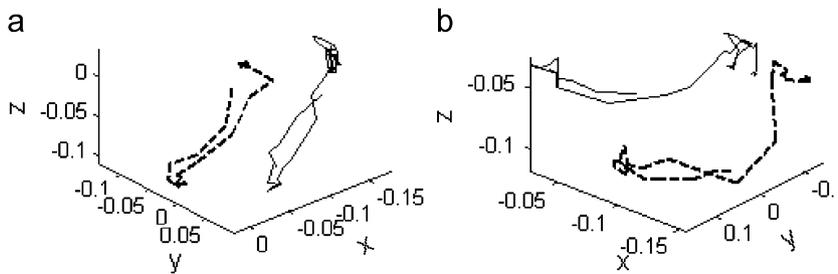


Fig. 24. Sign samples of the words 'hurry' (a) and 'exit' (b).

**Table 1**  
Recognition accuracy comparison

Solutions	Number of classes		
	2 (%)	4 (%)	8 (%)
Solution 1	92.03	87.52	81.19
Solution 2	89.12	84.39	77.27
Solution 3	92.73	88.58	83.02
Solution 4	87.98	75.74	63.85

**Table 2**  
Recognition efficiency comparison (units of milliseconds per query)

Solutions	Number of classes		
	2	4	8
Solution 1	730	1245	2286
Solution 2	121	127	138
Solution 3	163	187	191
Solution 4	145	162	179

These three flexible signature descriptions lead to adaptive applications in characterizing various motions which is verified in the three experiments conducted. The full signature is particularly good

at motion perception because it preserves complete motion features. In particular, the DTW based full signature matching provides an intuitive interface for motion perception. If high trajectory recognition

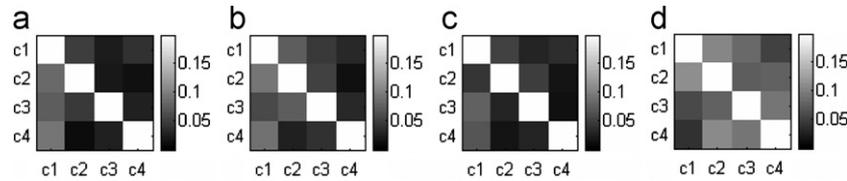


Fig. 25. The average confusion matrices of the four-class recognition (classes are indexed by from c1 to c4). (a) Solution 1, (b) Solution 2, (c) Solution 3 and (d) Solution 4.

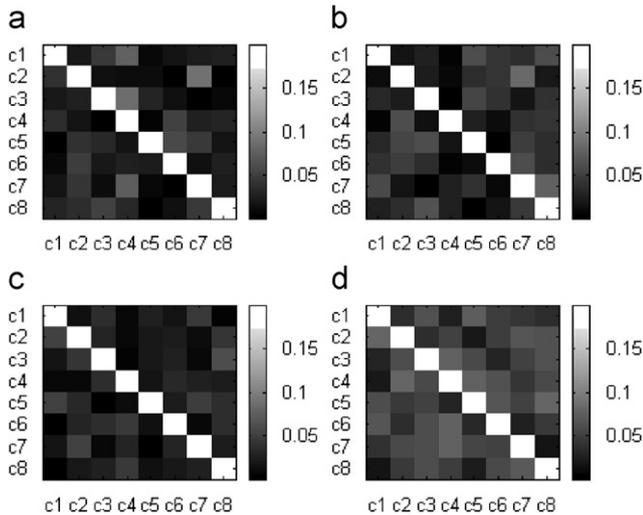


Fig. 26. The average confusion matrices of the eight-class recognition (classes are indexed by from c1 to c8). (a) Solution 1, (b) Solution 2, (c) Solution 3 and (d) Solution 4.

efficiency is specifically needed, the optimized signature can behave better than the full signature. The cluster signature has good performance at motion perception and recognition. The GMM signature visualization and GMR based generalization give an explicit interface to perceive motion patterns. The three signature recognition solutions show that adaptive trajectory recognition can be achieved by a compromise among accuracy, efficiency and the availability of training samples. In particular, note that the motion perception is an advantageous function, that many other descriptors (reviewed in Section 1) are not comparable. Hence, it can be concluded that the novel signature descriptions are flexible that can be adopted in different applications.

It is worthy to note that the signature has three descriptions, which essentially differs from the multi-resolution descriptors (FD, wavelet or CSS) in structure. The relation among the three signature descriptions is different from the hierarchy of the multi-resolution descriptors. Therefore, the signature does not suffer from the correspondence problems caused by multiple representation resolutions. One may have doubt on the optimized signature's correspondence problem due to different choices of the optimization degree in preserving data variance ( $\varphi$  in Eq. (12)). We clarify that this will not be problematic because in normal cases,  $\varphi$  is always set to a quantity close to 100% to preserve most of the signature data's variance.

The present work has established a foundation for the joint description to multiple different trajectories. Next, we will extend the signature descriptor to characterize complicated motions via analyzing multiple spatially parallel and (or) temporally continuous trajectories.

#### Acknowledgment

This work was supported by the Research Grants Council of Hong Kong under Project CityU117106.

#### References

- [1] S. Calinon, F. Guenter, A. Billard, On learning, representing and generalizing a task in a humanoid robot, *IEEE Trans. Syst. Man Cyber. part B* 36 (5) (2006) 286–298 special issue on robot learning by observation, demonstration and imitation.
- [2] A. Ude, C.G. Atkeson, M. Riley, Programming full-body movements for humanoid robots by observation, *Robotics Autonomous Syst.* 47 (2004) 93–108.
- [3] M. Bennewitz, W. Burgard, G. Cielniak, S. Thrun, Learning motion patterns of people for compliant robot motion, *Int. J. Robotics Res.* 24 (1) (2005) 31–48.
- [4] M.J. Black, A.D. Jepson, A probabilistic framework for matching temporal trajectories: condensation-based recognition of gestures and expressions, in: *Proceedings of the European Conference on Computer Vision*, Freiburg, Germany, vol. 1, 1998, pp. 909–924.
- [5] J. Martin, D. Hall, J. Crowley, Statistical gesture recognition through modeling of parameter trajectories, *Lecture Notes in Computer Science* 1739 (1999) 129–140.
- [6] M.H. Yang, N. Ahuja, Recognizing hand gestures using motion trajectories, in: *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, Fort Collins, Colorado, 1999, pp. 468–472.
- [7] S.J. McKenna, K. Morrison, A comparison of skin history and trajectory-based representation schemes for the recognition of user-specified gestures, *Pattern Recognition* 37 (5) (2004) 999–1009.
- [8] D. Meyer, J. Psl, H. Niemann, Gait classification with HMMs for trajectories of body parts extracted by Mixture densities, in: *Proceedings of British Machine Vision Conference*, Southampton, England, 1998, pp. 459–468.
- [9] J. Min, R. Kasturi, Activity recognition based on multiple motion trajectories, in: *Proceedings of the 17th International Conference on Pattern Recognition*, Cambridge, UK, 2004, vol. 4, pp. 199–202.
- [10] J. Min, R. Kasturi, Extraction and temporal segmentation of multiple motion trajectories in human motion, in: *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, Washington, DC, 2004, pp. 118–122.
- [11] W. Chen, S.F. Chang, Motion trajectory matching of video objects, in: *Proceedings of SPIE/IS&T Storage and Retrieval for Media Databases*, San Jose, CA, 2000, pp. 544–553.
- [12] C. Rao, A. Yilmaz, M. Shah, View-invariant representation and recognition of actions, *Int. J. Comput. Vision* 50 (2) (2002) 203–226.
- [13] C.B. Shim, J.W. Chang, A spatio-temporal representation scheme for modeling moving objects in video data, *Lecture Notes in Computer Science* 1961 (2000) 104–118.
- [14] V.V. Kindratenko, On using functions to describe the shape, *J. Math. Imaging Vision* 18 (33) (2003) 225–245.
- [15] E. Bribiesca, A chain code for representing 3D curves, *Pattern Recognition* 33 (5) (2000) 755–765.
- [16] P.R.G. Harding, T.J. Ellis, Recognizing hand gesture using Fourier descriptors, in: *Proceedings of the 17th International Conference on Pattern Recognition*, vol. 3, Cambridge, UK, 2004, pp. 286–289.
- [17] G. Chuang, C.C. Kuo, Wavelet descriptor of planar curves: theory and applications, *IEEE Trans. Image Process.* 5 (1) (1996) 56–70.
- [18] P. Wunsch, A.F. Laine, Wavelet descriptors for multiresolution recognition of handprinted characters, *Pattern Recognition* 28 (8) (1995) 1237–1249.
- [19] F. Mokhtarian, S. Abbasi, J. Kittler, Robust and efficient shape indexing through curvature scale space, in: *Proceedings of British Machine Vision Conference*, Edinburgh, UK, 1996, pp. 53–62.
- [20] M.K. Hu, Visual pattern recognition by moment invariants, *IEEE Trans. Inf. Theory* 8 (2) (1962) 179–187.
- [21] J. Aleotti, S. Caselli, Robust trajectory learning and approximation for robot programming by demonstration, *Robotics Autonomous Syst.* 54 (5) (2006) 409–413.
- [22] F.S. Cohen, Z. Huang, Z. Yang, Invariant matching and identification of curves using B-splines curve representation, *IEEE Trans. Image Process.* 4 (1) (1995) 1–10.
- [23] M.C. Shin, L.V. Tsap, D.B. Goldgof, Gesture recognition using Bezier curves for visualization navigation from registered 3D data, *Pattern Recognition* 37 (5) (2004) 1011–1024.
- [24] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (10) (2005) 1615–1630.
- [25] D. Zhang, G. Lu, Review of shape representation and description techniques, *Pattern Recognition* 37 (1) (2004) 1–19.
- [26] E. Calabi, P.J. Olver, C. Shakiban, A. Tannenbaum, S. Haker, Differential and numerically invariant signature curves applied to object recognition, *Int. J. Comput. Vision* 26 (2) (1998) 107–135.

- [27] M. Boutin, Numerically invariant signature curves, *Int. J. Comput. Vision* 40 (3) (2000) 235–248.
- [28] I. Jolliffe, *Principal Component Analysis*, Springer, New York, 1986.
- [29] J. Yang, D. Zhang, J.Y. Yang, A.F. Fragni, Two-dimensional PCA: a new approach to appearance-based face representation and recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* 26 (1) (2004) 131–137.
- [30] J.A. Bilmes, A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models, *Int. Comput. Sci. Inst. TR-97-021*
- [31] H.G. Sung, *Gaussian mixture regression and classification*, Ph.D. Dissertation, Rice University, 2004.
- [32] M. Vlachos, G. Kollios, D. Gunopulos, Elastic translation invariant matching of trajectories, *Mach. Learn.* 58 (2–3) (2005) 301–334.
- [33] A. Guézic, N. Ayache, Smoothing and matching of 3d space curves, *Int. J. Comput. Vision* 12 (1) (1994) 79–104.
- [34] L.J. Latecki, V. Megalooikonomou, Q. Wang, R. Lakaemper, C.A. Ratanamahatana, E. Keogh, Elastic partial matching of time series, in: *Proceedings of European Conference on Principles and Practice in Knowledge Discovery in Databases*, Porto, Portugal, 2005, pp. 577–584.
- [35] G.R. Bradski, Computer vision face tracking as a component of a perceptual user interface, in: *Proceedings of Workshop on Applications of Computer Vision*, Princeton, NJ, 1998, pp. 214–219.
- [36] UCI KDD ASL Archive, [Online], Available: (<http://kdd.ics.uci.edu/databases/auslan2/auslan.html>).
- [37] K. Ikeuchi, T. Suehiro, Toward an assembly plan from observation part I: task recognition with polyhedral objects, *IEEE Trans. Robotics Automation* 10 (3) (1994) 368–385.
- [38] H. Friedrich, S. Munch, R. Dillmann, S. Bocionek, M. Sassin, Robot programming by demonstration (RPD): supporting the induction by human interaction, *Mach. Learn.* 23 (2–3) (1996) 163–189.
- [39] C.A. Acosta Calderon, H. Hu, Robot imitation from human body movements, in: *Proceedings of the 3rd International Symposium on Imitation in Animals and Artifacts, AISB'05 Convention*, Hatfield, England, 2005, pp. 1–9.
- [40] J. Pisokas, D. Gu, H. Hu, Learning to plan for robots using generalized representations, *Int. J. Ind. Robot* 33 (4) (2006) 270–277.

**About the Author**—SHANDONG WU received the B.Sc. and M.Sc. degrees both in Computer Science from School of Computer Engineering and Science at Shanghai University, Shanghai, China, in 1998 and 2003, respectively. He is currently a Ph.D. student at Department of Manufacturing Engineering and Engineering Management at City University of Hong Kong, Hong Kong. His main research area consists of computer vision and robotics particularly human behavior description and recognition and robot learning by demonstration.

**About the Author**—Y.F. LI received the Ph.D. degree in Robotics from Oxford University, Oxford, UK, in 1993. From 1993 to 1995, he worked as a Postdoctoral Researcher in Department of Computer Science at University of Wales, Aberystwyth, UK. He joined City University of Hong Kong in 1995. He has published over 100 papers in refereed international journals and conferences. His research interests include robot vision, visual tracking, robot sensing and sensor-based control, mechatronics, and automation. He is a senior member of the IEEE. Dr. Li is an Associate Editor of IEEE Transactions on Automation Science and Engineering.