2011 11th International Conference on Control, Automation and Systems
Oct. 26-29, 2011 in KINTEX, Gyeonggi-do, Korea

Kinesthetic Learning of Behaviors in a Humanoid Robot

Sumin Cho and Sungho Jo

Department of Computer Science, KAIST, Daejeon, 305-701, Korea
(Tel : +82-42-350-3540; E-mail: {jerachiel7, shjo}@kaist.ac.kr)

Abstract: This work presents an approach for learning of behaviors by kinesthetic teaching in a humanoid robot. The approach enables the robot to improve and reproduce a specific behavior incrementally every time a new teaching trial is provided, and therefore, it is more suitable for real-world human-robot interaction. The algorithm consists of projection of motion data to a latent space and description of motion data in a Gaussian Mixture Model (GMM). The latent space and GMM can be refined incrementally after each kinesthetic teaching. The number of components in the GMM is adjusted accordingly in a real-time manner. Experiments with a Nao humanoid robot show the feasibility of the approach. We demonstrate that the robot can reproduce learned behaviors well through continuous kinesthetic trials.

Keywords: Incremental learning, Humanoid, GMM.

1. INTRODUCTION

Teaching a humanoid robot through demonstrations, also referred to as Learning by imitation, is a useful approach to allow the robot to conveniently learn new behavioral skills. A humanoid robot observes multiple demonstration of a same meaningful behavior or is taught kinesthetically over trials, and reproduces a generalized motion [1]-[8]. There are several advantages of this approach. First, it is a method that uses user-friendly interaction. A human can interact with a robot as he or she does with another human. Second, it facilitates acquisition of new skills. A human does not need to program new skills every time. Third, it can provide an efficient method to store learned behaviors. Instead of the detailed description of each behavior, the approach makes it possible to have various behaviors in compact information descriptions and to reproduce them easily whenever necessary. Fourth, it generalizes a specific behavior and therefore, the behavior can be easily adapted to similar tasks.

To be more adequate and applicable in a real situation of human-robot interaction, teaching methods for skillful behaviors should be dynamic and real-time. To simulate a real-world situation, teaching should be performed sequentially, and the robot should start learning a behavior from the first teaching. As demonstrations of the same task are subsequently provided, the robot should be able to refine the learned skillful behavior.

This paper describes an incremental approach for learning of behaviors by a humanoid robot kinesthetically. A robot encodes motion information while teacher moves a robot's limbs to execute desired motion. A learning method based on Principal Component Analysis (PCA) and Gaussian Mixture Model (GMM) to build a probabilistic representation of behaviors has been presented [1]-[3]. We improve upon the approach by reflecting upcoming data information directly on a latent space of motion and the GMM. By using this proposed approach, a robot modifies the latent space of motion steadily and refines progressively an appropriate number of Gaussian mixture components in the GMM whenever a new teaching is performed. The robot does not need to memorize all the data points explicitly. Instead, behavioral improvement is promptly realized through the update of the GMM.

This work was supported in part by the Korea government (MEST) under the KRF grant (No. 2010-0015226) and in part by the Korea government (MKE) under Human Resources Development Program for Convergence Robot Specialists.

2. ALGORITHM

This work uses GMM as model descriptor. GMM has been a popular approach to model a probabilistic density function to represent a group of similar motion trajectories [1][2]. The probabilistic density function can be represented as:

$$p(x_i) = \sum_{k=1}^{K} \pi_k p(x_i|k) \quad (1)$$

where

$p(x_i|k) = N(x_i; \mu_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^d|\Sigma_k|}} \exp\left(-\frac{1}{2}(x_i - \mu_k)^T {\Sigma_k}^{-1}(x_i - \mu_k)\right)$, π_k is the prior probability, and μ_k and Σ_k are respectively the mean and the covariance matrix of the k-th component.

The model parameters are π_k, μ_k, and Σ_k for k=1,2,…,K. The density function is defined by K Gaussian components. Using the GMM, our behavior learning algorithm procedure can be partitioned as in Fig 1. Each step is described in this section.

2.1 Teaching trial data encoding

A human teaches a robot kinesthetically by moving the robot's limbs. During each teaching trial, the motion is encoded in terms of 3D joint angle trajectories. The

978-89-93215-03-8 98560/11/$15 ⓒICROS

collected joint angle trajectory information is represented by $\{\xi_i^j\}_{i=1}^N$ where N^j is the total number of data points from the j-th teaching trial. Each ξ_i^j consists of a time value and a posture vector.

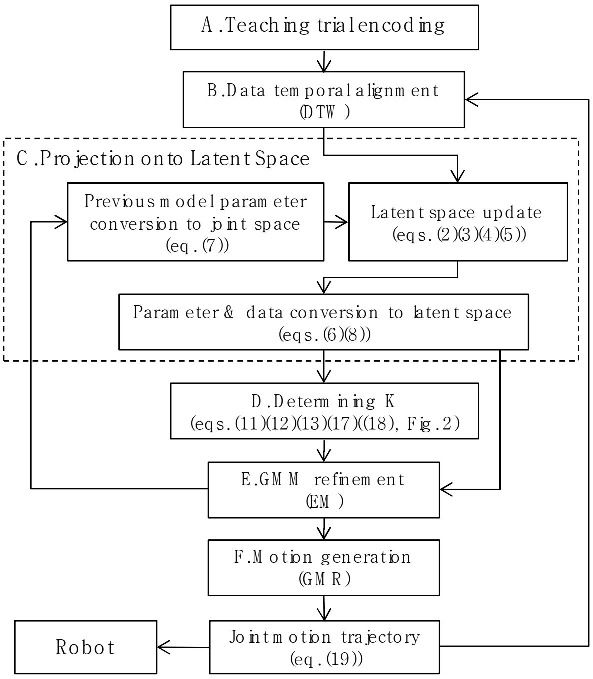

Fig 1. Overall algorithm procedure.

2.2 Temporal alignment of data

Each motion of the same task is similar but not exactly identical. Its duration and speed may be different from the previous ones. Therefore, motions are sampled in different length ($N^j \neq N^i$) by the sensors. To enable the motion data to be comparable, they should be aligned temporally. We apply dynamic time warping (DTW) to solve the problem [4][9]. Among several DTW versions, the step pattern is selected to alleviate the problem of singularities [10]. As a result, any j-th teaching trial is represented with the same number of data points N by $\{\vartheta_i^j\}_{i=1}^N$ where N is the total number of aligned data point. Each ϑ_i^j also consists of a time value $t_i^j \in R$ and a posture vector $\theta_i^j \in R^d$ where d is the dimensionality of the joint space. The temporal alignment is skipped for the first teaching trial. From the second trial, the reproduced trajectory (from (19)) in the previous trial plays a role of matching template.

2.3 Projection onto latent space

It is inefficient to manage complicated whole body motions in high dimensional joint angle space. Describing motions in a latent space of reduced dimensionality with uncorrelated components has already been commonly used in many works [2][11][12] due to its usefulness and simplicity. PCA is a popular method to obtain the latent space of reduced dimensionality. We update the mean vector and covariance matrix promptly whenever a new trial is performed. After the j-th trial, the mean m_θ^j and covariance matrix Σ_θ^j in the joint space are computed recursively as follows: j=1,2,3,…

$$m_\theta^j = \frac{(j-1)Nm_\theta^{j-1} + \sum_{i=1}^N \theta_i^j}{jN} \quad (2)$$

$$\Sigma_\theta^j = \frac{(j-1)N\left(\Sigma_\theta^{j-1} + \left(m_\theta^{j-1} - m_\theta^j\right)\left(m_\theta^{j-1} - m_\theta^j\right)^T\right)}{jN} + \frac{\sum_{i=1}^N \left(\theta_i^j - m_\theta^j\right)\left(\theta_i^j - m_\theta^j\right)^T}{jN} \quad (3)$$

where m_θ^0 is the zero vector and Σ_θ^0 is the zero matrix.

According to PCA, given Σ_θ^j, eignvectors and eigenvalues λ_i are calculated through the combination of the power and the deflation methods [13] taking into account the real symmetric covariance matrix.

The minimal number q of eigenvectors to construct the reduced dimensional latent space while sufficiently representing the original dataset is determined by $\frac{\sum_{i=1}^q \lambda_i}{\sum_{i=1}^d \lambda_i} > c$, which indicates that the projection to the latent space covers 100×c % of the data's spread. The q governs the dimensionality of the latent space. Let the projection to the latent space be represented by $V_{d\times q}^j$. Then, the projection matrix including temporal information is defined as

$$W^j = \begin{bmatrix} 1 & 0 \\ 0 & V_{d\times q}^j \end{bmatrix} \quad (5)$$

Let x^j be effective data points which describe the original dataset sufficiently. Then, it can be obtained by

$$x^j = {W^j}^T\left(\vartheta^j - m_\vartheta^j\right) \quad (6)$$

where $m_\vartheta^j = \left[0\ m_\theta^j\right]^T$.

Our GMM is described in the latent space. Since the latent space is updated, the model information from previous teaching trial should be projected to the new latent space as follows. The model information is projected back to the joint space.

$$\mu_{\vartheta,k}^{j-1} = W^{j-1}\mu_{x,k}^{j-1} + m_\vartheta^{j-1},\ \Sigma_{\vartheta,k}^{j-1} = W^{j-1}\Sigma_{x,k}^{j-1}{W^{j-1}}^T \quad (7)$$

where $\mu_{x,k}^{j-1}$ and $\Sigma_{x,k}^{j-1}$ are respectively the mean and the covariance matrix of the k-th Gaussian component in the GMM, computed before the j-th trial.

The mean and the covariance matrix in the joint space are transformed into the new latent space where the j-th trial information is taken into account respectively by

$$\mu_{x,k}^{j-1} = {W^j}^T\left(\mu_{\vartheta,k}^{j-1} - m_\vartheta^j\right),\ \Sigma_{x,k}^{j-1} = {W^j}^T\Sigma_{\vartheta,k}^{j-1}W^j \quad (8)$$

As a result, the model parameters $\mu_{x,k}^{j-1}$ and $\Sigma_{x,k}^{j-1}$ are with respect to the recently updated latent space in j-th trial.

2.4 Selecting the number of components in the model

The number of Gaussian components in the GMM in the latent space, K, is initially unknown. If all motion data are available at once, an optimal number of components K can be selected by minimizing the Bayesian information criterion (BIC) score S_{BIC}, which evaluates both model performance and complexity [1][2].

$$S_{BIC} = (\text{model fitting term}) + (\text{complexity term}) = -\mathcal{L} + \frac{n_p}{2}\log(N_{tot}) \quad (9)$$

where n_p is the total number of model parameters, N_{tot} the total number of datapoints, and $\mathcal{L} = \sum_{i=1}^{N}\log\left(p(x_i)\right) = \sum_{i=1}^{N}\log\left(\sum_{k=1}^{K}p(k)p(x_i^j|k)\right)$ is the log-likelihood of the model.

In our model, the number of model parameters is determined by K and q as follows:

$$n_p = (K-1) + K\left(q + 1 + \frac{1}{2}(q+1)(q+2)\right) \quad (10)$$

We are interested in a situation where the whole data are not available at once. An approximation of the BIC score based on the available data information is proposed to select K. Our method is inspired by the fact that each Gaussian density component is dominant over partial data points. The partial data points are linearly spread, locally in the latent space, in the neighbor of the mean of the Gaussian component. A remarkable breakpoint of the local characterization can be regarded as a starting point where another Gaussian component becomes more influential.

Virtual data points are sampled from a current model to reflect information from past data before the j-th teaching trial is included. α virtual data trajectories are generated where α is a positive integer greater than 1. The sampled data points are combined with the j-th trial data points to temporarily construct data points for the j-th learning. The data is described by $X^j = \{x_1^j, x_2^j, \dots, x_{N_D}^j; w_1^j, w_2^j, \dots, w_{N_D}^j\}$ where $\{x_1^j, x_2^j, \dots, x_N^j\}$ represents data from the j-th trial, and $\{x_{N+1}^j, x_{N+2}^j, \dots, x_{N_D}^j\}$ is sampled data. $N_D = (\alpha+1)N$.

Since there is no previous model in the first trial, we sample virtual data points from $N(0, \Sigma_0)$ where Σ_0 is a diagonal matrix and assign uniformly small weights.

Each weight w_i^j is assigned to be 1 for j-th data and $\frac{j-1}{\alpha}$ for sampled data. When K components are regarded, X^j can be grouped to be $X^j = \{Y_1^j, Y_2^j, \dots, Y_K^j\}$ where each Y_k^j presents data and corresponding weights associated with the k-th Gaussian component.

Two temporally sequential data points in X^j are initially grouped to avoid zero variances of each group along temporal axis. Therefore, an initial value of K is $\frac{N}{2}$.

When $y_{k,i}^j$ represents a data point in Y_k^j, $w_{k,i}^j$ its corresponding weight, L_k^j the number of data in Y_k^j, $\left|Y_k^j\right| = \sum_{i=1}^{L_k^j} w_{k,i}^j$ and $\sum_{k=1}^{K}\left|Y_k^j\right| = jN$, parameters to describe each component are computed as $G = \{\pi_k^j, \mu_k^j, \Sigma_k^j\}_{k=1}^{K}$:

$$\pi_k^j = \frac{\left|Y_k^j\right|}{jN} \quad (11)$$

$$\mu_k^j = \frac{\sum_{i=1}^{L_k^j} w_{k,i}^j y_{k,i}^j}{\left|Y_k^j\right|} \quad (12)$$

$$\Sigma_k^j = \frac{\sum_{i=1}^{L_k^j} w_{k,i}^j\left(y_{k,i}^j - \mu_k^j\right)\left(y_{k,i}^j - \mu_k^j\right)^T}{\left|Y_k^j\right|} \quad (13)$$

Once grouping is implemented, we compute an approximated estimation of $\mathcal{L}$ by considering the expected value of $\mathcal{L}$.

$$E[\mathcal{L}] = E\left[\sum_{i=1}^{N_2}\log\left(\sum_{k=1}^{K}\pi_k^j p(x_i^j|k)\right)\right] \approx \sum_{k=1}^{K}\sum_{i=1}^{L_k^j} E\left[\log\left(\pi_k^j p(y_{k,i}^j\,|k)\right)\right] \quad (14)$$

where it is assumed that $p(y_{k,i}^j\,|k) \gg p(y_{k,i}^j\,|/k)$, $/k \in \{1,2,\dots,K\}$ indicates any other number but k. $\sum_{k=1}^{K} L_k = N_D$.

$$\sum_{k=1}^{K}\sum_{i=1}^{L_k^j} E\left[\log\left(\pi_k^j p(y_{k,i}^j\,|k)\right)\right] = \sum_{k=1}^{K}\sum_{i=1}^{L_k^j} E\left[\left(-\frac{1}{2}\left(y_{k,i}^j - \mu_k^j\right)^T {\Sigma_k^j}^{-1}\left(y_{k,i}^j - \mu_k^j\right)\right)\right] + \sum_{k=1}^{K} L_k^j\left(\log(\pi_k^j) - \frac{1}{2}q\log(2\pi) - \frac{1}{2}\log\left(\left|\Sigma_k^j\right|\right)\right) \quad (15)$$

because

$$\sum_{i=1}^{L_k^j} E\left[\left(-\frac{1}{2}\left(y_{k,i}^j - \mu_k^j\right)^T {\Sigma_k^j}^{-1}\left(y_{k,i}^j - \mu_k^j\right)\right)\right] = -\frac{1}{2}qL_k^j$$

Therefore, approximately,

$$E[\mathcal{L}] = \sum_{k=1}^{K} L_k^j\left(\log(\pi_k^j) - \frac{1}{2}q(1 + \log(2\pi)) - \frac{1}{2}\log\left(\left|\Sigma_k^j\right|\right)\right) \quad (16)$$

Suppose that two k-th and (k+1)-th components are merged into a component (newly k-th component). Let log-likelihood estimations be $\mathcal{L}_k$ and $\mathcal{L}_{k+1}$ respectively before merging and $\mathcal{L}'_k$ after merging. Then, the change of the fitting term value is computed as follows:

$$\begin{aligned}
E[\Delta\mathcal{L}] &= E[\mathcal{L}'_k - (\mathcal{L}_k + \mathcal{L}_{k+1})] \\
&= L'^j_k\left(\log\left(\pi'^j_k\right) - \frac{1}{2}q(1 + \log(2\pi)) - \frac{1}{2}\log\left(\left|\Sigma'^j_k\right|\right)\right) \\
&\quad - L_k^j\left(\log(\pi_k^j) - \frac{1}{2}q(1 + \log(2\pi)) - \frac{1}{2}\log\left(\left|\Sigma_k^j\right|\right)\right) \\
&\quad - L_{k+1}^j\left(\log(\pi_{k+1}^j) - \frac{1}{2}q(1 + \log(2\pi)) - \frac{1}{2}\log\left(\left|\Sigma_{k+1}^j\right|\right)\right) \\
&= \log\frac{{\pi'^j_k}^{L'^j_k}}{{\pi_k^j}^{L_k^j}{\pi_{k+1}^j}^{L_{k+1}^j}} - \frac{1}{2}\log\frac{|\Sigma'^j_k|^{L'^j_k}}{|\Sigma_k^j|^{L_k^j}|\Sigma_{k+1}^j|^{L_{k+1}^j}}
\end{aligned} \quad (17)$$

where $G = \{\pi'^j_k, \mu'^j_k, \Sigma'^j_k\}_{k=1}^{K-1}$ after merging, and $L'_k = L_k + L_{k+1}$.

The change of the complexity term value is easily computed from (10):

$$\Delta n_p = -q - 2 - \frac{1}{2}(q+1)(q+2) \quad (18)$$

All possible mergings of two sequential groups are considered to compute the largest reduction of $\Delta S_{BIC} = -\Delta\mathcal{L} + \Delta n_p$. For the case of a merging that results largest reduction, the merging is confirmed if the sign of ΔS_{BIC} is negative. This process is repeated until the sign is negative no longer. The number of remaining Gaussian components is assigned as K after termination. The procedure is summarized in Fig. 2.

2.5 GMM refinement

Once K is selected, the expectation-maximization (EM) algorithm can be applied to revise the model parameters. The EM is implemented as follows.

0. Initialize the model parameters (Fig 2).
1. E – Step : $p_{k,i}^{t+1} = \frac{\pi_k^t N(x_i^j;\mu_k^t,\Sigma_k^t)}{\sum_{l=1}^{K} \pi_l^t N(x_i^j;\mu_k^t,\Sigma_k^t)}$
2. M – Step :

$$\pi_k^{t+1} = \frac{\sum_{i=1}^{N} w_i^j p_{k,i}^{t+1}}{jN}$$

$$\mu_k^{t+1} = \frac{\sum_{i=1}^{N} w_i^j p_{k,i}^{t+1} x_i^j}{jN\pi_k^{t+1}}$$

$$\Sigma_k^{t+1} = \frac{\sum_{i=1}^{N} w_i^j p_{k,i}^{t+1}\left(x_i^j-\mu_k^{t+1}\right)\left(x_i^j-\mu_k^{t+1}\right)^T}{jN\pi_k^{t+1}}$$

where t is the iteration number.

Repeat 1 and 2 until convergence $\left|\frac{\mathcal{L}^{t+1}}{\mathcal{L}^t} - 1\right| < C_2 \ll 1$ where $\mathcal{L} = \sum_{i=1}^{N} w_i^j \log\left(p(x_i^j)\right)$.

2.6 Motion generation

Once the parameters of the GMM are computed, a most likely generalized motion from trials so far can be reproduced by regression. We adopt a well-defined method called the Gaussian Mixture Regression (GMR) proposed in [1]. Motion trajectories in the latent space $\hat{x}^j$ are retrieved and transformed to joint trajectories $\hat{\vartheta}^j$ by projection.

$$\hat{\vartheta}^j = W^j\hat{x}^j + m_\vartheta^j \quad (19)$$

3. EXPERIMENTS

We use an Aldebaran Nao humanoid robot with 25 degrees of freedom (DOFs) to verify our algorithm. While a human teacher moves each limb to generate expected behaviors, the trajectories of each joint angle values are recorded at a rate of 20 Hz through motor encoders. In experiments, the 14 DOFs of the upper torso are used while the lower body stands still.

Six gestures of Korean martial art, Taewondo, are selected as test examples of rich gesture vocabulary. 10 teaching trials are provided for each gesture. In Fig. 3(a), the plots illustrate trajectories in the latent space during the learning of the selected gestures. Fig. 3(b) shows snapshots of gestures reproduced by the robot. Joint trajectories are generated by transforming regressed trajectories in the latent space to the joint space. The change of the dimensionality of the latent space and the number of Gaussian components over trials are presented in Table 1. A more complicated gesture tends to be described with a larger number of Gaussian components in a higher dimensional latent space. As the learning is processed, the number of Gaussian components tends to decrease. The joint space of 14 DOFs is reduced to the latent space of 4 to 6 dimensions after final teaching (c=0.99). The results demonstrate that the robot can reproduce the essential characteristics of behaviors well.

Table 1. Change of the number of Gaussian components (K) and the dimensionality of the latent space (q) over trials during learning of the Taekwondo gestures (K/q).

Trial No.	1	2	3	4	5	6	7	8	9	10
Gesture 1	10/3	8/3	6/3	7/4	7/4	6/4	6/4	6/4	6/4	6/4
Gesture 2	9/4	10/4	10/4	10/4	9/4	9/4	9/4	9/4	9/4	9/4
Gesture 3	8/2	7/2	6/2	5/2	5/3	4/3	4/3	5/3	5/4	4/4
Gesture 4	10/4	10/5	10/5	9/6	9/6	9/6	9/6	9/6	9/6	9/6
Gesture 5	9/4	9/4	10/5	9/5	9/4	9/5	9/5	9/5	9/5	9/5
Gesture 6	10/4	10/5	9/5	9/5	9/5	9/5	9/5	8/5	8/5	8/5

4. CONCLUSION

This work presents an incremental approach to implement Kinesthetic learning of behaviors. The experimental results verify the feasibility of the proposed method. The learning strategy has potential aspects. First, a teacher can refine his/her teaching depending on robot's performance. Second, a robot can make use of a learned behavior to learn quickly a new behavior which is similar to the learned. Third, a robot may recognize bad teaching trials by itself once it has learned a behavior at a certain level. Therefore, the operation may be advantageous for practical human-robot interaction.

REFERENCES

[1] S. Calinon, and A. Billard, "Incremental learning of gestures by imitation in a humanoid robot", in *Proc. IEEE Int. Conf. HRI*, Arlington, Virginia, USA, 2007, pp. 255-262.

[2] S. Calinon, F. Guenter, and A. Billard, "On learning, representing and generalizing a task in a humanoid robot", *IEEE Trans. Systems, Man and Cybernetics, Part B*, vol. 37, no. 2, pp. 286-298, 2007.

[3] M. Hersch, F. Guenter, S.Calinon, and A. Billard, "Dynamic system modulation for robot learning via kinesthetic demonstrations", *IEEE Trans. Robotics*, vol. 24, no. 6, pp. 1463-1467, 2008.

[4] A. Coates, P. Abbeel, and A.Y. Ng, "Learning for control from multiple demonstrations", in *Proc. 25th Int. Conf. Machine Learning*, Helsinki, Finland, 2008.

[5] D. B. Grimes, D. R. Rashid, and R. P. N. Rao, "Learning nonparametric models for probabilistic imitation", in *Proc. Adv Neural Inf Process Syst* , vo. 19, 2006, pp. 521–528.

[6] A. P. Shon, D. Verma, and R. P. N. Rao, "Active imitation learning", in *Proc. AAAI*, 2007, pp. 756-762.

[7] A.J. Ijspeert, J. Nakanishi, and S. Schaal, "Trajectory formation for imitation with nonlinear dynamical systems", in *Proc . IEEE Int. Conf. Intelligent Robots and Systems*, 2001, pp.752-757.

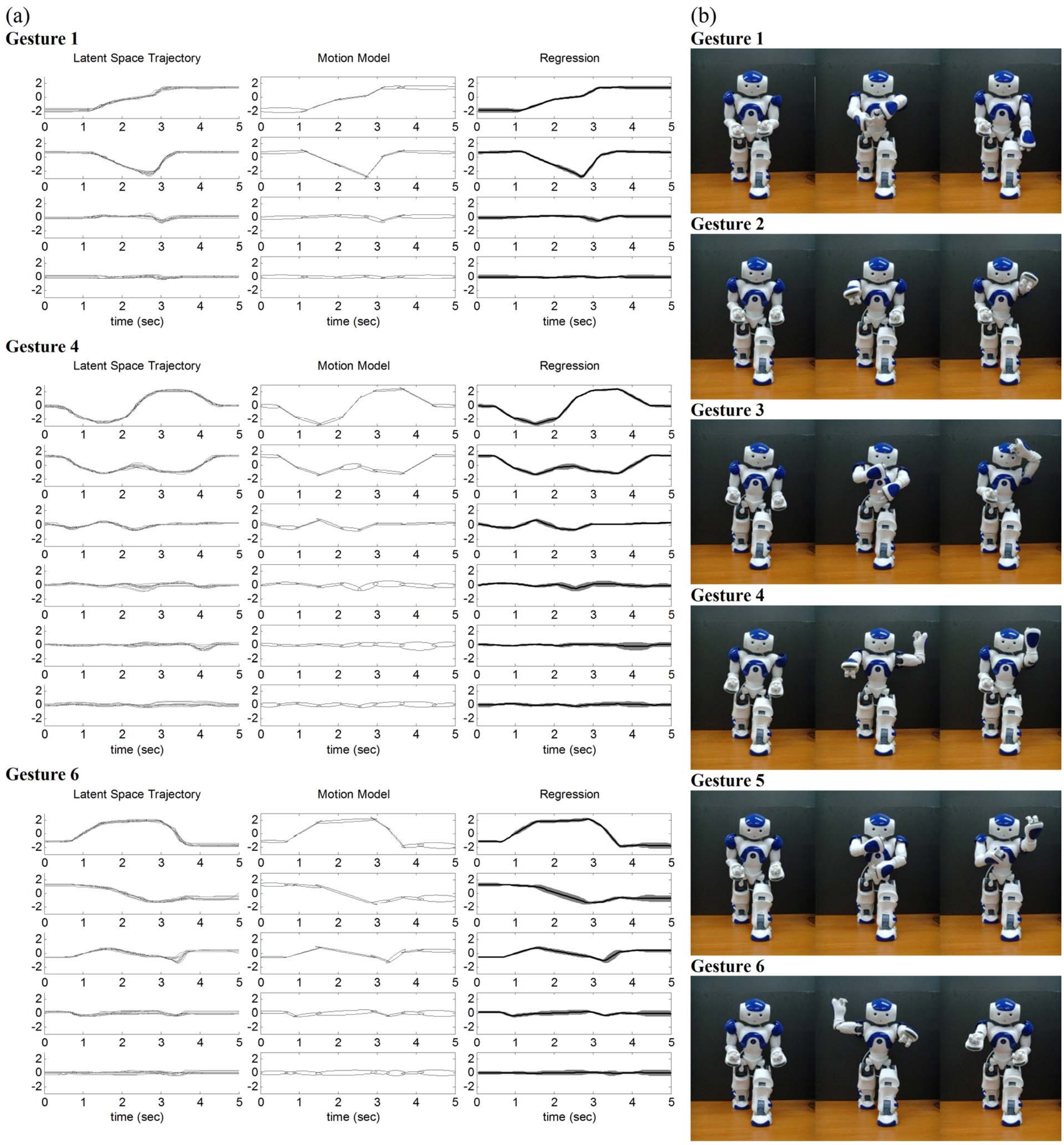

Fig. 3 (a) Illustration of learning of selected gestures: the data point trajectories for selected gestures (left column), learned model component alignment (middle column), and reproduced trajectories (right column). In each column, the y axis values are along the principal component directions of the latent space from the first principal direction down to the q-th principal direction respectively. Ellipses in the middle column represent covariance matrices. (b) Snapshots of the reproduced gestures by a robot.

[8] C. G. Atkeson, and S. Schaal, "Robot learning from demonstration", In *Proc. 14th Int. Conf. (ICML'97),* Nashville, TN, 8–12 July 1997 (ed. D. H. Fisher Jr),pp. 12–20. San Mateo, CA: Morgan Kaufmann.F. Zhou,

[9] C. Wan, L. Liu, "Research and improvement on embedded system application of DTW-based speech recognition", *in IEEE Int. Conf. ASID*, 2008.

[10] E. J. Keogh, and M. J. Pazzani, "Derivative Dynamic Time Warping", in *First SIAM Int. Conf. Data Mining*, Chicago, IL, 2001.

[11] H. Zhao, P.C. Yuen, and J.T. Kwok, "A novel incremental principal component analysis and its application for face recognition", *IEEE Trans Syst Man Cybern B Cybern*, vol 36, no. 4, pp. 873-86, 2006.

[12] M. Artac, M. Jogan, and A. Leonardis "Incremental PCA or On-Line Visual Learning and Recognition" in *Proc. Int. Conf. Pattern Recognition*, Quebec City, QC, Canada, 2002, vol. 3, pp. 781–784.

[13] M. Kurtz, *Handbook of Applied Mathematics for Engineers and Scientists*, 1st ed., McGraw-Hill Professional, 1991.