INT J COMPUT COMMUN, ISSN 1841-9836
8(6):869-877, December, 2013.

Dimensionality Reduction and Generation of Human Motion

S. Qu, L.D. Wu, Y.M. Wei, R.H. Yu

Shi Qu*
Air Force Early Warning Academy
No.288, Huangpu Road, Wuhan, 430019, China
*Corresponding author: qushi@nudt.edu.cn

Ling-da Wu, Rong-huan Yu
School of Equipment, Beijing, 101400, China

Ying-mei Wei
National University of Defense Technology
Changsha, 410073, China

Abstract: To reuse existing motion data and generate new motion, a method of human motion nonlinear dimensionality reduction and generation, based on fast adaptive scaled Gaussian process latent variable models, is proposed. Through statistical learning on motion data, the motion data are mapped from high-dimensional observation space to low-dimensional latent space to implement nonlinear dimensionality reduction, and probability distributing of posture space which measures the nature of posture is obtained. The posture which meets constraints and has maximal probability can be computed as the solution of inverse kinematics. This method can avoid cockamamie computation and posture distortion existing in traditional inverse kinematics. The experiments show that our method has higher convergence velocity and precision and extends editing range of motion by adapting motion editing direction.
Keywords: dimensionality reduction, Gaussian process, kernel function, motion generation.

1 Introduction

As the development of motion capture technology, it became reality to generate virtual character motion with motion capture data. This method, easy to implement and generating vivid motion, has been applied to movie, advertisement, game, military affairs and so on. As complexity and multiformity of motion, existing motion capture data are not enough to satisfy application in practice, it is need to reuse existing motion capture data to generate new motion.

Motion capture data, often high dimensional, require to be reduced dimension before analysis and research on them. According to mapping relation between high dimensional data and low dimensional data, dimensionality reduction techniques can be classified to linear or nonlinear. Principal component analysis (PCA) [1] is a linear dimensionality reduction technique in common use, which projects the high dimensional data along principal axis chosen by computing variance. Through improving PCA, many correlative techniques are proposed, which have better capability than PCA in specific application field. The probabilistic principal component analysis (PPCA) [2], introducing probability model into PCA, can complete dimensionality reduction and density estimate by modeling likelihood. In PPCA, which is linear too, high dimensional data and low dimensional data obey probabilistic distribution with noise. Combining PCA and kernel method [3], the kernel principal component analysis (KPCA) [4] is proposed. The basic idea of KPCA is that map the high dimensional data to a feature space by nonlinear method, and do principal component analysis in it. KPCA, as a nonlinear method, has the excellence of kernel method that establishes connection between high dimensional data and low dimensional data with kernel function not requires determining the specific relation between them. Gaussian process latent variable models (GPLVM) [5] [6] is another effective nonlinear dimensionality reduction technique, which completes dimensionality reduction and density estimate using Gaussian process regress. The GPLVM, different from PPCA, does not estimate the parameters of mapping but margin them. From the viewpoint of Bayesian, GPLVM maximizes the posterior probability to reduce dimensionality given prior probability.

The literature [7] implemented dimensionality reduction and generation of human motion based on scaled Gaussian process latent variable models (SGPLVM), in which there are two shortages, the one is slow convergence, and the other is small editing range of motion. In this paper, we improve on the work to propose fast adaptive scaled Gaussian process latent variable models (FASGPLVM), and implement dimensionality reduction and generation of human motion based on it. Our method has higher convergence velocity and precision and extends editing range of motion by adapting motion editing direction.

2 Overview

2.1 Human Skeleton and Motion Depiction

Human body consists of skeleton, muscle, skin and so on, and the skeleton determines human posture from the angle of motion. In this paper, we construct a simplified human skeleton as shown in Figure 1, which consists of 22 bones. The number in parentheses denotes the degree of freedom of corresponding bone, amounting to 50.

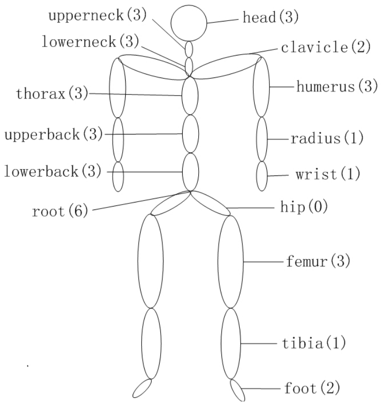


Figure 1: Simplified human skeleton

The degree of freedom denotes translation or rotation direction of bone. In the simplified human skeleton, the root has 3 translation directions and 3 rotation directions, the rest bones have 0 to 3 rotation directions separately. In the process of motion, human posture and location are determined by these degrees of freedom, concretely, the translation of root determines human location and the rotations of all bones determine human posture. So, a feature vector y can be constructed to represent human motion posture, which consists of all rotation degrees of freedom. In this paper, y is a column vector and $y \in R^{47}$. A motion, containing N frames, can be represented as a matrix $Y = [y_1, \cdots, y_t, \cdots, y_N]^T$, where $t(t = 1, 2, \cdots, N)$ is the time index and y_t is the feature vector correspond to the posture at time t.

2.2 Dimensionality Reduction and Generation of Human Motion

The framework of dimensionality reduction and generation of human motion based on FASGPLVM, shown as Figure 2, contains two stages, which are training model stage and generating posture stage. In training model stage, train FASGPLVM using sample motion data to obtain optimized latent trajectory and model parameters. At the same time, the probability distribution of sample motion is also obtained. In generating posture stage, synthesize the posture which has the maximal probability and satisfies constraints on the end effectors given by user, and update the active set which is a subset of sample motion data. The active set, instead of all sample motion data, are used to train model and synthesize new posture in order to improve computing efficiency.

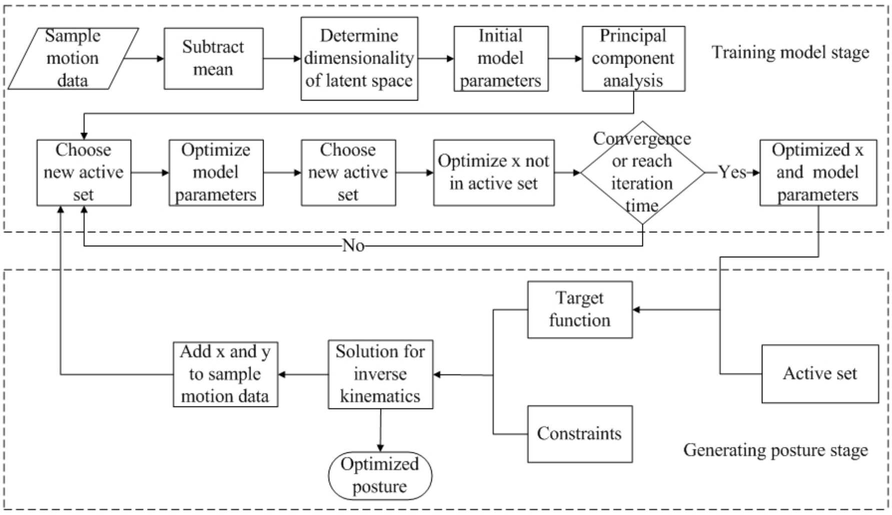

Sample motion data
Subtract mean
Determine dimensionality of latent space
Initial model parameters
Principal component analysis
Training model stage
Choose new active set
Optimize model parameters
Choose new active set
Optimize x not in active set
Convergence or reach iteration time
Yes
Optimized x and model parameters
No
Target function
Add x and y to sample motion data
Solution for inverse kinematics
Active set
Constraints
Generating posture stage
Optimized posture

Figure 2: Framework of dimensionality reduction and generation of human motion

3 Fast Adaptive Scaled Gaussian Process Latent Variable Models

Our work is based on the literature [7], in which SGPLVM is proposed by introducing scaled genes to GPLVM. In this paper, FASGPLVM is proposed, based on SGPLVM, to implement dimensionality reduction and generation of human motion. Similar to SGPLVM, scaled genes ω and active set are also included in FASGPLVM. The scaled genes $\omega = diag(\omega_1, \omega_2, \cdots, \omega_D)$ denote relatively important degree of each entry of feature vector y, ωy is used to train model instead of y. The active set is a subset of sample motion data, which is used to improve computing efficiency. Kernel function is the core of model, which has a great influence on training and application of model. With improvement on kernel function of SGPLVM, the summation of radius base function (RBF) and vectorial angle cosine is used as kernel function in FASGPLVM.

$$k(x_i, x_j) = \alpha exp(-\frac{\gamma}{2}(x_i - x_j)^T(x_i - x_j)) + \eta exp(\frac{x_i \cdot x_j}{|x_i||x_j|}) + \delta_{i,j}\beta^{-1} \tag{1}$$

where x_i is the latent variable correspond to y_i, the parameters α and η mean how correlated pairs of points are in general, γ means the spread of kernel function, β^{-1} is variance of noise and $\delta_{i,j}$ is Kronecker delta function which is 1 when i is equal to j, and 0 otherwise. Kernel function measures how similar two points in latent space are, concretely, a larger $k(x_i, x_j)$ denotes that x_i and x_j are more similar. In kernel function of literature [7], similarity between two points is measured by Euclid distance, and the larger Euclid distance is, the smaller similarity is. In FASGPLVM, vectorial angle cosine is introduced to associate similarity with position of point, and the similarity degrees of two pairs of points who have the same Euclid distance are not always equal. As shown in Figure 3,x_1, x_2, x_3, x_4, x_5 and x_6 are points in latent space which satisfy $(x_1 - x_2)^T(x_1 - x_2) = (x_3 - x_4)^T(x_3 - x_4) = (x_5 - x_6)^T(x_5 - x_6)$, we have the result $k(x_1, x_2) = k(x_3, x_4) = k(x_5, x_6)$ according to kernel function of literature [7]. According to kernel function of this paper, we have the result $k(x_1, x_2) < k(x_3, x_4)$,$k(x_1, x_2) < k(x_5, x_6)$ because of $\theta > \lambda, \theta > \varphi$, which is more reasonable. x_3x_4 results from rotating x_1x_2 which is perpendicular to center axis (the line connecting origin and center of x_1x_2), and x_1, x_2 have the smallest similarity. When $\lambda = 0$, x_3x_4 and center axis coincide. In this case, $x_3 = cx_4$(c is a constant) and the similarity between x_3 and x_4 reaches maximum. x_5x_6 results from translating x_1x_2, the difference between x_5 and x_6 is reduced because of $||x_5|| > ||x_1||$ and $||x_6|| > ||x_2||$. When x_5 and x_6 apart infinitely from origin, the similarity between x_5 and x_6 reaches maximum since they can be treated as one point.

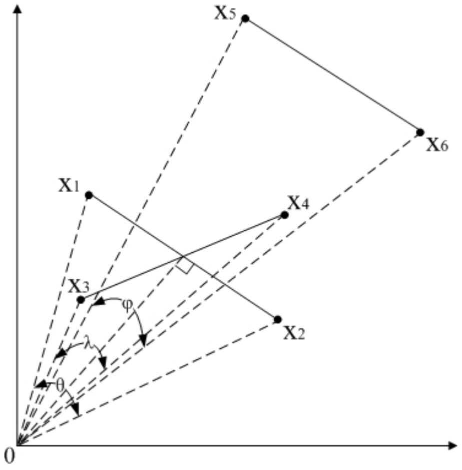


Figure 3: Influence of vectorial angle cosine on similarity measure

Another important improvement on SGPLVM is to introduce dynamic active set. In the motion editing process, the active set of FASGPLVM change, the model parameters $\alpha, \beta, \gamma, \eta, \omega_k$ and K covariance matrix change accordingly, which makes FASGPLVM adaptive to motion editing. In SGPLVM, the new posture is restricted to be similar to sample motion at full steam in order to ensure it vivid. However, the extent of motion editing can not be very large, otherwise, the new posture will be anamorphic. In FASGPLVM, it can enlarge the extent of motion editing and ensure new posture vivid by changing active set. The third improvement on SGPLVM is that convergence judgment, instead of fixed iteration time in SGPLVM, is introduced to training algorithm.

The algorithm of learning sample motion data based on FASGPLVM can be described as follows:

Step 1. Subtract the mean of motion data. The mean vector $\bar{y}$ is computed by

$$\bar{y} = \frac{1}{N} \sum_{1}^{N} y_i \tag{2}$$

Then, subtract the mean from each line of motion data matrix Y.

$$\bar{Y} = [\bar{y_1}, \bar{y_2}, \cdots, \bar{y_N}]^T = [y_1 - \bar{y}, y_2 - \bar{y}, \cdots, y_N - \bar{y}]^T \tag{3}$$

Step 2. Determine the dimensionality of latent space. FASGPLVM maps the motion data from high-dimensional observation space y^D to low-dimensional latent space x^d. The dimensionality of latent space is 2 or 3, and is determined by

$$d = \begin{cases} 2 & D < 60 \\ 3 & D \geq 60 \end{cases} \tag{4}$$

In this paper, $D = 47$, so $d = 2$.

Step 3. Initialize model parameters. In our experiments, $\alpha = 1, \beta = 1, \gamma = 1, \eta = 1, \omega_k = 1, M = 200, T = 100, C = 0.01$, where M is size of active set, T is allowable maximum iteration time, and C is convergence threshold.

Step 4. Initialize the latent variable with PCA. Compute d principal components of motion data and project motion data y_i to these principal components to obtain latent variable x_i. The initialization results are saved as a matrix $X = [x_1, x_2, \cdots, x_N]^T$.

Step 5. Update the active set with informative vector machine (IVM) [8].

Step 6. Estimate parameters $\alpha, \beta, \gamma, \eta, \omega_k$ by minimizing (5) with scaled conjugate gradients (SCG) [9].

$$L_{GP} = \frac{D}{2} ln|K| + \frac{1}{2}\sum_k \omega_k^2 \hat{\hat{Y}}_k^T K^{-1} \hat{\hat{Y}}_k + \frac{1}{2}\sum_l x_l^T x_l + ln(\alpha\beta\gamma\eta / \prod_k \omega_k^N) \tag{5}$$

where parameters D is the dimensionality of feature vector, $\hat{X}$,$\hat{\hat{Y}}$ are active set, $\hat{\hat{Y}}_k$ is the k-th column of $\hat{\hat{Y}}$, and K is covariance matrix whose entries are computed by (1), $K(i, j) = k(x_i, x_j)$.

Step 7. Update the active set with IVM according to step 5.

Step 8. Update the latent variable not in active set by minimizing (6) with SCG.

$$L_{IK(x,y)} = \frac{||\omega y - g(x)||^2}{2\sigma^2(x)} + \frac{D}{2} ln\sigma^2(x) + \frac{1}{2}||x||^2 \tag{6}$$

Here

$$g(x) = \mu + \hat{\hat{Y}}^T K^{-1} k(x) \tag{7}$$

$$\sigma^2(x) = k(x, x) - k(x)^T K^{-1} k(x) \tag{8}$$

Where parameters μ is the mean of motion data, $k(x)$ is a column vector in which the i-th entry contains $k(x_i, x)$, and x is latent variable.

Step 9. Stop iteration if $max\{||\Delta\alpha||, ||\Delta\beta||, ||\Delta\gamma||, ||\Delta\eta||, ||\Delta\omega_k||\} < C$, otherwise go to step 10.

Step 10. Stop iteration if iteration time reaches T, otherwise go to step 5.

After new posture was generated, corresponding (x, y) is added to sample set. The active set is updated using IVM, parameters $\alpha, \beta, \gamma, \eta, \omega_k$ and covariance matrix K is updated according to step 6. These are used in motion editing of next time in order to ensure FASGPLVM adaptive to motion editing and enlarge motion editing extent.

4 Posture and Motion Generation

Posture generation is a problem of inverse kinematics, which generates vivid posture satisfying constraints on the end effectors given by user. Constraints can be satisfied using geometric method easily; the difficulty in inverse kinematics is how to make new posture vivid because it is difficult to model human motion rules. In FASGPLVM, the probability space of motion posture is modeled, in which the posture near sample postures has bigger probability. The sample postures are motion capture data which has the best fidelity, so the probability of posture can be used as a measure of fidelity and the posture with bigger probability has better fidelity. In inverse kinematics, the posture which satisfies constraints and has the biggest probability is the solution of inverse kinematics. The problem of generating new posture given constraints on end effectors can be described as a nonlinear optimization problem.

$$\begin{aligned} Min(x, y) &= L_{IK}(x, y) \\ s.t. f(y) &= c \end{aligned} \tag{9}$$

Where $f(y) = c$ are constraints on end effectors, denoting the destination position for some bones of human skeleton. Generally speaking, not only one but a set of postures, called feasible posture set, can satisfy the constraints on end effectors, but some of them do not satisfy the motion rule and are not vivid. The posture, vivid and satisfying motion rule, is selected from feasible posture set by minimizing the target function $Min(x, y) = L_{IK}(x, y)$.

In our method, it is not needed to give constraints for all bones of human skeleton. In $f(y) = c$, just a few constraints interested for user are contained, which are called sparse constraints. In the process of optimization, weight gene is introduced to transform this problem to unconstrained nonlinear optimization.

$$Min(x, y) = (1 - \lambda)L_{IK}(x, y) + \lambda(f(y) - c) \tag{10}$$

where λ is the weight gene $(0 < \lambda < 1)$, it denotes intensity of constraints. In this paper, $\lambda = 0.999$.

Motion generation is based on posture generation, which is implemented by editing motion trajectory. Given a motion, we edit the motion trajectory of one joint and optimize (9) constrained by new motion trajectory to generate new motion.

5 Results and Analysis

In this section, we design experiments of dimensionality reduction and generation of human motion to prove the algorithms in this paper correct and effective. In our experiments, the human skeleton consists of 22 bones with 50 degrees of freedom. The observation space is 47 dimensional and the latent space is 2 dimensional. The sample motion data come from human motion capture database of Carnegie Mellon University[1].

5.1 Posture Generation

Jump and kick motion data are used to train FASGPLVM, which consist of 180 frames shown as figure 4. To represent motion process clearly, the left arm and left leg of human skeleton are rendered with gray color.

After trained with jump and kick motion data, the parameters of FASGPLVM are estimated as $\alpha = 456.912, \beta = 0.116, \gamma = 4.745, \eta = 326.582$. Then, we can generate new posture satisfying constraints given by the user. To compare with SGPLVM, train SGPLVM with the same motion data and generate new posture satisfying the same constraints. The experiment results are shown as figure 5, in which (a) is the origin posture, coming from sample motion data; (b), (c) and (d) are new postures generated by FASGPLVM; (e), (f) and (g) are new postures generated by SGPLVM. The dot on the right foot denotes the position of the right foot constrained by the user, (b) and (e) have the same constraint, and so do (c) and (f), (d) and (g). The process of editing posture (a) is to raise the right foot to generate new postures (b), (c), (d), (e), (f)

[1] http://mocap.cs.cmu.edu/

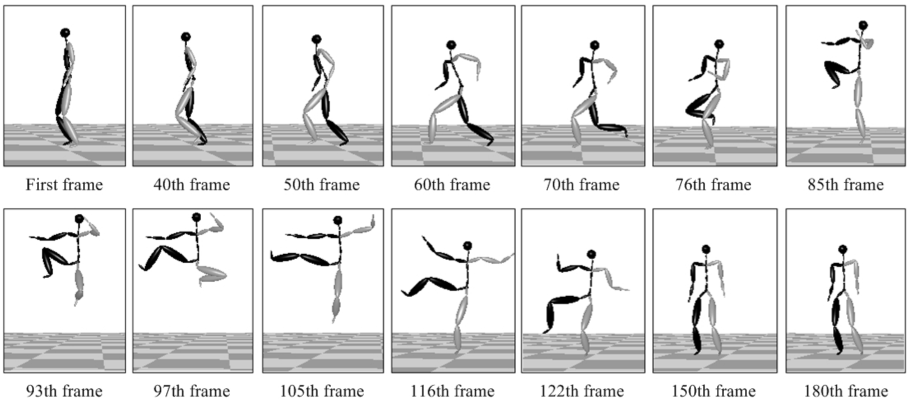


Figure 4: Motion capture data(jump and kick)

and (g). It can be seen from figure 5 that (d) is a vivid posture but the posture (g) distorts. This is because that the height of the right foot in the posture (g) oversteps the limitation of editing range of SGPLVM; but the posture (d) is generated based on FASGPLVM which has larger editing range.

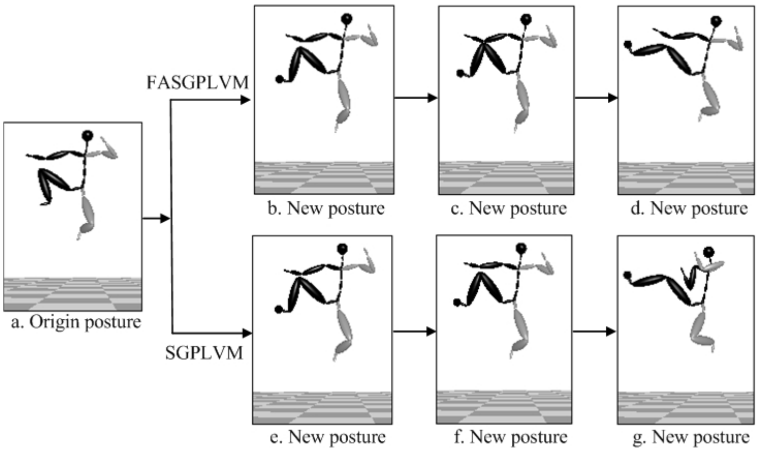
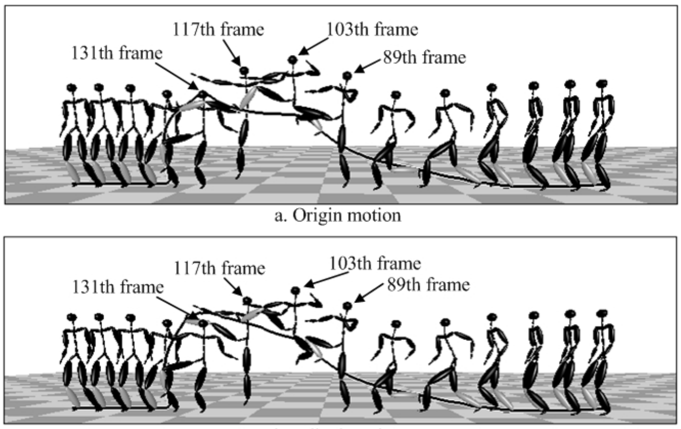


Figure 5: Posture generation (jump and kick)

5.2 Motion Generation

In this section, we generate new motion based on FAGPLVM. Editing the jump and kick motion shown in figure 4 to generate new motion. The 103th frame and the 117th frame are edited, raising the right foot. To generate new motion, the right foot positions of the 89th frame, the 103th frame, the 117th frame and the 131th frame are used as controlling points for Hermite interpolation to generate new motion trajectory. The new motion trajectory constrains the positions of the right foot in new motion, optimizing posture of every frame satisfying this constraint to generate new motion. As shown in figure 6, the black curve denotes the motion trajectory of the right foot.



a. Origin motion

b. Edited motion

Figure 6: Motion generation (jump and kick)

5.3 Comparison and Analysis

Table 1 is the convergence precision comparison of FASGPLVM and SGPLVM after the same iteration time. The convergence precision is defined as the change of model parameters. So, the convergence precision of FASGPLVM is $max\{||\Delta\alpha||, ||\Delta\beta||, ||\Delta\gamma||, ||\Delta\eta||, ||\Delta\omega_k||\}$, and the convergence precision of SGPLVM is $max\{||\Delta\alpha||, ||\Delta\beta||, ||\Delta\gamma||, ||\Delta\omega_k||\}$. It can be seen from table 1 that the convergence precision of FASGPLVM is higher than that of SGPLVM, this is because of improving on the kernel function in FASGPLVM.

Tab.1 The convergence precision comparison of FASGPLVM and SGPLVM

Sample motion	Frame amount	Iteration time	convergence precision	convergence precision
			(FASGPLVM)	(SGPLVM)
Kick	341	52	0.004	0.021
Run	264	46	0.002	0.014
Run and jump	536	71	0.005	0.017
Box	452	63	0.008	0.023
Long jump	367	58	0.006	0.026

Table 2 is the editing range comparison of FASGPLVM and SGPLVM retaining the new posture vivid. It can be seen from table 2 that the editing range of FASGPLVM is larger than that of SGPLVM, this is because of introducing dynamic active set in FASGPLVM.

Tab.2 The editing range comparison of FASGPLVM and SGPLVM

Sample motion	Edited frame number	Editing way	Editing range/cm	Editing range/cm
			(FASGPLVM)	(SGPLVM)
Kick	206	Raise fight foot	43	21
Run	124	Raise left hand	26	16
Run and jump	385	Raise fight foot	37	14
Box	162	Lower right hand	28	17
Long jump	256	Lower head	22	12

6 Conclusions

By improving on SGPLVM, FASGPLVM is proposed. A method of dimensionality reduction and generation of human motion based on FASGPLVM is also proposed. Compared with SGPLVM, FASGPLVM has some excellences, such as higher convergence precision and larger editing range.

Acknowledgments

Thanks to the anonymous reviewers for their helpful comments, Neil Lawrence for his on-line source code, and Carnegie Mellon University for human motion capture data.

Bibliography

[1] Sun Hongwei, Gu Ming and Sun Jiaguang,A coding algorithm using PCA-based correlation vector quantization, *Journal of Ccomputer-Aided Design & Computer Graphics*,17(8):1662-1666,2005.

[2] Gift N, Lorraine B and Isobel C G, Probabilistic principal component analysis for metabolomic data, *BMC Bioinformatics*,11(1):571-582,2010.

[3] Nisbet R, Elder J and Miner G, *Statistical analysis and data mining*, New York:Academic Press,2009.

[4] Roman Rosipal et al, Kernel PCA for feature extraction and de-noising in non-linear regression, *Neural Computing & Applications*,10(3):231- 243,2001.

[5] Carl H, Philip H S and Neil D L, Gaussian process latent variable models for human pose estimation, *Proc. of Machine Learning for Multimodal Interaction.* Brno: Springer-Verlag Press:132-143,2007.

[6] QU Shi et al, Pose Synthesis of Virtual Character Based on Statistical Learning, *The International Symposium on Computer Network and Multimedia*, Wuhan, China:36-39,2009.

[7] Keith Grochow et al, Style-based inverse kinematics, *ACM Transactions on Graphics*,23(3): 522-531,2004.

[8] Neil D. Lawrence, Matthias Seeger and Ralf Herbrich, Fast sparse Gaussian process methods: the informative vector machine,*Proceedings of Neural Information Processing Systems 15.* MIT Press:609-616,2003.

[9] Martin F. Muller, A scaled conjugate gradient algorithm for fast supervised learning, *Neural Networks*,6(4):525-533,1993.