

Article

# Whole-Body Joint Angle Estimation for Real-Time Humanoid Robot Imitation Based on Gaussian Process Dynamical Model and Particle Filter

Jian Mi \* and Yasutake Takahashi

Department of Human and Artificial Intelligent Systems, Graduate School of Engineering, University of Fukui, 3-9-1, Bunkyo, Fukui 910-8507, Japan; yasutake@ir.his.u-fukui.ac.jp

\* Correspondence: jmi@ir.his.u-fukui.ac.jp; Tel.: +81-776-27-8540 (ext. 4820)

Received: 1 October 2019; Accepted: 13 December 2019; Published: 18 December 2019



**Abstract:** Real-time imitation enables a humanoid robot to mirror the behavior of humans, being important for applications of human–robot interaction. For imitation, the corresponding joint angles of the humanoid robot should be estimated. Generally, a humanoid robot comprises dozens of joints that construct a high-dimensional exploration space for estimating the joint angles. Although a particle filter can estimate the robot state and provides a solution for estimating joint angles, the computational cost becomes prohibitive given the high dimension of the exploration space. Furthermore, a particle filter can only estimate the joint angles accurately using a motion model. To realize accurate joint angle estimation at low computational cost, Gaussian process dynamical models (GPDMs) can be adopted. Specifically, a compact state space can be constructed through the GPDM learning of high-dimensional time-series motion data to obtain a suitable motion model. We propose a GPDM-based particle filter using a compact state space from the learned motion models to realize efficient estimation of joint angles for robot imitation. Simulations and real experiments demonstrate that the proposed method efficiently estimates humanoid robot joint angles at low computational cost, enabling real-time imitation.

**Keywords:** particle filter; state estimation; 3D motion model; Gaussian process dynamical model (GPDM); prediction; humanoid robot

## 1. Introduction

The particle filter [1] is an important probabilistic method for state tracking and estimation [2,3], being widely utilized in robotics. For instance, particle filters are widely applied in mobile robots' self-localization [4,5] and visual tracking [6,7]. Three major factors greatly affect the performance of particle filters; namely, measurement model, state space, and motion model. The likelihood function in a measurement model is used for calculating the weights of particles and contributes to accurate state estimation. The particle filter distributes particles in the state space and performs estimation in the state space. The state space affects the number of particles that determines the computational cost of estimation. In addition, the motion model is applied in the state space for state tracking and enables the particle state update to the next state.

Figure 1 shows a real-time human motion imitation system using a particle filter with data acquired from a Microsoft Kinect sensor. The joint angles of the humanoid robot are estimated in real time to mirror the human motions. Note that a humanoid robot usually comprises dozens of joints, such as the NAO humanoid robot shown in Figure 2, which has 25 actuated joints (degrees of freedom—DOFs). Conventionally, a particle filter would be applied to the 25-dimensional (25D) state space, including all the joints, to estimate the corresponding angles in the humanoid robot for

imitation. This high-dimensional state space demands increasing the number of particles to ensure adequate accuracy. Consequently, the computational cost becomes relatively high, as state estimation with a particle filter is very complex for high-dimensional state spaces.

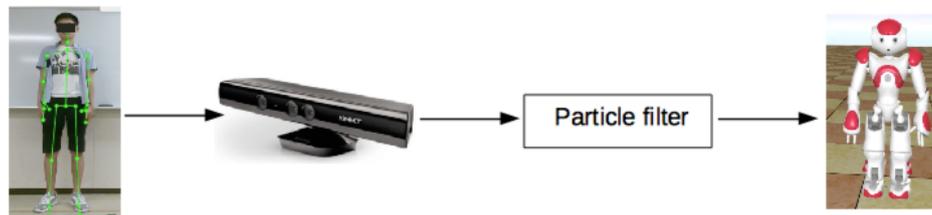


Figure 1. Estimation of joint angles for robot imitation using a particle filter.

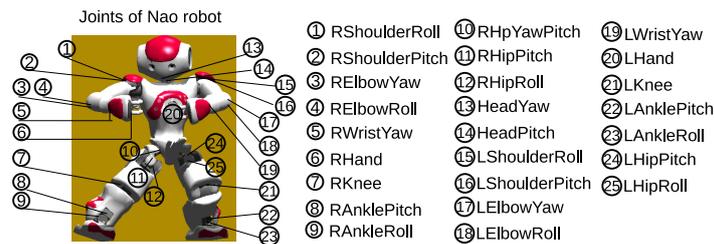


Figure 2. NAO humanoid robot comprising 25 joint angles.

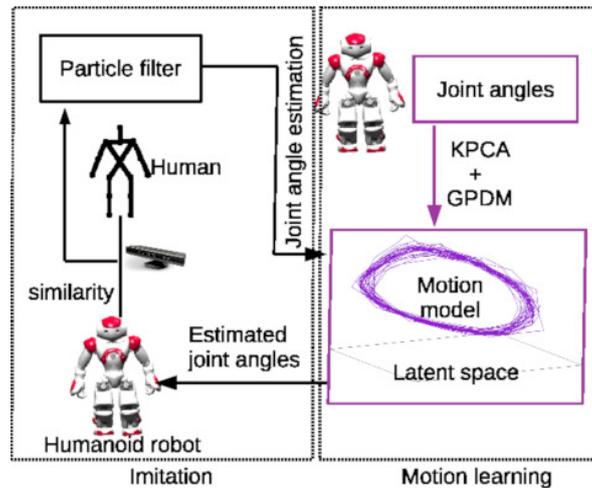
Usually, a particle filter applied to a high-dimensional state space requires many particles to perform accurate estimation, and the computational cost increases exponentially. The computational cost becomes a serious problem for humanoid robots given their usual structure with many DOFs. Hence, reducing the state space efficiently helps to constrain the computational cost of the particle filter. Such reduction can be achieved by replacing the high-dimensional state space with a low-dimensional and more efficient state space via various methods [8].

The goal of our work is to construct a real-time humanoid robot imitation system with low computational cost using a particle filter and a Gaussian process dynamical model (GPDM). One of the contributions of the work lies in combining a particle filter and GPDM. The high-dimensional state space is replaced by a GPDM learned, low-dimensional, latent state space to reduce the computational cost of state estimation using a particle filter. A GPDM learned model defines the motion model of the particle filter.

The architecture of the proposed system is illustrated in Figure 3. Motion data of a humanoid robot are collected as sequences of joint angles. A low-dimensional latent space and a motion model are constructed based on the motion data. Then, a human demonstrator performs a motion in front of a motion capture system. Our system explores an appropriate set of joint angles for the humanoid robot to imitate the demonstrated motion in real time.

Principal component analysis (PCA) [9] is an efficient and simple method for dimensionality reduction of data. Although it enables learning in low-dimensional latent space, it cannot provide a motion model for particle state tracking and may not be suitable for the nonlinear behavior of humanoid robot joints. Alternatively, kernel PCA [10] allows efficiently learning nonlinear data. Still, it also lacks a motion model for state tracking. Hence, a dynamical motion model should be constructed to realize particle state tracking in the latent state space.

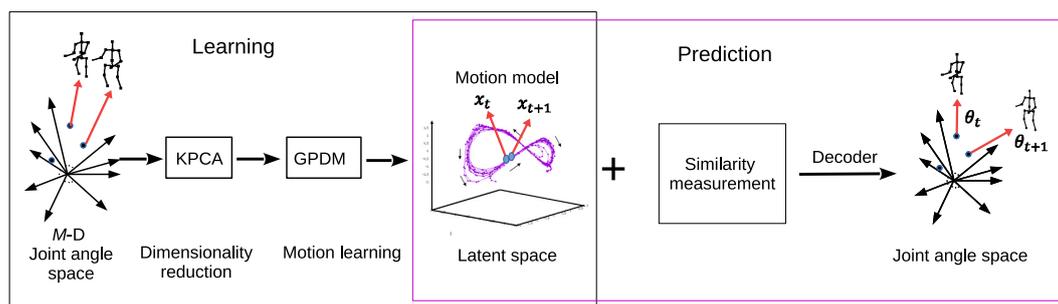
Gaussian process dynamical models (GPDMS) [11] are efficient for motion learning and suitable for dynamical motion modeling in the latent state space. Urtasun et al. [12] and Kim et al. [13] proved that human dynamical motion models can be constructed in a low-dimensional latent space. Humanoid robot dynamical motion models were learned in the research by Mi and Takahashi [14] by using kernel PCA-GPDM. Compared with conventional neural network methods, GPDM has advantages over learning motion models with modest amounts of training data. It is efficient in motion learning and the time cost is much less.



**Figure 3.** Proposed joint angle estimation of humanoid robot using particle filter in 3D space with learned motion models for imitating human motion (kernel principal component analysis (KPCA)).

In this study, the main function of GPDM is to construct a compact state space and build motion models. Then, we replace the state space of particle filter with kernel PCA-GPDM learned compact latent state space, and efficient humanoid robot motion models based on GPDMs are applied to state tracking in the learned latent state space based on a particle filter.

Following the architecture shown in Figure 3, we use a particle filter and GPDM learning to realize high-dimensional humanoid robot state estimation in a reduced 3D latent space at low computational cost. Figure 4 details the humanoid robot motion learning and prediction system. The high-dimensional space of humanoid robot joint angles is first reduced to a 3D state space using kernel PCA. Then, a 3D dynamical motion model is learned by GPDM. A decoder is used to reconstruct the humanoid robot joint angles to ensure that the learned 3D latent variables can be mapped onto the joint angle space, such that the learned motion models can be applied to particle-filter state estimation. A 3D latent state space is obtained for reducing the computational cost of robot state estimation. The learned 3D dynamical motion models are applied to the particle filter for particle state tracking. Specifically, a prediction system with similarity measurements based on the dynamical motion models is proposed for humanoid robot state estimation.



**Figure 4.** Motion learning and prediction using a Gaussian process dynamical model (GPDM).  $M$  is the dimension of the joint angle space,  $x_t$  is the state of a particle at time  $t$ , and  $\theta_{t+1}$  represents the joint angles of the humanoid robot, which are reconstructed from state  $x_{t+1}$  using a GPDM decoder.

The main contributions of this paper can be summarized as follows.

- We construct a real-time humanoid robot imitation system with low computational cost based on particle filter and GPDM. The high-dimensional state space is replaced by the GPDM learned compact state space and the GPDM model defines the motion model of the particle filter.

- A prediction system with similarity measurement for state tracking of the particle filter is proposed to improve state estimation accuracy.

Experiments demonstrate that the proposed particle filter can estimate the joint angles of a humanoid robot efficiently using the GPDM in the 3D state space for real-time imitation of human motion.

## 2. Related Works

Mi and Takahashi [15] performed robot state estimation with particle filter in a 3D state space. However, the conventional particle filter suffers the problem of a high computational cost when a particle filter is applied in high-dimensional state space. Su et al. [6] combined a particle filter and a visual significance model to solve the sudden movement tracking problem. By embedding a visual significance model into particle filter, their method was efficient in handling abrupt motion tracking problems. However, their particle filter method was also applied in a 3D state space so that it suffers the same problem as the conventional particle filter. In addition, their method is not suitable for estimating high-dimensional humanoid robot joint angles. Zhang [7] proposed a novel correction particle filter (CPF) for robust visual tracking. Similarly to the method in [6], the CPF method was efficient at handling partial occlusion and scale variation for object tracking; however, it is not suitable for constructing a real-time humanoid robot imitation system.

Cao et al. [16] proposed an approach to detect 2D pose of people in an image. Mehta et al. [17] presented a real-time method to capture the full global 3D skeletal pose of a human using a single RGB camera. Those studies used a convolutional neural network (CNN) method to estimate human pose. Unlike those studies, we aimed to estimate a humanoid robot's pose instead of a human's pose. There is no necessity to build complex neural networks to capture human poses for the purpose of estimating humanoid robot joint angles. A RGB-D camera, such as Kinect, is more convenient for us to capture human poses.

Capecchi et al. [18] used a RGB-D camera to extract human motion features and developed a hidden semi-Markov model (HSMM) based approach to evaluate the performance of people affected by motor disabilities. They proposed a HSMM based method to model human motions for the purpose of evaluating the correctness of rehabilitation exercises. Unlike their research, for the purpose of constructing a real-time imitation system with low computational cost, we combine GPDM and particle filter to estimate high-dimensional humanoid robot joint angles. We first model humanoid robot motions in a compact state space using GPDM. Then, the estimation is performed in a GPDM learned compact state space, and finally, we restore the joint angles from the compact state space to the high-dimensional humanoid robot joint angle space.

Kondo and Takahashi [8] proved that a high-dimensional state space can be replaced by a low-dimensional latent space learned by dimension reduction. They utilized an autoencoder, a type of artificial neural network, to acquire a 3D state space from the original high-dimensional space of the humanoid robot joint angles. A particle filter was then applied to the resulting 3D state space, achieving state estimation of the humanoid robot with a relatively low computational cost. Unfortunately, the autoencoder constructed the 3D state space to replace the high-dimensional space without the required motion models for particle state tracking. Therefore, they assumed that motion in the 3D space was random according to a Gaussian distribution. An appropriate motion model realizes efficient state tracking from current state  $s_t$  to next state  $s_{t+1}$ .

Humanoid robot dynamical motion models were learned in the research by Mi and Takahashi [14] by using kernel PCA-GPDM. In their research, they only constructed 3D humanoid robot motion models, but the learned motion models were not verified. In this paper, we combine the particle filter and GPDM to realize real-time humanoid robot imitation with low computational cost.

### 3. Humanoid Robot State Estimation Based on a Particle Filter

In this section, we first introduce a humanoid robot imitation system based on particle filter with neither dimension reduction of the state space nor motion model construction. Then, we highlight the drawbacks of this system, motivating the use of GPDM.

We used the NAO humanoid robot manufactured by Softbank Robotics for the experiments of state estimation in the imitation system. A Microsoft Kinect V2 sensor was used for acquiring human motion data. The joint angles of the humanoid robot were estimated by a particle filter, as the robot imitated human motion. Our experimental procedure is close to that by Kondo et al. [8] and Takahashi and Sakakibara [19]. The humanoid robot imitates human motions using the particle filter system, as shown in Figure 1. Human skeleton joint data are obtained from the Kinect sensor, and the robot joint angles are estimated using the particle filter. The state space of the particle filter system has 25 dimensions, corresponding to the 25 actuated joints of the robot.

The human skeleton joints detected by the Kinect sensor are used for calculating the particle weights during the estimation of the humanoid robot joint angles. The joints of the human skeleton model are shown in Figure 5. The elements in vector  ${}^h\mathit{link} = [{}^h\mathit{link}_1, {}^h\mathit{link}_2, \dots, {}^h\mathit{link}_{11}]$  are shown as green lines and correspond to the model links, which were used to calculate the similarity between the human and robot poses. The similarity is used for calculating the particle weights. Link  ${}^h\mathit{link}_1$  joins HEAD and SPINE\_SHOULDER, and the other links, similarly join other skeleton joints to resemble the human body structure. The links of the humanoid robot are defined similarly, and the corresponding vector is denoted as  ${}^r\mathit{link}$ . Pose similarity  $\mathit{sim}$  between the human and robot is calculated by a cosine similarity measurement:

$$\mathit{sim} = \frac{1}{N_{\mathit{link}}} \sum_{i=1}^{N_{\mathit{link}}} \frac{{}^h\mathit{link}_i \cdot {}^r\mathit{link}_i}{\|{}^h\mathit{link}_i\| \|{}^r\mathit{link}_i\|}, \tag{1}$$

where  $N_{\mathit{link}}$  is the number of links. Then, weight  $w$  of a particle is calculated by the following likelihood function:

$$w = \exp\left(-\frac{1}{2\sigma^2}(1 - \mathit{sim})^2\right). \tag{2}$$

The particles are defined in a 25D joint angle space, and the particle set is denoted as  $P_t = [p_t^1, p_t^2, \dots, p_t^Q]$ , where  $Q$  is the number of particles. Each particle is given by  $p_t^q = [\theta_t^1, \theta_t^2, \dots, \theta_t^M]$ , where  $M$  is number of the humanoid robot joints. The estimation process using particle filter is detailed in Algorithm 1.

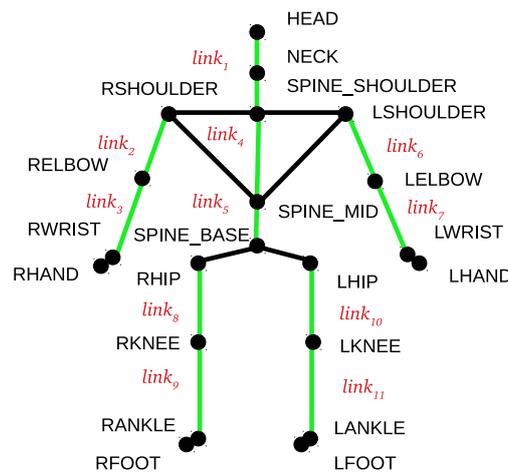


Figure 5. Skeleton joints and links of human body model.

A motion model is also important for particle state tracking in the particle filter. However, it is difficult to determine an analytical function for the 25D space of the robot joint angles. Instead, a simple motion model can be defined as

$$\mathbf{p}_t = \mathbf{p}_{t-1} + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2)\Delta t, \quad (3)$$

where  $\Delta t$  is the sampling time and  $\mathcal{N}(\mathbf{0}, \sigma^2)$  denotes Gaussian noise with zero mean and variance  $\sigma^2$ . Below, we replace the motion model with one based on GPDM.

The joint angles of the humanoid robot can be estimated as the weighted mean of the particles:

$$\mathbf{p}_t = \frac{\sum_{q=1}^Q \mathbf{p}_t^q w_t^q}{\sum_{q=1}^Q w_t^q}, \quad (4)$$

where  $w_t^q$  is the weight of particle  $q$  at time  $t$ .

---

**Algorithm 1:** Particle filter for humanoid robot joint estimation

---

- 1: Initialize particle sets  $\bar{\mathbf{P}}_t = \emptyset$  and  $\mathbf{P}_t = (\mathbf{p}_t^1, \mathbf{p}_t^2, \dots, \mathbf{p}_t^Q)$ ;
  - 2: **for**  $q = 1$  to  $Q$  **do**
  - 3:   Update particles based on motion model:  $\mathbf{p}_t^q \leftarrow \mathbf{p}_{t-1}^q$ ;
  - 4:   Calculate particle weights:  $w_t^q \propto sim$ ;
  - 5:    $\bar{\mathbf{P}}_t = \bar{\mathbf{P}}_t + \langle \mathbf{p}_t^q, w_t^q \rangle$ ;
  - 6: **end for**
  - 7: Clear set  $\mathbf{P}_t = \emptyset$ ;
  - 8: **for**  $q = 1$  to  $Q$  resample particles **do**
  - 9:   Draw  $q$  from  $\bar{\mathbf{P}}_t$  with probability  $\propto w_t^q$ ;
  - 10:   Add  $\mathbf{p}_t^q$  to  $\mathbf{P}_t$ ;
  - 11: **end for**
  - 12: Return  $\mathbf{P}_t$ ;
- 

The 25D state space leads to a very high computational cost of the particle filter. This hinders estimation of joint angles for humanoid robot imitation in real time. Figure 6 shows an example sequence of humanoid robot joint angle estimation in real time using this implementation of the particle filter. As it is difficult to estimate 25 joint angles in real time directly, we estimated the joint angles of only the left and right arms, reducing the state space to 12 dimensions. In this case, the number of particles  $Q = 400$ . Even when only estimating 12 joint angles, a delay exists, as seen in Figure 6. From Figure 6c–e, the human motion takes about 2 s, whereas the humanoid robot needs approximately 4 s to follow the movement, with a substantial lag of approximately 2 s. Note that the computational cost would be even higher when estimating the 25 joint angles.

To reduce the computation time, we use GPDM learning to construct a reduced 3D state space with a humanoid robot dynamical motion model. Then, the humanoid robot joint angles are estimated in the learned 3D state space at low computational cost.

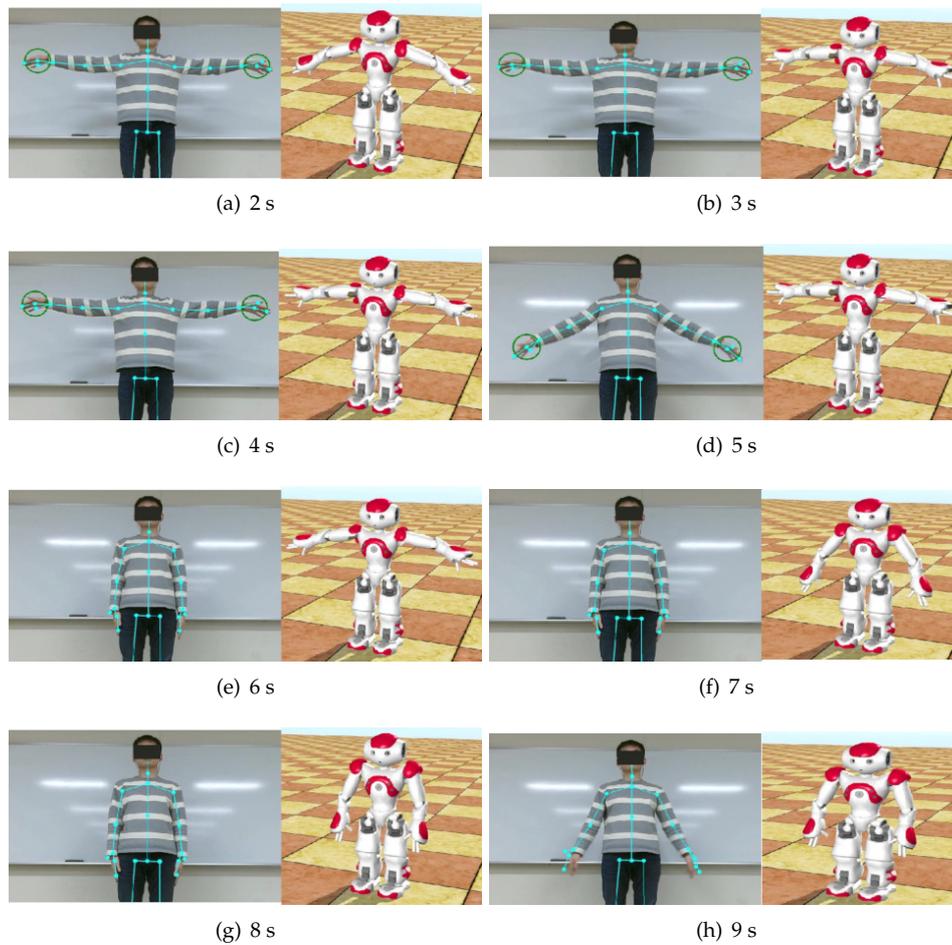


Figure 6. Joint angle estimation using direct implementation of particle filter.

#### 4. Humanoid Robot Motion Learning

The GPDM is applied to construct a compact state space and learn efficient humanoid robot motion models. The proposed humanoid robot motion learning is illustrated in Figure 4. Observed humanoid robot joint angles  $\theta$  are reduced to  $x$  in a 3D state space using kernel PCA [14] as shown in Figure 7a. Then, 3D latent variables  $x$  are relearned using GPDM [20]. A dynamical mapping  $f_x : x_t \rightarrow x_{t+1}$  is constructed as shown in Figure 7b. Moreover, a decoder mapping  $f_\theta : x_t \rightarrow \theta_t$  (Figure 7c) is learned to reconstruct high-dimensional joint angles (for details of GPDM learning, see [14]).

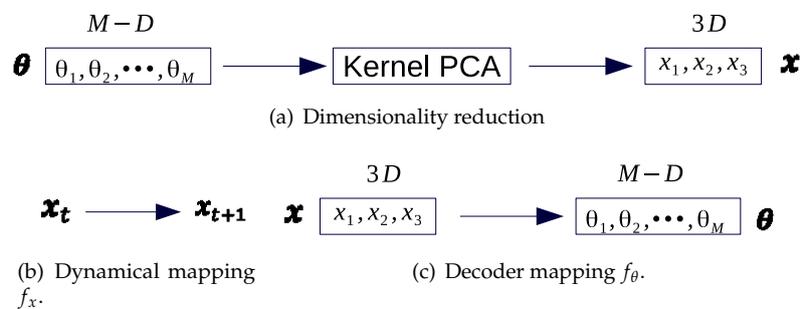


Figure 7. Mappings for humanoid robot motion learning.

### 5. Motion Learned Using Gaussian Process Dynamical Model

We obtained the humanoid robot joint angles using the particle filter estimation detailed in Section 3. The humanoid robot was supposed to wave its hands, as shown in Figure 6. The learning data set  $X$  comprised 500 frames of the humanoid robot joint angles selected from 1000 frames of continuous motion data.

The learning hyperparameters were initially set to  $\alpha = [1.0, 1.0, 1.0, \exp(1.0)]$ ,  $\beta = [1.0, 1.0, \exp(1.0)]$ , and  $\omega_j = 1.0$  (for more details, please see [14]). The GPDM learned 3D motion models are illustrated in Figure 8a. According to [11], if learned trajectories are smooth, prediction based on the learned motion models tends to be accurate. We used the GPDM [21] to make predictions, as Gaussian process prediction [11] retrieves a dynamical mapping for state transition and GPDM decoder mapping to reconstruct the humanoid robot joint angles.

For dynamical mapping  $f_x$  in the 3D state space, state  $x_{t+1}$  is predicted from  $x_t$  as follows:

$$x_{t+1} \sim \mathcal{N}(f_x(x_t); \sigma_x^2(x_t)\mathbf{I}), \tag{5}$$

$$x_{t+1} = f_x(x_t) = X_{2:T}^T K_X^{-1} k_X(x_t, X). \tag{6}$$

Similarly, for GPDM decoder mapping  $f_\theta$ , given latent state  $x_t$ , humanoid robot joint angle  $\theta_t$  can be reconstructed as

$$\theta_t \sim \mathcal{N}(f_\theta(x_t); \sigma_\theta^2(x_t)\mathbf{I}), \tag{7}$$

$$\theta_t = f_\theta(x_t) = \Theta^T K_\Theta^{-1} k_\Theta(x_t). \tag{8}$$

In dynamical prediction mapping  $f_x$  and decoder mapping  $f_\theta$ ,  $\sigma_x^2(x_t)$  and  $\sigma_\theta^2(x_t)$  represent the corresponding variances.  $k_X(\cdot)$  and  $k_\Theta(\cdot)$  are kernel functions.  $K_X$  and  $K_\Theta$  are kernel matrix.

Figure 8b depicts the modeled and predicted 3D latent variables. The predicted latent variables are consistent and mostly overlap with the learned GPDM for motion. Hence, dynamical mapping  $f_x$  and the learned GPDM motion model are accurate. Figure 9 shows the angles from two joints of the humanoid robot reconstructed using GPDM decoder  $f_\theta$ , demonstrating the accuracy of decoder  $f_\theta$ .

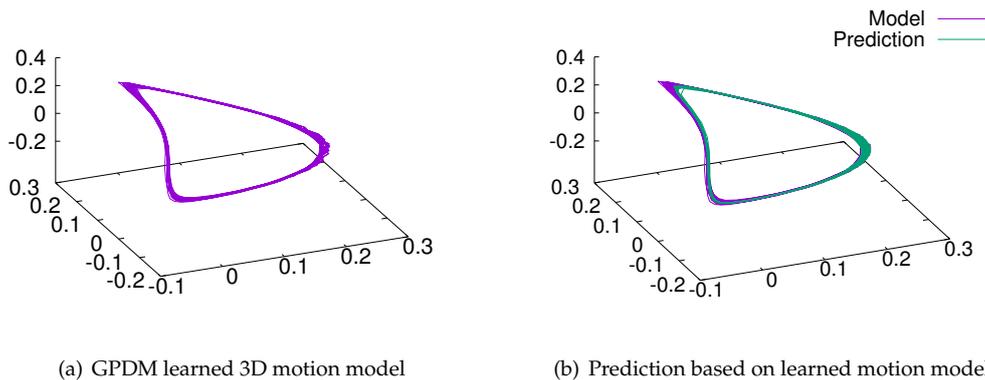


Figure 8. GPDM learned 3D dynamical motion model and model-based prediction.

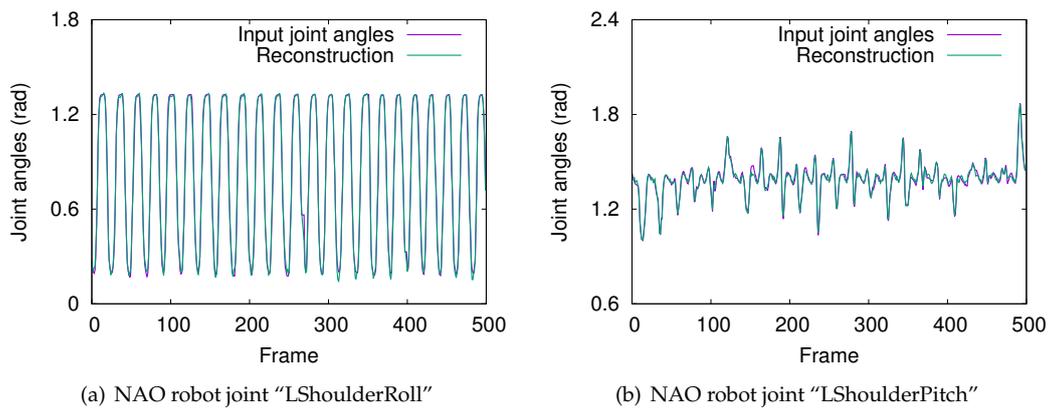


Figure 9. Joint angle reconstruction using decoder  $f_{\theta}$ .

### 6. Joint Angle Prediction Using Gaussian Process Dynamical Motion Models

State  $x_t$  is used to predict  $x_{t+1}$  through the learned dynamical mapping,  $f_x$ . Figure 10 shows the proposed state transition system for real-time robot imitation. The dynamical mapping enables state transition, and the GPDM decoder reconstructs the humanoid robot joint angles. Hence, we can predict the humanoid robot joint angles in real time by map  $x_t \rightarrow \theta_{t+1}$ .

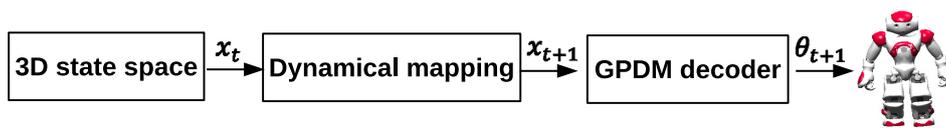


Figure 10. Real-time joint angle prediction using GPDM motion models.

In addition, we devised a similarity measurement module to improve the prediction accuracy of the humanoid robot joint angles, as the accuracy from discrete state  $x_t$  is limited. The similarity measurement determines the position of  $x_t$  in the learned motion model. A sequence of states  $X^* = [x_{t-k-1}, \dots, x_{t-1}, x_t]$  is used instead of discrete state  $x_t$ , with  $k$  being the size of  $X^*$ . The learned motion model is divided to form database  $X_u = [X_1, X_2, X_3, \dots]$  (as shown in Figure 11), where the size of each  $X_i$  ( $X_i \subset X_u$ ) is also  $k$ . A cosine distance function is adopted to calculate the similarity between  $X^*$  and  $X_i$ :

$$dis = \frac{X^* \cdot X_i}{\|X^*\| \|X_i\|}. \tag{9}$$

Then, input  $X^*$  is updated with the  $X_i$  retrieving the highest similarity. Thus, the joint angle is predicted with an updated  $X^*$ , reducing the uncertainty of discrete state  $x_t$ , and thus, improving the prediction accuracy.

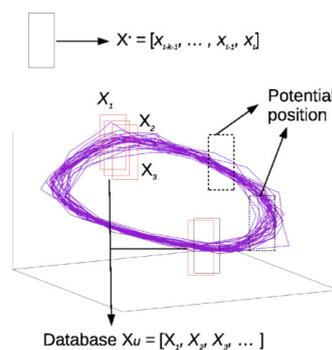


Figure 11. Similarity measurement for state  $x_t$ .

To verify the prediction system, we collected the humanoid robot joint angles from the same motion as learning data to form a test dataset. The test dataset contained 500 frames of humanoid robot joint angles. Figure 12 illustrates the encoding of humanoid robot joint angle  $\theta_t$  into  $x_t$ . Based on the GPDM learned motion models, we made predictions using the dynamical mapping to obtain  $x_{t+1}$ . Finally, the humanoid robot joint angles in state  $\theta_{t+1}$  were reconstructed. Note that the GPDM does not define an encoder mapping, which is obtained by a feed-forward neural network, where the learning dataset of humanoid robot joint angles,  $\Theta$ , is the input, and the GPDM learned motion model,  $X$ , is the output. We perform relearning for map  $\Theta \rightarrow X$  as the encoder. Note that we divide the learning dataset in to training set (70%), validation set (15%), and testing set (15%) in learning encoder mapping.

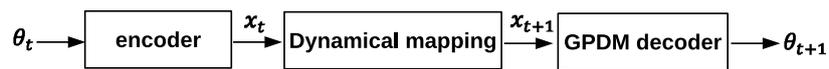


Figure 12. Verification of joint angle prediction.

We set  $k = 5$  and performed predictions of the humanoid robot joint angles in real time to verify the prediction system. We selected two important humanoid robot joint angles (LShoulderRoll and LElbowRoll) and obtained the accurate predictions shown in Figure 13. For further verification, we applied the predicted joint angles to a simulated NAO humanoid robot. We found that the predicted joint angles without using the proposed similarity measurement elicit unnatural motions of the humanoid robot. Errors occur by the predicted joint angles being beyond the range of humanoid robot motion. Figure 13a,b shows the predicted joint angles using the proposed similarity measurement. The predicted joint angles with the similarity measurement module appear to be smooth and are consistent with the real joint angles. The predicted joint angles with the proposed similarity measurement make the robot wave its hands naturally, as shown in Figure 14, verifying the effectiveness of using this measurement.

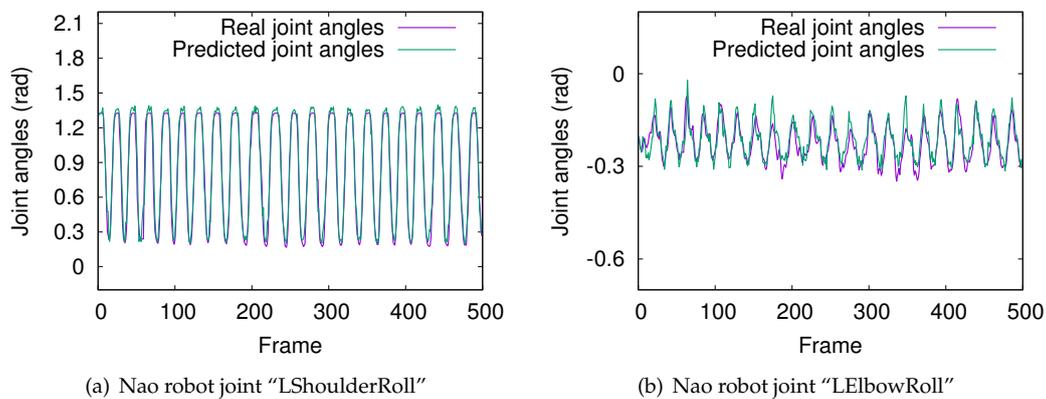
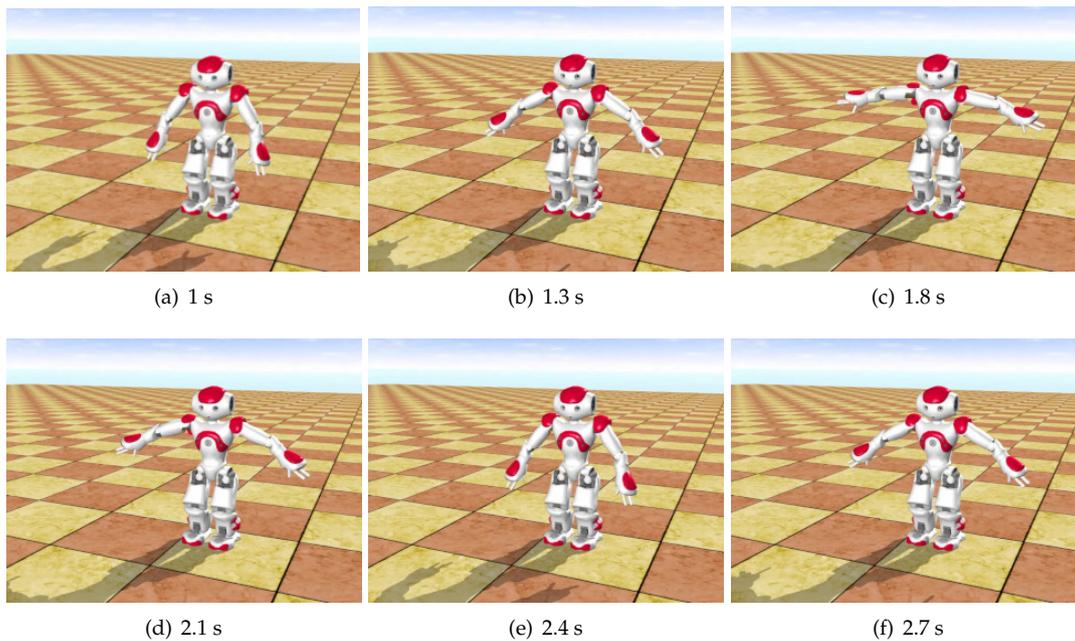


Figure 13. Prediction of joint angles for a humanoid robot using similarity measurement.



**Figure 14.** Motion simulation for predicted joint angles using similarity measurement.

## 7. Humanoid Robot Joint Angle Estimation Using a Gaussian Process Dynamical Motion Model and Particle Filter

The proposed particle filter was then applied to the learned 3D state space shown in Figure 8a. In the conventional particle filter detailed in Section 3, each particle is defined as  $\mathbf{p}_t^q = [\theta_t^1, \theta_t^2, \dots, \theta_t^M]$ , with  $M$  being number of the humanoid robot joint angles. Using the GPDM, the particles are distributed over the 3D state space, and each particle is given by  $\mathbf{p}_t^q = [x_t^1, x_t^2, x_t^3]$ .

Table 1 shows the differences between conventional particle filter method and our proposed system. The high-dimensional state space is replaced by a 3D state space. The computational cost is greatly reduced when replacing the high-dimensional state space by the 3D GPDM learned state space.

**Table 1.** Comparison of conventional particle filter (PF) and our GPDM based particle filter.

Method	State Space	Number of Particles	Motion Model	Measurement Model
PF	$M$	$Q$	Gaussian	Cosine similarity
PF + GPDM	3	$Q$	GPDM	Cosine similarity

Figure 15 shows the particle filter estimation system using the learned GPDM, where particles are distributed in the 3D state space. The current state,  $\mathbf{p}_t$ , can be obtained from the previous state,  $\mathbf{p}_{t-1}$ , using learned GPDM dynamical mapping  $f_x$  in the 3D state space. Once the humanoid robot states are estimated in the 3D state space, the joint angles are reconstructed using GPDM decoder  $f_\theta$ .

Figure 16 shows a sequence of real-time joint angle estimation using the GPDM and particle filter. This real-time estimation outperforms that using only the particle filter, illustrated in Figure 6. The experiment results of real-time joint angle estimation using only particle filter in Figure 6 shows that it delayed about 2 s when the Nao robot imitated human motions. This is because the computational cost is relatively high when a particle filter performs estimation in high-dimensional state space. It cannot be avoided that the system takes much time to estimate humanoid robot joint angles in high-dimensional state space and delay exists. By combining a particle filter and a GPDM, we reduced the high-dimensional joint angle space into the GPDM learned 3D state space, and the computational cost was greatly reduced. The GPDM motion model was also adopted to make accurate the estimation

of humanoid robot joint angles. The imitation experiments show that the humanoid robot moves almost simultaneously with the human, and the delay is reduced greatly by applying the estimation in the 3D state space. These results verify that the learned GPDM enable efficient humanoid robot state estimation in the 3D state space with low computational cost for real-time imitation of human motion.

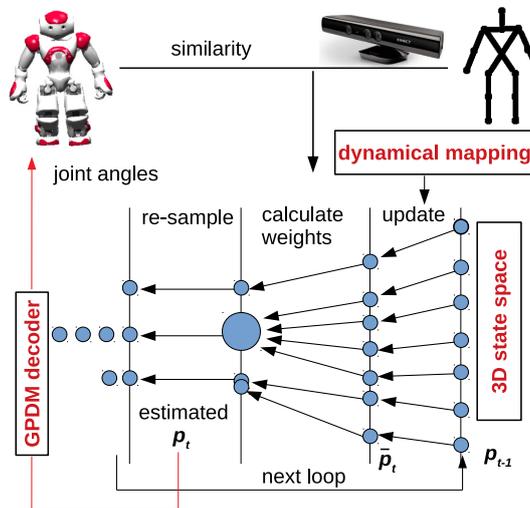


Figure 15. Particle filter estimation with GPDM.

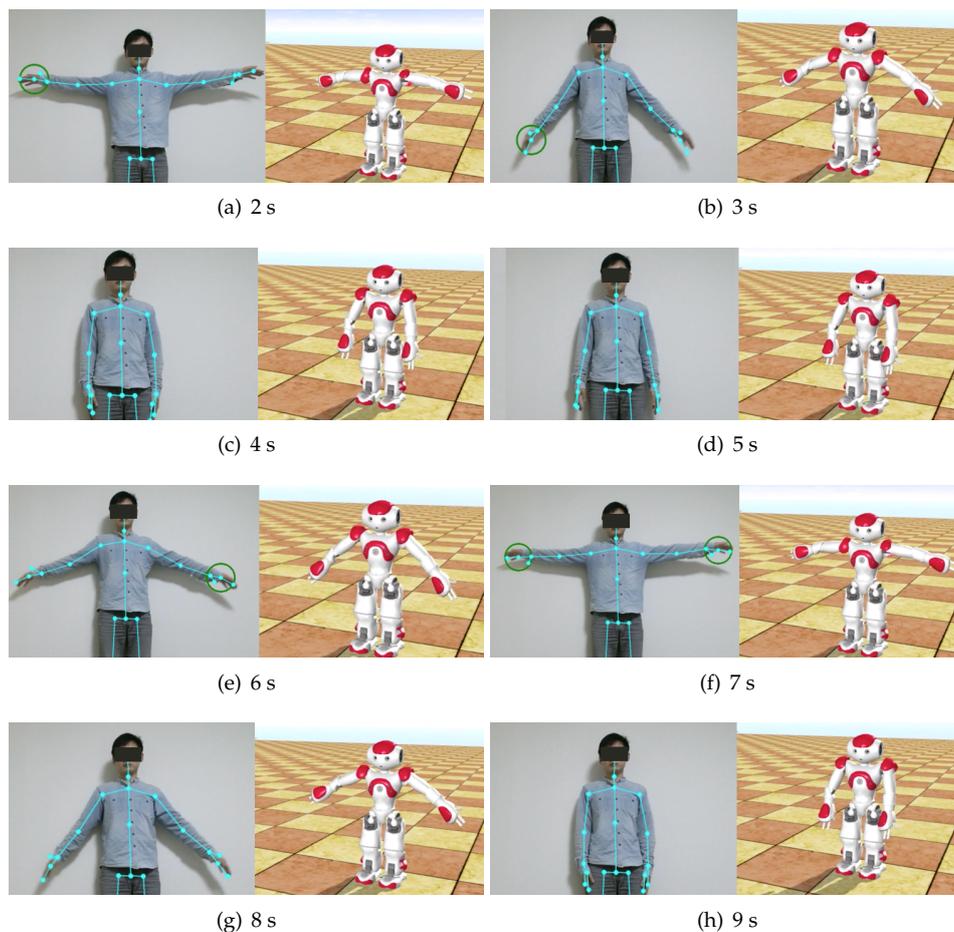


Figure 16. Joint angle estimation for robot imitation using GPDMs and particle filter.

## 8. Conclusions

We propose a system using GPDM motion models and a particle filter for high-dimensional humanoid robot state estimation. The GPDM is applied to the particle filter for estimating the high-dimensional space of robot joint angles at a low computational cost in a 3D state space, which replaces the original high-dimensional robot state space. The GPDM is used to learn 3D humanoid robot motion models, which are adopted for particle state tracking during estimation. Moreover, prediction is implemented based on the learned GPDM. We applied the learned GPDM motion models for particle filter state estimation. Compared with estimation of humanoid robot joint angles only using the particle filter, the GPDM along with the particle filter improves efficiency, and the estimation of joint angles provides better robot imitation. Both simulations and real experiments demonstrate that the GPDM learned humanoid robot motion models are efficient, and, along with the particle filter, can realize high-dimensional humanoid robot state estimation in a 3D state space at low computational cost for real-time imitation. The simple motion model learning in this study will be extended to construct complex motion models in future work. The particle filter and GPDM systems were evaluated by the delays between robot motion and human motion in the imitation experiments rather than computational time. In the future we will evaluate the systems by computational time and consider evaluating the proposed method from view of the imitation accuracy.

**Author Contributions:** Y.T. conceived the idea and prepared the experiment equipment. J.M. designed the experiments and wrote the paper. This work was supervised by Y.T. All authors have read and agreed to the published version of the manuscript.

**Acknowledgments:** The authors would like to thank those who provided assistance to our work.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

GPDM	Gaussian process dynamical model
PCA	principal component analysis
DOF	degree of freedom

## References

1. Thrun, S.; Burgard, W.; Fox, D. *Probabilistic Robotics*; The MIT Press: Cambridge, MA, USA, 2006.
2. Dong, H.; Cao, B.; Yang, Y. Application of Particle Filter for Target Tracking in Wireless Sensor Networks. In Proceedings of the 2010 International Conference on Communications and Mobile Computing, Shenzhen, China, 12–14 April 2010; Volume 3, pp. 504–508. doi:10.1109/CMC.2010.228. [[CrossRef](#)]
3. Xu, Y.; Xu, K.; Wan, J.; Xiong, Z.; Li, Y. Research on Particle Filter Tracking Method Based on Kalman Filter. In Proceedings of the 2018 2nd IEEE Advanced Information Management, Communication, Electronic and Automation Control Conference (IMCEC), Xi'an, China, 25–27 May 2018; pp. 1564–1568. doi:10.1109/IMCEC.2018.8469578. [[CrossRef](#)]
4. Mi, J.; Takahashi, Y. Low cost design of HF-band RFID system for mobile robot self-localization based on multiple readers and tags. In Proceedings of the 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO), Zhuhai, China, 6–9 December 2015; pp. 194–199. doi:10.1109/ROBIO.2015.7418766. [[CrossRef](#)]
5. Mi, J.; Takahashi, Y. Design of an HF-Band RFID System with Multiple Readers and Passive Tags for Indoor Mobile Robot Self-Localization. *Sensors* **2016**, *16*, 1200. doi:10.3390/s16081200. [[CrossRef](#)] [[PubMed](#)]
6. Su, Y.; Zhao, Q.; Zhao, L.; Gu, D. Abrupt motion tracking using a visual saliency embedded particle filter. *Pattern Recognit.* **2014**, *47*, 1826–1834. doi:10.1016/j.patcog.2013.11.028. [[CrossRef](#)]
7. Zhang, T.; Liu, S.; Xu, C.; Liu, B.; Yang, M. Correlation Particle Filter for Visual Tracking. *IEEE Trans. Image Process.* **2018**, *27*, 2676–2687. doi:10.1109/TIP.2017.2781304. [[CrossRef](#)] [[PubMed](#)]

8. Kondo, Y.; Yamamoto, S.; Takahashi, Y. Real-Time Posture Imitation of Biped Humanoid Robot Based on Particle Filter with Simple Joint Control for Standing Stabilization. In Proceedings of the 2016 Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS), Sapporo, Japan, 25–28 August 2016; pp. 130–135. doi:10.1109/SCIS-ISIS.2016.0039. [[CrossRef](#)]
9. Jolliffe, I.T. *Principal Component Analysis*; 0172-7397; Springer: New York, NY, USA, 2002. doi:10.1007/b98835. [[CrossRef](#)]
10. Schölkopf, B.; Smola, A.J. *Learning with Kernels*; The MIT Press: Cambridge, MA, USA; London, UK, 2001.
11. Wang, J.M.; Fleet, D.J.; Hertzmann, A. Gaussian Process Dynamical Models for Human Motion. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 283–298. doi:10.1109/TPAMI.2007.1167. [[CrossRef](#)] [[PubMed](#)]
12. Urtasun, R.; Fleet, D.J.; Fua, P. 3D People Tracking with Gaussian Process Dynamical Models. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 17–22 June 2006; Volume 1, pp. 238–245. doi:10.1109/CVPR.2006.15. [[CrossRef](#)]
13. Kim, T.; Park, J.; Heo, S.; Sung, K.; Park, J. Characterizing Dynamic Walking Patterns and Detecting Falls with Wearable Sensors Using Gaussian Process Methods. *Sensors* **2017**, *17*, 1172. doi:10.3390/s17051172. [[CrossRef](#)] [[PubMed](#)]
14. Mi, J.; Takahashi, Y. Humanoid Robot Motion Modeling Based on Time-Series Data Using Kernel PCA and Gaussian Process Dynamical Models. *J. Adv. Comput. Intell. Intell. Inform.* **2018**, *22*, 965–977. doi:10.20965/jaciii.2018.p0965. [[CrossRef](#)]
15. Mi, J.; Takahashi, Y. Performance analysis of mobile robot self-localization based on different configurations of RFID system. In Proceedings of the 2015 IEEE International Conference on Advanced Intelligent Mechatronics (AIM), Busan, Korea, 7–11 July 2015; pp. 1591–1596. doi:10.1109/AIM.2015.7222770. [[CrossRef](#)]
16. Cao, Z.; Simon, T.; Wei, S.E.; Sheikh, Y. Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
17. Mehta, D.; Sridhar, S.; Sotnychenko, O.; Rhodin, H.; Shafiei, M.; Seidel, H.P.; Xu, W.; Casas, D.; Theobalt, C. VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera. *ACM Trans. Graph.* **2017**, *36*, 44:1–44:14. doi:10.1145/3072959.3073596. [[CrossRef](#)]
18. Capecci, M.; Ceravolo, M.G.; Ferracuti, F.; Iarlori, S.; Kyrki, V.; Monteriù, A.; Romeo, L.; Verdini, F. A Hidden Semi-Markov Model based approach for rehabilitation exercise assessment. *J. Biomed. Inform.* **2018**, *78*, 1–11. doi:10.1016/j.jbi.2017.12.012. [[CrossRef](#)] [[PubMed](#)]
19. Takahashi, Y.; Sakakibara, K. Real-time joint angle estimation for a humanoid robot to imitate human motion using particle filter. The Japanese Society for Artificial Intelligence (JSAI), SIG-Challenge, Proceedings of the 42nd Meeting of Special Interest Group on AI Challenges, Fukui, Japan, 3 May 2015; pp. 21–22.
20. Wang, J.; Hertzmann, A.; Fleet, D.J. Gaussian Process Dynamical Models. In *Advances in Neural Information Processing Systems 18*; The MIT Press: Cambridge, MA, USA, 2006; pp. 1441–1448.
21. Mackay, D.J. *Information Theory, Inference, and Learning Algorithms*; Cambridge University Press: Cambridge, UK, 2003. doi:10.1007/b98835. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).