Long time prediction of human lower limb movement based on IPSO-BPNN

Dongnan Jin a, Yali Liu *, Xunju Ma b, and Qiuzhi Song c

School of Mechatronical Engineering, Beijing Institute of Technology, Beijing 100081, China.

*3120180205@bit.edu.cn, b 3120195134@bit.edu.cn, c qzhson@bit.edu.cn

Abstract. System delay caused by mechanical transmission, control calculation and data communication are the main factor affecting the man-machine collaborative control of lower extremity exoskeleton. Improved Particle Swarm Optimization Algorithm (IPSO) was proposed to optimize BPNN (Back Propagation Neural Network) to predict the future joint angle of human lower limb. The 3d motion capture system was used to collect the Angle data of human lower limb joints, and time span was added to reconstruct the time series, which was taken as the input of the model. Compared to PSO (Particle Swarm Optimization), IPSO added a three-route competitive optimization trajectory, the training feedback of BPNN and mutation operation, which accelerated the convergence of the algorithm and avoided local optimization. Besides, we established a prediction evaluation criterion with prediction duration, iteration efficiency, Root Mean Square Error (RMSE) and Determination Coefficient (DC) as the core to analyze the prediction results of BPNN, PSO-BPNN (Support Back Propagation Neural Network by Particle Swarm Optimization) and IPSO-BPNN (Support Back Propagation Neural Network by Improved Particle Swarm Optimization). The results show that the average RMSE of IPSO-BPNN is less than 0.75 and DC is more than 98%. IPSO-BPNN can make more accurate prediction of human lower limb joint angle, which is beneficial to improve the man-machine coordination performance of exoskeleton.

Keywords: Joint angle prediction; PSO; BPNN; Evaluation.

1. Introduction

With the development of science and technology, exoskeleton robot has shown great application prospects in military, medical, fire protection, anti-terrorism and other fields, which has become a research hotspot in the field of robotics [1-5]. The special technology of the exoskeleton robot is that human in the control loop. Ideally, the exoskeleton robot movement should be in consistent with the human movement trajectory at every moment. However, the system delay, caused by mechanical transmission, control calculation and data communication [6], inevitably makes assistance of the robot lag behind the movement of the human body. By means of human-computer interaction, the real-time
motion of human can be detected and predicted so as to realize the coordination and unity of human-computer movement. Therefore, motion prediction becomes a key part of robot system.

For a long time, scholars have proposed many prediction algorithms for exoskeleton robot control system. Zhiyu Zhao [7,8] used firefly algorithm to optimize the prediction model based on Extreme Learning Machine (ELM), which was used to predict the foot pressure of walking, jogging, climbing stairs and descending stairs at different walking speeds. The average RMSE of his predicted results was about 0.68, with a determination coefficient of 96.47%. Tao Qin [9] used the phase space reconstruction theory of chaos model to reconstruct the time series, and then built the model with the improved Least Squares Support Vector Machine (LS-SVM) to predict gait. The best accuracy of his prediction results was 96.6%. Hailin He [10] combined Whale Optimization Algorithm (WOA) and Genetic Algorithm (GA) to optimize the model of Support Vector Machine (SVM), and the accuracy of gait detection could reach 98.8%. Shilei Li [11] used the improved Taknes algorithm to predict gait, with an accuracy rate of 95.29% and a Smoothness Factor (SF) of 0.0018. In addition, there are some research on prediction based on database template matching [12], machine learning [13], and online sparse Gaussian process [14].

There are also some problems in the above research, such as the delay of the system not covered by the prediction time and the imperfect evaluation system of the prediction effect. Therefore, we proposed a new human lower limb joint angle prediction algorithm. Firstly, we reconstructed the time series of lower limb joint angle collected by the 3d motion capture system, with a time span to control the prediction duration freely. Then, we improved the particle movement mode of PSO, which was changed into a three-route competitive Optimization trajectory added the training feedback of BPNN, to optimize the prediction model based on BPNN. In addition, mutation operation was added to avoid local optimization. Finally, an evaluation criterion with prediction duration, iteration efficiency, RMSE and DC as the core was established to conduct in-depth analysis of the prediction results of various prediction models.

The remainder of this article is organized as follows. Section 2 presents the data acquisition, data processing, complete prediction model and evaluation criteria of angle prediction effect. Section 3 introduces the experimental results. Section 4 discusses the results and methods. Finally, Section 5 concludes the paper.

2. Experimental and Method

2.1. Experiment

In this study, 10 healthy men (male, height: 177.30 ± 6.03, weight: 62.00 ± 4.65, foot length: 27.00 ± 0.55, BMI: 19.70 ± 0.47) participated in the experimental data collection process. The subjects’ postures (including knee, hip, ankle joint angle data) while walking on a Bertec treadmill (Bertec Corporation, Columbus, OH, USA) with an inclination angle of 0° at a speed of 4.5 km/h were captured by the 3d motion capture system (Motion Analysis, USA). The sampling frequency f was 120Hz. More than 100 continuous gait cycles were collected for each subject. All subjects were provided with an informed consent form for the experiment approved by the Ethics Committee at the Beijing Sport University (Approval code: 2019007H).
Figure 1. Original angle data of the hip, knee and ankle joint

Figure 1 shows part of the original angle of the hip, knee and ankle from one of the subjects. The red lines represent the data on the right; the black lines represent the data on the left. The data of the left and right joints cross each other and show some symmetry. On average, a full gait has about 120 sample points. All the data were repeated periodically in a gait cycle, which made it possible to predict the motion posture. Data of hip joint is similar to sinusoidal wave, and the fluctuation range is -25°~20°. The overall fluctuation range of knee data was 0°~70°, and there was a small crest with a peak of about 20° near the trough. There is also a small crest with a peak value of about 7° in the ankle joint data, and the overall fluctuation range is -20°~25°.

2.2. Data Preprocessing

Due to the problems of point loss and pulse interference in the original data, it is necessary to process the original time series by median average filtering method to protect the continuity and smoothness of the data. Secondly, since the angle fluctuation range of different joints was not consistent, to get the prediction model faster and better, the original data was normalized using min-max normalization [15]. Formula (1) was used to restrict the original data.

\[ y_i = y_{\min} + \left( y_{\max} - y_{\min} \right) \times \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \]  

where \( x_i \) represents the original data, \( x_{\max} \) and \( x_{\min} \) are the maximum and minimum values of \( x_i \), respectively. \( y_i \) is the normalized data, and \( y_{\max} \) and \( y_{\min} \) are the maximum and minimum values of \( y_i \), respectively. In this paper, we set \( y_{\max} = 1 \) and \( y_{\min} = -1 \).

Finally, time span T was added to reconstruct the original time series. As shown in Figure 2, the set of the discrete data \( \{t_1, t_2, \ldots, t_n\} \) after filtering and normalization constituted the original time series \( \{R_t\} \). The input data length of a single joint Angle of the BPNN model was defined as L, and the output data length was N. In this paper, we set L=6 and N=3. In other words, we used the 6 historical data to predict the data of 3 observation points in the future. During the reconstruction, the k-th sequence started with the data \( t_{1+(k-1)\cdot T} \). Then, we intercepted the data from \( t_{1+(k-1)\cdot T} \) to \( t_{n-(L+N-k+1)\cdot T+1} \), which constituted the time series \( \{S_k\} \), where k is a positive integer between [1, L+N]. So, we ended up with a total matrix \( \{S_1\}; \{S_2\}; \ldots; \{S_L\}; \ldots; \{S_{L+N}\} \), which represented the information of one joint. The first L rows of each column of the matrix would be part of the input inputs of the BPNN model, which constituted the input matrix \( \{In_L\} \). The last N rows of each column would be the output, which constituted the output matrix \( \{Out_N\} \). Apart from this, the first 70% of the matrix was used for training and the last 30% for testing.
Figure 2. The reconstruction of time series

The above process was repeated to reconstruct the time series of other joint angle data. In this paper, we built different models to predict different joint angles. Therefore, the input of the model was the vertical superposition of the input matrix of 6 joints, and the output matrix of only one joint was selected.

The relationship between the time span \( T \) and the prediction duration \( T_{\text{pre}-i} \) of the \( i \)-th observation point in the future was explained in Formula 2, where \( \tau \) was the system delay and \( f \) was the sampling frequency. The predicted duration must be larger than the system delay to have a positive effect on the collaborative control.

\[
T_{\text{pre}-i} = \frac{i \tau T}{f}
\]  

(2)

2.3. Back Propagation Neural Network
As a widely used classical neural network model, BPNN has the advantages of simple model calculation, strong nonlinear mapping ability, local fault tolerance, self-learning and self-adaptation.

Figure 3. Structure of the BPNN

In this paper, we designed a 3-layer BPNN model, as shown in Figure 3. The input of the model was the historical angle data of 6 joints, and the data length \( L \) was 6, so the number of nodes in the input layer was 36. A single model only predicted the Angle of a single joint, and the output data length \( N \) was 3, so the number of nodes in the output layer was 3. The tansig function and purelin function were used as the transfer function from the input layer to the hidden layer, and from the hidden layer to the output layer. The training function and threshold training function of BPNN were trainlm function and learndm function. In addition, the Mean Square Error (MSE, Formula 3) was applied as the error performance function of the model.

\[
MSE = \frac{1}{n} \sum_{k=1}^{n} |y_k - f(x_k)|
\]  

(3)
where \( n \) is the length of the data, \( y_k \) is the actual measured value, and \( f(x_k) \) is the predicted value.

The design of hidden layer referred to empirical Formula 4, and finally 12 nodes were set according to the prediction results.

\[
\sum_{i=0}^{n} C_{n+1}^i > k
\]  

(4)

where \( n \) is the input unit number, \( n_1 \) is the hidden unit number, and \( k \) is the sample number. If \( i > n_1 \), then \( C_{n+1}^i = 0 \). Besides, the learning goal, the learning rate, and the number of training epochs of BPNN were set to be 0.01, 0.1, and 50,000, respectively. After data testing and verification, the randomness of the initial value of BPNN caused the instability of the model prediction results, so we used the improved particle swarm optimization algorithm to optimize the BPNN model.

### 2.4. Classical PSO

PSO algorithm sets \( N \) particles to imitate birds to search for the optimal solution in multidimensional space. Each particle has the velocity property \( V_i \) and the position property \( P_i \), where \( i \) is the particle label. \( V_i \) represents the movement of the particle, as shown in Formula 5. \( P_i \) represents the result of the particle's movement, as shown in Formula 6, which is also a candidate solution to the corresponding optimization problem. The distance between the current candidate solution and the actual optimal solution is measured by the fitness value \( F_i \) of each particle, which is evaluated by MSE. The individual extremum \( G_i \) is the optimal solution for individual particle search compared with its' historical and current solutions. The population extremum \( Z \) is the optimal solution for the whole particle swarm compared with its' historical solution and all individual extremum of current generation. In the process of searching and iteration, particle population keeps updating the velocity and position of particles by tracking individual extremum and group extremum, so as to obtain the optimal solution satisfying the requirements. The final group extremum \( Z \) is the final optimization result.

\[
V_{i-new} = \omega V_i + c_1 r \text{and}(G_i - P_i) + c_2 r \text{and}(Z - P_i)
\]

(5)

\[
P_{i-new} = V_{i-new} + P_i
\]

(6)

where the left side of the equation is the new velocity and position of the current generation, and the right side is the property of the particle of the previous generation. \( \omega \) is the inertia factor, and the value is 0.8. \( c_1 \) and \( c_2 \) are learning factors, both of which are 1.5. \( r \text{and}() \) generates a random number between (0,1). Apart from this, we set the population size \( N \) and the maximum number of iteration \( M \) to be 50 and 300 respectively. To prevent excessive random searches, the speed and position threshold were set as [-1, 1] and [-5, 5], respectively.

### 2.5. IPSO

Since the classical PSO has some problems such as slow search speed and local optimization, we proposed an Improved Particle Swarm Optimization (IPSO) algorithm, which mainly optimized the movement process of individual particles. Figure 4 explained the particle movement process of PSO and IPSO, where the solid line was the track before optimization, and the dotted line was the track after optimization.
The calculation of PSO only reflected the results $\overline{SC}$, which ignored the role of point A and B. Based on that, we proposed a new particle update mode (Formula 7-10, Figure 5), which add two search directions passing through point A and B to expand the search space of the particle population. The new three search paths were $\overline{SADC}$, $\overline{SBDC}$ and $\overline{SCDC}$. In addition, since the fitness calculation after particle movement required the verification training of BPNN model which also brought the particle position update, we enhanced this feedback to improve the search efficiency of particles. The new position $P_{i-new}$ was the closest of the three search paths to the actual optimal solution whose fitness value was the minimum. Through the competition of three search paths and training feedback, we accelerated the search speed of individual particles, thus accelerating the convergence of the whole particle swarm.

$$V_{i-A} = \omega V_i$$
$$V_{i-B} = \omega V_i + c_1 r a n d () (G_i - P_i)$$
$$V_{i-C} = \omega V_i + c_1 r a n d () (G_i - P_i) + c_2 r a n d () (Z - P_i)$$
$$P_{i-new} = \min (F_i (A), F_i (B), F_i (C))$$

Figure 5. The flow chart of IPSO

In order to solve the local optimization problem, a mutation operation was applied to the individual extremum after the update, which meant the position attribute of the particle would be reinitialized with a certain probability $T_i$ (Formula 11). The updating of group extremum was consistent with the original algorithm.

$$T_i = \sqrt{r a n d () \frac{K}{M}}$$

where K is the current number of iterations, and M is the maximum number of iterations.

2.6 Complete Prediction Model

Figure 6 shows the complete prediction model, which takes the historical angle data of 6 joints of human lower limbs as input and the future data as output. We developed separate models to predict the angle of the hip, knee, and ankle, respectively.

The prediction model can be divided into the following three parts: initialization of the model, IPSO and test. The initialization consists of data preprocessing and BPNN model architecture. On the one hand, the original data needs to be filtered, normalized and reconstructed. On the other hand, the structure, initial value and other parameters of BPNN need to be set. Then, we used IPSO to optimize
BPNN, which would be trained with training dataset. Finally, the prediction was given on the test dataset, and the prediction evaluation system was established to analyze the results.

Figure 6. The prediction model

2.7. Evaluation Criteria of Prediction Effect
The BPNN model optimized by IPSO would be further trained. Then, the test data set would be used to test the prediction effect. We also established a prediction evaluation criterion with prediction duration, iteration efficiency, RMSE (Formula 12) and DC (Formula 13) as the core to analyze the prediction results.

\[
RMSE = \sqrt{\frac{1}{n} \sum_{k=1}^{n} (y_k - f(x_k))^2}
\]  
\[
DC = 1 - \frac{\sum_{k=1}^{n} (y_k - f(x_k))^2}{\sum_{k=1}^{n} (y_k - \bar{y})^2}
\]

where \( n \) is the length of the data, \( y_k \) and \( \bar{y}_k \) are the actual measured angle and its’ average value, and \( f(x_k) \) is the predicted value.

3. Results
The prediction model set three observation points for future Angle data, and the prediction duration of each observation point was shown in Table 1. The system delay of the exoskeleton robot prototype in this paper was about 50ms. The prediction duration of the first observation point was close to the system delay, but the prediction durations of the second and third observation points were longer than the system delay. The three prediction observation points complemented each other, which could improve the cooperative control performance of exoskeleton robot.

| \( T_{pre-i} \) (ms) | 1    | 2    | 3    |
|----------------------|------|------|------|
| \( i \)              | 41.67| 83.33| 125  |
Figure 7. Iteration efficiencies of PSO and IPSO

The algorithm implementation and result analysis were implemented in a custom software (MATLAB R2016a). Figure 7 showed the partial algorithm iteration process of the right three joints, where the solid and dotted lines respectively represented the optimization iteration process of PSO and IPSO, and the yellow, green and blue lines respectively represented the hip, knee and ankle joints. The results showed that the PSO iteration results converged to 0.03 on average. Compared with PSO, the average fitness value of IPSO optimization results was less than 0.01, which decreased by about 0.02. During the iterative process, PSO was extremely easy to fall into local optimization. On average, PSO approached its final convergence position at the 50th generation and truly converged at the 100th generation. Moreover, there was a certain probability of continuous optimization in the later iteration, but the overall iteration efficiency was very low. However, on average, the fitness value of IPSO approached 0.01 at the 10th generation, and truly converged at the 20th generation, saving 80 iterations. In addition, IPSO improved the stability of optimization for different models. In general, IPSO has higher iteration efficiency and better optimization results than PSO.

Table 2 showed the average value of multiple prediction results of BPNN, PSO-BPNN and IPSO-BPNN, which were evaluated by RMSE and DC. The prediction results of BPNN model showed a certain randomness, with the mean values of total RMSE and DC being 4.678 and 0.876 respectively. The RMSE mean values of the first, second and third observation points of PSO-BPNN were 0.304, 0.913 and 1.676, respectively. Compared with BPNN, the total RMSE mean was 0.964, which decreased by about 3.714. Meanwhile, the DC of all observation points was higher than 0.96, which increased by about 11%.

| Points     | RMSE          | DC             |     |
|------------|---------------|----------------|-----|
|            | BPNN | PSO-BPNN | IPSO-BPNN | BPNN | PSO-BPNN | IPSO-BPNN |     |
| R-Hip-1    | 2.868 | 0.263    | 0.160      | 0.937 | 0.998    | 0.998     |     |
| R-Hip-2    | 3.526 | 0.785    | 0.463      | 0.906 | 0.995    | 0.998     |     |
| R-Hip-3    | 2.811 | 1.485    | 0.837      | 0.945 | 0.984    | 0.995     |     |
| R-Knee-1   | 5.354 | 0.405    | 0.331      | 0.927 | 0.998    | 0.998     |     |
| R-Knee-2   | 8.350 | 1.104    | 0.942      | 0.814 | 0.997    | 0.998     |     |
| R-Knee-3   | 8.998 | 1.954    | 1.765      | 0.783 | 0.989    | 0.991     |     |
| R-Ankle-1  | 2.800 | 0.245    | 0.212      | 0.908 | 0.998    | 0.998     |     |
| R-Ankle-2  | 3.262 | 0.850    | 0.631      | 0.875 | 0.991    | 0.995     |     |
| R-Ankle-3  | 4.137 | 1.587    | 0.974      | 0.792 | 0.969    | 0.988     |     |
Figure 8. Results of IPSO for right hip, right knee and right ankle joint

Figure 8 showed the angle prediction results of IPSO-BPNN, where the solid line and the dotted line represented the actual observed value and the predicted value of the model respectively. Compared with PSO-BPNN and BPNN, the prediction results of IPSO-BPNN at each observation point were closer to the actual observation values. The RMSE mean value of the test results was 0.702, which was about 3.976 lower than BPNN, and 0.262 lower than PSO-BPNN. The minimum DC was increased to 0.98, that was, the predicted result of IPSO-BPNN was more than 98% similar to the actual observed value in the future on the whole, which was about 10.4% higher than that of BPNN and about 1.6% higher than that of PSO-BPNN. Besides, from the first prediction observation point to the third prediction observation point, the RMSE of IPSO-BPNN risen by a step with a difference value close to 0.3, and the DC declined slightly. The further away from the current observation point, the worse the evaluation of the prediction results. In conclusion, IPSO showed better prediction results than BPNN and PSO, which was conducive to improving the collaborative control performance of the exoskeleton robot.

4. Discussion

The active movement of the wearer was passively assisted by the exoskeleton robot, and the delay of each system would inevitably make the robot control later than the human movement. Therefore, a motion prediction algorithm based on IPSO-BPNN was proposed in this paper, and the relevant evaluation criteria were established. The prediction results were compared with those of BPNN and PSO-BPNN, which showed that IPSO-BPNN could make better prediction of human lower limb motion joint angle.

During data preprocessing, adding time span to reconstruct time series was a resampling of original data, which compressed and extracted characteristic data. This made the prediction time controllable and significantly longer than the system delay, thus enabling the entire prediction model to establish a positive impact on collaborative control.

Compared with PSO, IPSO has a higher optimization efficiency for BPNN model and was superior to PSO in terms of iterative convergence speed and final optimization results. IPSO improved the basic particle movement process of PSO, adding two cognitive inflection points on the basis of the original search path, which were two search paths to strengthen the historical cognition and individual cognition respectively. These three search paths respectively strengthened the ability of historical inheritance, local search and global search, thus enlarged the search space of particle swarm laterally. Besides, the feedback from the BPNN model to IPSO was added to improve the complexity of the search path vertically. Finally, a competitive optimization model based on three search paths was formed, which accelerated the updating of individual particle extremum. The rapid iteration of individual particles further accelerated the convergence of particle swarm. In addition, the subsequent mutation operation enhanced the algorithm's ability to avoid local optimization. The results showed that the above
improvement steps promoted the rapid convergence of particle swarm and improve the optimization results.

The above results were based on the conclusions of limited experimental subjects. Due to individual differences, different objects may have different prediction biases. Therefore, we will collect exercise data of healthy male and female subjects of different ages in the future to further verify the prediction model. In addition, the data collection in this paper is carried out in an ideal environment, but the actual movement environment of human is more complex. The current model may not be able to adapt to the more complex motion environment, and the subsequent research will focus on the motion prediction model for the complex motion environment.

5. Conclusions
In order to solve the problem that the machine moves later than the human in exoskeleton robot, a motion prediction algorithm based on IPSO-BPNN was proposed to realize the man-machine collaborative control. Firstly, the original time series data was reconstructed by adding time span. Then, the BPNN model optimized by IPSO was used to predict the future joint angles. Finally, the evaluation criterion with prediction duration, iteration efficiency, RMSE and DC as the core was used to analyze the prediction results. The results showed that the predicted duration of IPSO-BPNN covered the system delay, and it converged 80 times earlier than PSO-BPNN. In addition, IPSO-BPNN was also superior to BPNN and PSO-BPNN in the evaluation of prediction results. The average RMSE decreased by 3.976 and 0.262 respectively, and the average DC increased by 10.4% and 1.6% respectively. The method proposed in this paper has wide potential application in human-machine smooth motion control and can effectively improve human-machine coordination performance in the exoskeleton.

In the future, we will collect more data from healthy subjects of different ages to verify this method. Besides, more experiments under complex terrain will be added to further optimize IPSO-BPNN. Finally, it will be applied to the real time motion control of exoskeleton, and its effect on exoskeleton performance will be compared and analyzed through the evaluation criteria of exoskeleton performance.

Acknowledgments
The study was funded by grants from the Ministry of Science and Technology national key RD program (Grant Number: 2017YFB1300500).

Thanks to Wang Lina for helping provide experimental equipment and venues, and the subjects who participated in the experiment.

References
[1] Lee J W, Kim H, Jang J, et al. A simulation study on the control of lower extremity exoskeleton for assistance of human locomotion[C]//International Symposium on Robotics. IEEE, 2013:1-4.
[2] Xie, H.; Li, G.; Zhao, X.; Li, F. Prediction of limb joint angles based on multi-source signals by GS-GRNN for exoskeleton wearer. Sensors 2020, 20, 1104.
[3] He, W.; Li, Z.J.; Dong, Y.T.; Zhao, T. Design and adaptive control for an upper limb robotic exoskeleton in presence of input saturation. IEEE Trans. Neural Netw. Learn. Syst. 2018, 30, 97–108.
[4] Wu, X.; Li, Z. Cooperative manipulation of wearable dual-arm exoskeletons using force communication between partners. IEEE Trans. Ind. Electron. 2020, 67, 6629–6638.
[5] Campeau-Lecours, A.; Cote-Allard, U.; Vu, D.S.; Routhier, F.; Gosselin, B.; Gosselin, C. Intuitive adaptive orientation control for enhanced human–robot interaction. IEEE Trans. Robot. 2018, 35, 509–520.
[6] Ding M, Nagashima M, Cho S G, et al. Control of walking assist exoskeleton with time-delay based on the prediction of plantar force[J]. IEEE Access, 2020, PP(99):1-1.
[7] Zhiyu Zhao, Lei Ma, Yongkui Sun. Recognition of gait phase using plantar pressure sensors [J]. Electronic Measurement Technology,2019,42(13):26-31.
[8] Zhiyu Zhao. Gait prediction and recognition based on extreme learning machine [D]. Southwest Jiaotong University, 2019.

[9] Tao Qin. Research on human gait perception method of lower limb exoskeleton robot [D]. University of Electronic Science and Technology of China, 2019.

[10] Hailin He, Jianbin Zheng, Fangli Yu, Lie Yu, Enqi Zhan. Exoskeleton robot gait detection based on improved whale optimization algorithm [J]. Journal of Computer Applications, 2019, 39(07): 1905-1911.

[11] Shilei Li. Gait planning and control method of lower extremity exoskeletal robot [D]. Harbin Institute Of Technology, 2018.

[12] Congwei Yang. Research on the perception of body motion trend based on the solution and prediction of lower extremity joint torques [D]. Harbin Institute Of Technology, 2015.

[13] Wuyi Luo. Prediction of posture and movement state of human lower limbs [D]. Huazhong University Of Science And Technology, 2017.

[14] Yi Long. Human motion prediction and human-robot coordination control for lower extremity exoskeleton [D]. Harbin Institute Of Technology, 2017.

[15] Ma, X.; Wang, C.; Zhang, R.;; Wu, X. A Real-Time Gait Switching Method for Lower-Limb Exoskeleton Robot Based on sEMG Signals. In Proceedings of the 2018 International Conference on Communications in Computer and Information Science, Beijing, China, 30 November-1 December 2018; Springer: Singapore, 2019; Volume 1005, pp. 511-523.