Recognition of the Gait Phase Based on New Deep Learning Algorithm Using Multisensor Information Fusion

Fangzheng Wang, Lei Yan,[*] and Jiang Xiao

School of Technology, Beijing Forestry University,
No. 35, Tsinghua East Road, Haidian District, Beijing 100083, China

Gait phase recognition is an effective method of analyzing human motion and behavior that can be very meaningful in people's daily life, especially when struggling with assisted rehabilitation. In this paper, a new algorithm that can recognize a human gait phase more accurately is proposed. The new gait phase recognition algorithm is based on a deep memory convolutional neural network (DM-CNN) using multiple sensor fusion. We used the plantar pressure sensor array and acceleration sensor array gait data, and then extracted the gait features using the DM-CNN. The measured data of the continuous gait cycle were divided into unit steps, and the data were analyzed and preprocessed. Then, a feature map of each sensor array was extracted by constructing a separate DM-CNN. Finally, each feature map was combined into a fully connected network, and a memory function was introduced to simulate historical behavior. We then tested the algorithm on the phases of a gait cycle and compared the evaluation indicators of each phase. In the experiment, we compared single-mode and multimode recognition results, and compared those with the new hidden Markov model (N-HMM), K-nearest neighbor (KNN), and hidden Markov model (HMM) algorithms. The experimental results show that when the multisensor data are fused, the average recognition accuracy can reach 97.1%, which is higher than those of the other algorithms and improves the recognition of a human gait phase. The accurate recognition of human gait can provide a better theoretical basis for the design of exoskeleton robot control strategies.

1. Introduction

Accurate gait phase recognition is very important to some people's mobility, assisted rehabilitation, and everyday lives. Research on gait recognition can be used for the control of exoskeleton-assisted robots and smart wearable devices as well as rehabilitation for various diseases.[1] It is very helpful for a wearer's exercise therapy and rehabilitation to accurately identify a human movement phase. By recognizing the gait phase and selecting appropriate parameters to achieve safe and effective motion control, a device can act at the correct moment, helping people work with greater precision, safety, and stability.[2]

---

[*]Corresponding author: e-mail: mark_yanlei@bjfu.edu.com

Human gait phase recognition algorithms have mainly included threshold-based methods, time-frequency analysis, machine learning models, and combinations of these methods. Machine learning methods have the ability to learn to capture linear and nonlinear relationships and model them.[3]

Thus far, different types of sensor, including force-sensitive resistors (FSRs), air pressure sensors, inertial sensors, inclinometers, foot switches, and electromyography (EMG) sensors, are readily available in the industry and can be applied to gait analysis.[4] Recent advances in sensor technology have enabled the development of small low-cost wearable devices for a wide range of applications in physiology, biomechanics, and motion data collection.[5] In the field of gait phase recognition, sensors based on 3D optical systems exist, but have a limited application space and can only be applied indoors or under special conditions. The EMG sensor requires a complicated and specific mounting method that makes direct contact with the skin. Because EMG sensors are susceptible to sweat between the skin and the sensor, this reduces the accuracy of data collection.[6]

Machine learning algorithms are widely used in the field of gait phase recognition. Ylli *et al.* introduced a method of gait phase recognition using the hidden Markov model (HMM) to recognize the human motion phase, an inertial measurement unit (IMU) sensor was connected to the leg, and a foot pressure sensor was placed on the sole of the foot.[7] Lerner *et al.* and Torrealba *et al.* used the motion data of the human leg and knee angle acquired by the IMU sensor, and the gait phase was identified by a machine learning algorithm.[8,9] Huang *et al.* used the foot pressure sensor and other techniques to obtain the body stride length.[10] The walking pattern of stroke patients with Parkinson's disease was analyzed using a support vector machine (SVM), a random forest, or a mixed model. Yuan *et al.* used two depth cameras to obtain the trajectory of the rotation angle and the overall velocity of the selected body joint, and pathological gait was classified using the K-nearest neighbor (KNN) classifier.[11] Lim *et al.* proposed a method of distinguishing gait patterns between normal and Parkinson's disease patients using artificial neural networks.[12] However, 37 reflective markers should be attached during the experiment, and data were collected using six infrared cameras, so this method is limited for use in indoor environments. Zhou *et al.* and Bayón *et al.* used the three acceleration axes (*x*-, *y*-, and z-axes) data acquired using a smartphone to identify three walking modes using a one-dimensional convolutional neural network algorithm.[13,14] Because the mobile phone was placed in the hand or pocket during data collection, the movement of the hand or pocket during walking made it difficult to accurately measure the gait data. Altilio *et al.* and Aoike *et al.* used the decision tree algorithm to classify the gait types of the elderly using pressure sensors, acceleration, and gyro data.[15,16] However, because the method was classified by the experimenter according to the set conditions, there was a problem of incorrect classification of some types of gait. Xin *et al.* used discriminant analysis algorithms and a pressure sensor to classify gait; however, because only one pressure sensor was used, it could not accurately reflect the characteristics of the gait phase.[17,18]

To obtain more accurate gait information and improve the recognition accuracy of the gait phase, we propose a DM-CNN gait phase recognition algorithm in this paper. We collected

motion data from 10 healthy test subjects, including plantar pressure and leg acceleration data, and fused these data to recognize the motion phase. Then, we constructed a DM-CNN. The method includes dividing the continuous gait data into one cycle step size, then preprocessing the data and identifying the gait type. Features were extracted by using a DM-CNN. In the preprocessing stage, the continuous gait data were accurately segmented according to the characteristics of the phase of the gait cycle. In the classification phase, we constructed multiple features for the foot pressure and inertial sensors to extract the data feature map for each sensor. Then, a DM-CNN was constructed to recognize the gait phase based on the feature map. The experimental results show that when the data are fused on the basis of multiple sensors, the average recognition rate of the algorithm is as high as 97.1%, which can accurately recognize the gait of the human body, improve the recognition effect of the gait stage, and provide a theoretical basis for the accurate control of the exoskeleton.

The organizational structure of the rest of this paper is as follows. In Sect. 2, we describe the data acquisition system and experiment. In Sect. 3, we describe the DM-CNN. In Sect. 4, we detail the results and analysis. Section 5 is the conclusion.

2. Materials and Experiments

2.1 Acquisition system

The gait data acquisition system consists of two types of sensor, namely, acceleration and pressure sensors. The acceleration sensors acquire human leg motion data; two sensors are affixed in the middle of the calf and two sensors in the middle of the thigh for a total of four. Pressure sensors are used to sense the pressure on the sole of the foot. Eighteen pressure sensors are placed on the sole of the foot. These two types of sensor can be used together to obtain accurate gait information and effectively detect the state of human motion. Figure 1 shows the data acquisition process of wearable sensors.
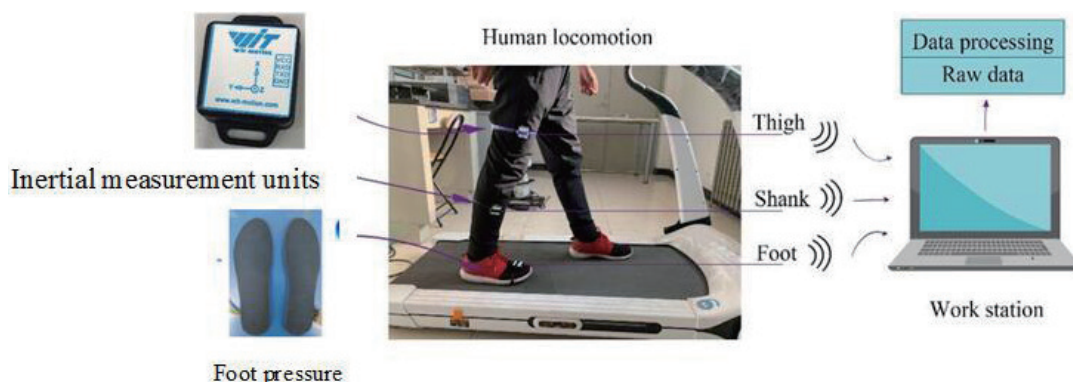

Fig. 1.   (Color online) Data collection using wearable sensors.

2.1.1   Acceleration sensor

BWT901CL IMUs are used, and their performance parameters are shown in Table 1. Each IMU integrates high-precision gyroscopes, accelerometers, and geomagnetic sensors to measure three-dimensional acceleration and three-dimensional angular velocity.[19]   With a built-in high-performance microprocessor, an advanced dynamic solution, and the Kalman filtering algorithm, each IMU not only rapidly solves the real-time motion attitude of the module, but also effectively reduces the measurement noise.[20]

2.1.2   Plantar pressure sensor

The pressure sensor used in the gait information acquisition system is FSR402. The greater the pressure applied to the sensing element, the smaller the resistance of the internal circuit, resulting in a higher output voltage. The allowable pressure range for such sensors is typically from 100 g to 10 kg of gravity.[21,22]   Table 2 shows the characteristics of the FSR402 pressure sensor. To collect more plantar pressure information during the walking phase, nine pressure sensors are affixed to each foot for a total of 18 plantar pressure sensors.

2.2   Division of gait phase

Gait is a periodic cycle from the first contact of one foot on the ground to the second contact of the heel on the ground. Usually, a gait cycle is divided into two main phases, the stance and swing phases. Heel striking the ground and the toe off the ground mark the beginning of the stance and swing phases, respectively.[23]

During the walking cycle, each leg in turn supports the person and moves him or her forward. As shown in Fig. 2, there are four gait stages, namely, heel strike (HS), foot flat (FF),

Table 1
Specifications of BWT901CL.

Index	Parameter
Voltage (V)	3.3–5
Current (mA)	<40
Volume (mm × mm × mm)	51.3 × 36 × 15
Frequency (Hz)	0.1–200

Table 2
Characteristics of the FSR402 pressure sensor.

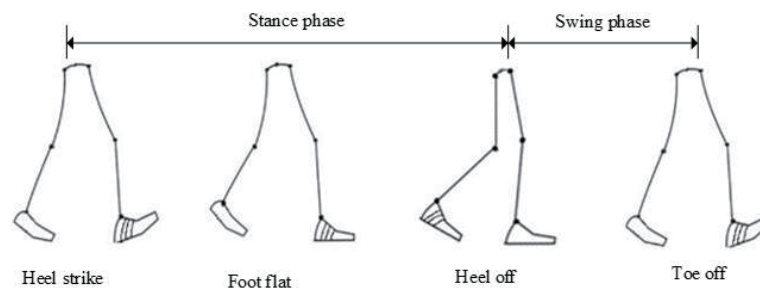| Index	Parameter
Force range (N)	0–110
Sensing area (cm)	2.54
Size [length (cm) × width (cm)]	5.68 × 3.18
Linearity error (%)	3

Fig. 2.   Distribution of foot pressure in a cycle. HS: heel strike; FF: foot flat; HO: heel off; TO: toe off.

heel off the ground (HO), and toe off the ground (TO). Initial contact is a short-lived step that shows the interaction between the heel and the ground. During the loading response, the leg absorbs the impact until the forefoot falls on the ground.[24] When standing in the middle, the foot is stationary and supports the weight because the other foot begins to swing. When the heel is off the ground, the terminal posture begins and continues while the toe still touches the ground. Once the toe is off the ground, the cycle enters the swing phase.[25]

2.3　Description of experiment

In the experiment, exercise data of 10 healthy test subjects were collected (age: 26 ± 2 years old; height: 170 ± 7 cm; weight: 60 ± 8 kg; shoe size: 24.5 ± 2.5 cm). All subjects were healthy, able to walk normally, and had no gait disturbances. Written informed consent was obtained before the start of the experiment. Specific characteristics of the test subjects (age, height, weight, and shoe size) are shown in Table 3. For precision, the materials of the shoes used in the experiment were the same.

On a treadmill, each test subject was tested three times at a speed of 2.8 km/h with a recording interval of three minutes per test. In the experiment, each participant walked an average of 300.0 steps. All the motion data signals obtained from each participant were mixed together to construct a larger data set to extract different motion features for training the network model, and a total of 3000 walking cycle data were collected.

2.4　Gait data processing

Wavelet decomposition performs time-frequency transformation on the signal.[26] By shifting and stretching, high-frequency noise can be separated from the motion gait data. By smoothing the processing, irregular points in the data can be eliminated, and information can be better obtained from the data.

For plantar pressure data acquisition, nine pressure sensors were placed on each foot for a total of 18 foot pressure sensors. For pressure sensors, the values are 0 or 1, depending on the pressure strength. A value of 0 indicates that there is no pressure, that is, the swing stage. A value of 1 indicates a pressure value, that is, when the foot touches the ground. To divide the

Table 3
Test subject characteristics.

Subject	Sex	Age (year)	Height (cm)	Weight (kg)	Shoe size (cm)
1	Male	28	177	58	26.0
2	Male	26	175	60	26.0
3	Male	24	173	56	26.5
4	Female	27	164	52	22.5
5	Male	25	163	55	23.5
6	Male	26	165	54	24.5
7	Male	28	174	62	27.0
8	Female	27	167	64	23.5
9	Male	26	174	66	24.5
10	Male	27	176	68	23.5

measured data into unit steps, we divided the length of unit steps according to the standing and swing phases of the gait cycle. Gait data samples are stored in matrix form in each pressure sensor array and acceleration sensor array. The columns and rows of the matrix are the indices for each sensor measurement. Therefore, each foot pressure acquisition device consists of nine sensors for a total of 18 columns. The acceleration sensor has a total of four columns.

To segment the gait data of each cycle, all unit steps were adjusted according to the time of the shortest unit step ($t = 60$) to standardize the length of all gaits. Therefore, the normalized unit step measurements of the pressure and acceleration sensor arrays were converted to 60×18 and 60×12 arrays, and then stored as $1080 \times 1720 \times 1$ vectors using the dictionary sorting operator (X). Figure 3 shows the raw and normalized data for the unit steps.

3. Design of Deep Memory Convolutional Neural Network (DM-CNN)

3.1 Neural network architecture

Because human gait is a continuous operation, the data collected is also continuous, leading to the correlation between sensor measurements at each time point becoming large. Because the convolutional neural network does not have any time concept, it only considers the current example X and could not remember the previous data. Therefore, we propose the use of a DM-CNN. By introducing a memory network to simulate historical behavior, memory encodes information observed in the past, and it can use data correlation to recognize gait types based on different sensor measurements. The model relies not only on the signals acquired at time t, but also on the historical data of the signals $X_d = [X_{t-k}, ..., X_t]$ to recognize current and future gaits.

In general, DM-CNNs consist of feature extractors and fully connected networks. Each feature extractor consists of the following three layers: the filter, nonlinear activation function, and the feature pool layers. Figure 4 shows the overall structure of the gait phase recognition method.

In a fully connected neural network, information flows back and forth from the lower layer to the higher layer of the network, allowing it to learn high-level representations of the
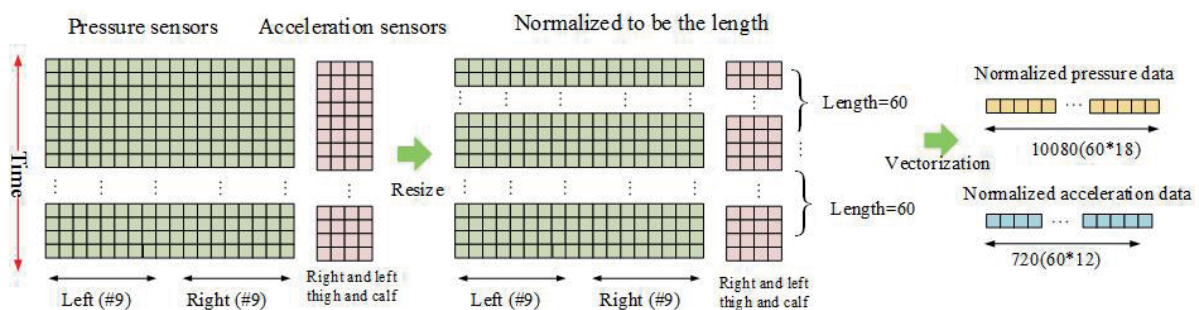

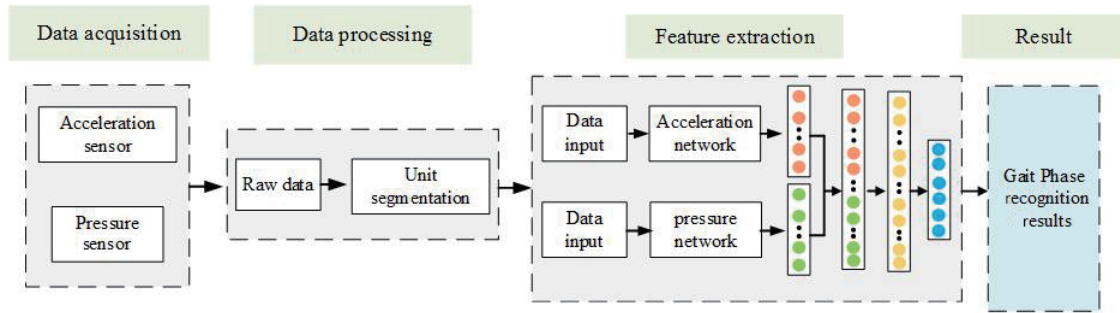Fig. 3. (Color online) Raw and normalized data for unit steps.

Fig. 4. (Color online) Overall structure of gait phase recognition method.

input data. A similar process has been observed in recurrent neural networks (RNNs). The difference is that the network depends on not only the input X, but also the activation function of the hidden unit in the previous time step. Therefore, these networks learn to map input sequences $X_d = [X_{t-k}, ..., X_t]$ and output sequences $O_d = [O_{t+1}, ..., O_{t+n}]$, as shown in Fig. 5. According to the characteristics of RNNs, these networks have the ability to accurately predict dense gait events. We created a neural network architecture that simulates the memory of an RNN.

To simulate the memory of the RNN, the input tensor x is based on the following equation.

$$M = \left[X_{(t-d):t} \right]_{t=d}^{O} \tag{1}$$

Here, $X_{(t-d):t}$ is a sequence of windows from beginning to end, ranging from $t - d$ to t, where t is the number of samples in the signal.

3.2 Network configuration

3.2.1 Input data format

The DM-CNN receives data in the form of a two-dimensional array and performs convolution operations using various filters in the convolutional layer. The data are preprocessed and the measurement data for each sensor array are normalized to $t \times W$ size. These standardized data are used as inputs to the DM-CNN. W is the number of sensors in the sensor array; t is 60. For pressure and acceleration sensor arrays, the W values are 18 and 12, respectively. To determine the number of steps required to extract the gait feature to distinguish the gait type, a one-step data sample is used for the classification experiment. The gait data sample of a cycle is defined as one step, and the input data size of the DM-CNN is $(t \times 1) \times W$.
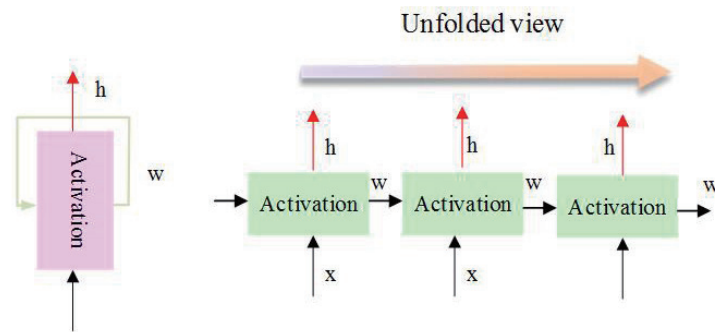
Fig. 5. (Color online) Information flow of the deep memory neural network. The left figure shows the RNN infinite loop network with the output of the model as input feedback. The right figure is the unfolded representation of an RNN.

3.2.2 Convolution layer

The DM-CNN used in this method consists of three convolutional layers. Figure 6 shows the structure of the single DM-CNN for each sensor array. Each layer contains filters for the corresponding functional level. For each layer of convolution, the following three parameters are determined: the number of filters, f, the filter size $w \times h$, and the span s. In the first convolutional layer, 32 filters are used, and the size of the filter is set to $w \times 20$. The DM-CNN input data are one cycle of gait data. The second and third convolution layers use 64 and 128 types of filter, respectively. All filters operate at a location in the input data to generate a feature vector.

The DM-CNN is trained using a back-propagation algorithm. In the learning process, when the number of neural network layers increases, the weight difference in the error back-propagation is close to zero. To prevent this phenomenon, we used the rectified linear unit (ReLU) as the activation function. To prevent the internal covariance shift caused by the nonlinear activation function used in each layer of the neural network, applying batch normalization after the batch activation function enhances learning stability.

3.2.3 Fully connected network and output

Each feature map is converted into a DM-CNN feature vector (flattened) as an input to a fully connected network. The output layer consists of the same number of nodes as the class to be identified. On each node, the weights of the previous layer are applied to the SoftMax function to calculate the final output. In this paper, single- and multimode DM-CNNs are constructed according to the number of sensor arrays. In the single-mode DM-CNN, the feature vectors extracted from a sensor array are employed as an input to a fully connected network. In the multimode DM-CNN, the feature vectors of the two sensor arrays are connected together as an input to a fully connected network. Figure 7 shows the structure of two mode networks that use single- and multimode DM-CNN feature vectors as inputs to determine the gait phase.
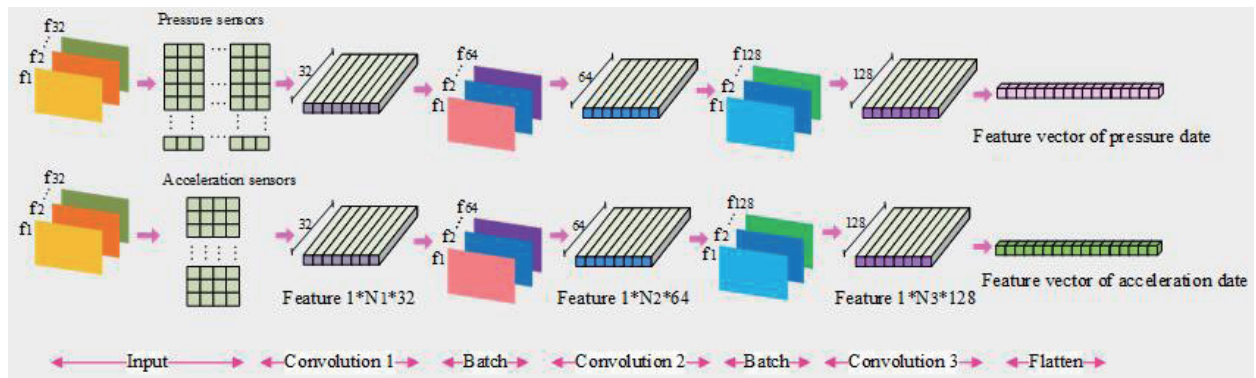
Fig. 6.　(Color online) Structure of the DM-CNN for each sensor array.
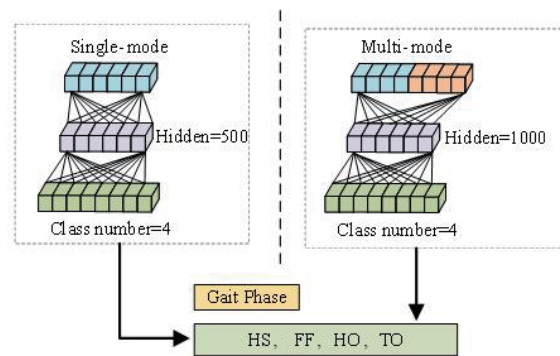

Fig. 7.　(Color online) Fully connected network structure that uses single- and multimode DM-CNN feature vectors as inputs to determine gait phase.

3.3　Evaluation indicators

To test the recognition of a human gait phase based on the DM-CNN algorithm, we selected the precision, recall rate, and accuracy indicators to evaluate gait recognition performance. The precision indicator considers the impact of the sample distribution on the detection rate. The recall rate indicator refers to the proportion of positive samples correctly identified to the total number of positive samples, which is regarded as the detection rate of a single gait stage.[27] The accuracy indicator is the result of positive samples detected by the classifier and the proportion of real positive samples. The calculation formulas are as follows:

$$\text{Precision: } P = \frac{TP}{TP + FP}, \tag{2}$$

$$\text{Recall: } R = \frac{TP}{TP + FN}, \tag{3}$$

$$\text{Accuracy: } A = \frac{TP + TN}{TP + TN + FP + FN}. \tag{4}$$

4.   Results

4.1   Results and analysis: Part 1

To evaluate the recognition performance of single- and multimode data fusion based on the DM-CNN algorithm, 1500 samples were randomly extracted from the data set to establish a training and test set, 1000 samples were selected for the training set, and 500 samples for the test set. To increase the statistical confidence, the above procedure was repeated 20 times to calculate the average recognition accuracy, as shown in Figs. 8(a) and 9(a). We also performed a *k*-fold cross validation experiment, as shown in Figs. 8(b) and 9(b). For the *k*-fold cross validation, the total training samples were randomly divided into *k* equal-sized subsets. In the *k* subset, a separate subset was retained as the validation set for the validation model, while the
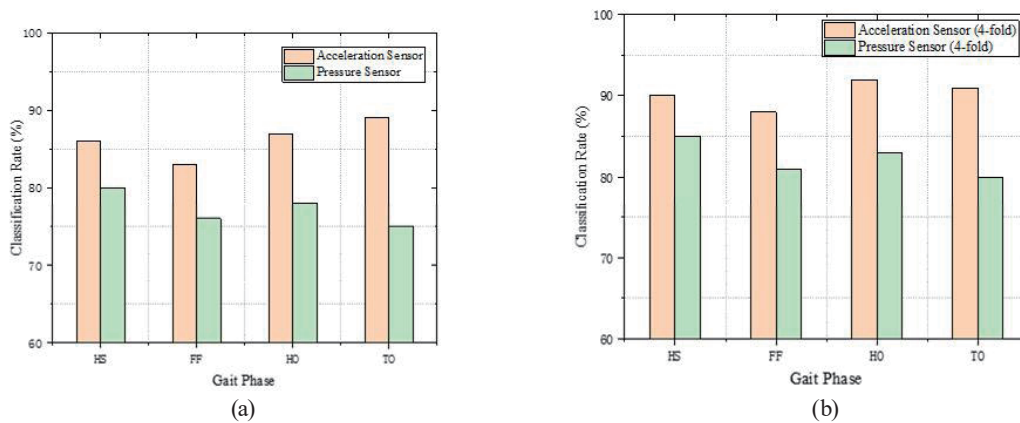

Fig. 8.   (Color online) Average accuracy of DM-CNN phase recognition for single mode. (a) Randomly selected 1000 training samples and 500 test samples; (b) DM-CMM recognition rate for single-mode 4-fold cross validation.
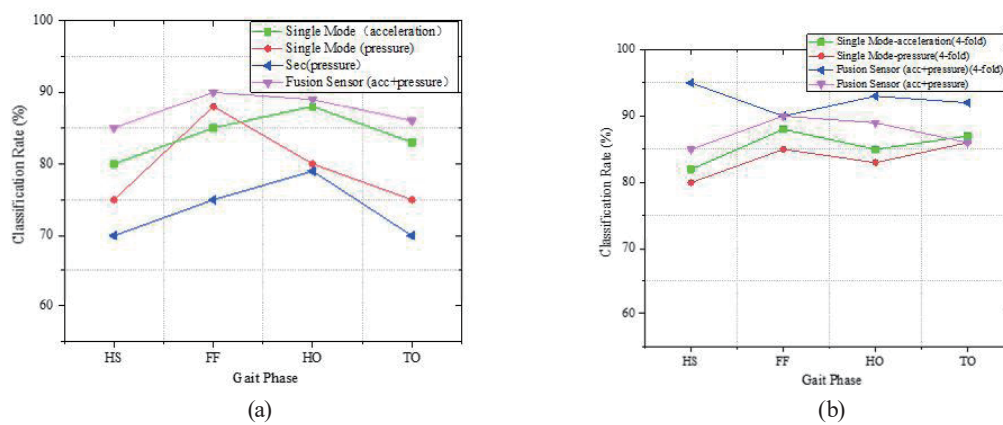

Fig. 9.   (Color online) Average accuracy of DM-CNN phase identification for multimode. (a) Randomly selected 1000 training data samples and 500 test set data samples; (b) 4-fold cross validation.

remaining $k-1$ subsets were used as the training set. Since k is usually set to the number of classes, we performed a 4-fold cross validation in the experiment.

The recognition performance was evaluated using a single-mode DM-CNN experiment of a sensor array data learning network and a multimode DM-CNN experiment using a data learning network of two sensor arrays. The single mode refers to the recognition result obtained using a sensor and the multimode refers to the result obtained using two sensor arrays together. Figure 8 shows the results of gait phase recognition using a single-mode DM-CNN, which trains each sensor array independently. In Fig. 8(a), the average recognition performance of the single-mode acceleration sensor is slightly higher than that of the pressure sensor array. The gait information generated by acceleration motion can be considered better for gait phase recognition. Figure 8(b) shows the results of gait phase average recognition based on 4-fold cross validation using 1000 training and 500 test samples. The average recognition rate is between 85 and 90%, and the gait recognition rate is higher than that shown in Fig. 8(a).

Figure 9 shows experimental results for single- and multimode DM-CNNs. To clearly show the effectiveness of the method, we compared it with that of Sec *et al.*'s method, the latest method using smart insoles.[28] Since Sec *et al.*'s method uses only pressure sensors, we first compared a single-mode DM-CNN with Sec *et al.*'s method, and the average recognition rates of single-mode DM-CNNs were 6% higher than those of Sec *et al.*'s method. At the same time, a multimode DM-CNN was compared with a single-mode DM-CNN, its average recognition rate is 8% higher than that of the single mode. The results in Fig. 9(a) show that the multimode DM-CNN has a higher gait recognition than the single-mode DM-CNN. In the 4-fold cross validation experiment, as shown in Fig. 9(b), the multimode DM-CNN gait phase average recognition accuracy of the 4-fold cross validation is as high as 97.1%, and the average accuracy is 7.9% higher than that without cross validation.

In Figs. 8 and 9, the experimental results show that using two types of sensor array (multimode) can achieve a higher recognition performance than using only one type of sensor array. Moreover, the performance of the multimode DM-CNN gait phase recognition with 4-fold cross validation is more accurate.

4.2   Results and analysis: Part 2

To evaluate the effectiveness of the algorithm in gait phase recognition, we compared the recognition effects of the HMM, K-KNN, and new hidden Markov model (N-HMM) algorithms applied in the gait phase and analyzed their performance characteristics. The acceleration signals of human leg and foot pressure data were used for training. These data were derived from the movement data of 10 test subjects on a treadmill. Using the same training set and test set data, the gait stage recognition model based on the DM-CNN, HMM, N-HMM, and KNN was used in the 4-fold cross validation. The trained model was used to identify the test data and compare the gait recognition results of the HMM, N-HMM, and KNN. The precision, recall rate, and average recognition accuracy are shown in Table 4 and Fig. 10.

As shown in Fig. 10(a), the DM-CNN is higher in the recall indicators of HS, FF, HO, and TO than the N-HMM, HMM, and KNN in the gait stage, which shows that the DM-CNN has

Table 4
Comparison of average recognition accuracy of algorithms.

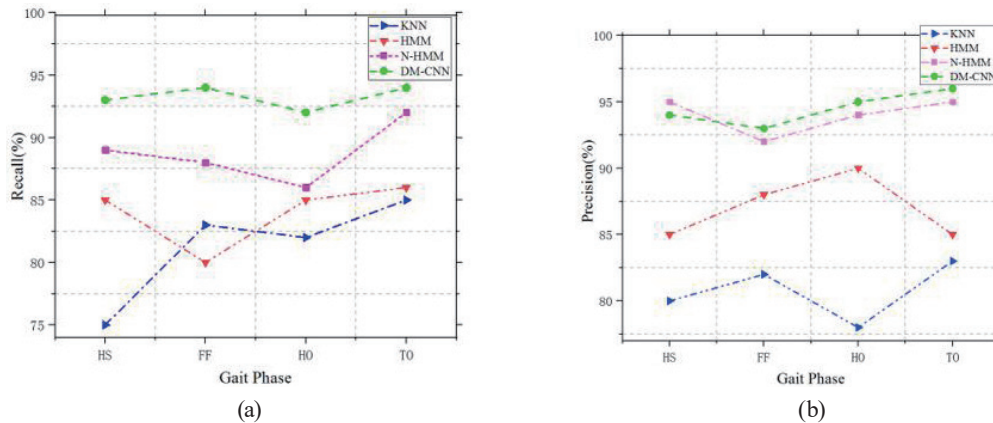| Algorithm	HS	FF	HO	TO	Average accuracy
DM-CNN	97.2%	97.1%	96.8%	97.3%	97.1%
N-HMM	96.5%	96.3%	95.4%	96.6%	96.2%
HMM	92.3%	93.4%	89.5%	94.2%	92.3%
KNN	90.2%	85.6%	86.5%	93.1%	88.5%

Fig. 10.   (Color online) Evaluation results of different algorithms: (a) evaluation of recall rate experimental results; (b) evaluation of precision rate experimental results.

a better recognition effect for these stages. The average recall rate of the DM-CNN is 94.5%, whereas that of the N-HMM is 93.5%, that of the HMM is 90.5%, and that of the KNN is 82.5%. Figure 10(b) shows that the average precision rate of the DM-CNN is 95.9%, whereas that of the N-HMM is 95.5%, that of the HMM is 92%, and that of the KNN is 81.5%. From these results, we can also conclude that the DM-CNN has a better gait phase recognition than the N-HMM, HMM, and KNN. Table 4 shows that the average recognition accuracy rate of the DM-CNN is 97.1%, that of the N-HMM is 96.2%, that of the HMM is 92.3%, and that of the KNN is 88.5%. Therefore, the experimental results show that the DM-CNN proposed in this paper has a higher gait phase recognition effect than other algorithms.

5.   Conclusion

Gait phase recognition plays an important role in motion analysis, identity recognition, and other fields. In this paper, a gait recognition algorithm based on a DM-CNN is proposed. The algorithm uses the fusion of human body motion acceleration and plantar pressure data to extract features from the fused data and accurately identify the gait phase. Feature maps were extracted from the preprocessed data, and each feature map was independently learned for each sensor array using a DM-CNN. We combined these feature maps and used them as inputs to a fully connected network to recognize gait phases. Data from 10 healthy test subjects were collected for recognition experiments. The 4-fold cross validation method was used. It

was found that the recognition performance of the multimode DM-CNN with the two sensor arrays is higher than that of the single-mode DM-CNN with one sensor array. Furthermore, the recognition performance of the multimode DM-CNN was compared with those of the N-HMM, HMM, and KNN. The experimental results show that the precision, recall rate, and accuracy of the DM-CNN based on multi-sensor fusion are higher than those of the other algorithms. The average recognition accuracy of the DM-CNN is 97.1%, which is higher by 0.9, 4.8, and 8.6%. The proposed algorithm can recognize human gait more effectively than other algorithms, which helps lay the theoretical foundation for the development of smart wearable robots.

In future work, researchers need to study the phases of different motion conditions, such as walking, climbing stairs, and uphill and downhill slopes. In addition, more work needs to be done to determine whether more sensors will be needed to recognize the phase under different motion conditions.

Acknowledgments

References

1 D. H. Lim, W. S. Kim, and H. J. Kim: Int. J . Precis. Eng. Manuf. **18** (2017) 681. https://doi.org/10.1007/s12541-017-0081-9
2 R. Atta, S. Shaheen, and M. Ghanbari: Pattern Recogn. **69** (2017) 213. https://doi.org/10.1016/j.patcog.2017.04.015
3 C. Xi, H. Guang, S. Xin, and W. Jinkuan: IEEE Trans. Biomed. Eng. **64** (2017) 2618. https://doi.org/10.1109/TBME.2017.2653246
4 K. Ylli, D. Hoffmann, A. Willmann, P. Becker, B. Folkmer, and Y. Manoil: Smart Mater. Struct. **24** (2015) 1. https://doi.org/10.1088/0964-1726/24/2/025029
5 W. C. Hua, H. F. Mao, H. J. Sheng, W. T. Yun, and H. W. Li: J. Neuroeng. Rehabil. **15** (2018) 14. https://doi.org/10.1088/0964-1726/24/2/025029
6 B. Hang and D. Jeon: Int. J. Control. Autom. Syst. **16** (2018) 1. https://doi.org/10.1007/s12555-016-0545-1
7 K. Ylli, D. Hoffmann, A. Willmann, P. Becker, B. Folkmer, and Y. Manoil: Smart Mater. Struct. **24** (2015) 1. https://doi.org/10.1088/0964-1726/24/2/025029
8 Z. F. Lerner, D. L. Damiano, and H. S. Park: IEEE Trans. Neural Syst. Rehabil. Eng. **25** (2016) 650. https://doi.org/10.1109/TNSRE.2016.2595501
9 R. R. Torrealba, S. B. Udelman, and E. D. Fonseca: Mech. Mach. Theory **116** (2017) 248. https://doi.org/10.1016/j.mechmachtheory.2017.05.024
10 H. Huang, Z. Fan, L. J. Hargrove, D. R. Rogers, and K. B. Englehart: IEEE Trans. Biomed. Eng. **58** (2011) 2867. https://doi.org/10.1109/tbme.2011.2161671
11 L. Z. Yuan, C. Xiang, Z. Xu, and T. K. Yu: Int. J. Neural Syst. **27** (2017) 587. https://doi.org/10.1142/S0129065717500095
12 D. H. Lim, W. S. Kim, and H. J. Kim: Int. J . Precis. Eng. Man. **18** (2017) 681. https://doi.org/10.1007/s12541-017-0081-9
13 H. Zhou, N. Ji, O. W. Samuel, and Z. Zhao: Sensors **26** (2016) 1634. https://doi.org/10.3390/s16101634
14 C. Bayón, O. Ramírez, J. I. Serrano, M. D. Castillo, A. Pérez-Somarriba, J. M. Belda, C. Cifuentes, and A. Frizer: Rob. Auton. Syst. **91** (2017) 101. https://doi.org/10.1016/j.robot.2016.12.015
15 R. Altilio, M. Paoloni, and M. Panella: Med. Biol. Eng. Comput. **55** (2016) 1. https://doi.org/10.1007/s11517-016-1546-1
16 K. Aoike, K. Nagamune, K. Takayama, R. Kuroda, and M. Kurosaka: Proc. 2016 IEEE Int. Conf. Systems, Man, and Cybernetics (2016).

17 L. D. Xin, W. X. Yu, D. W. Bin, W. Can, and X. T. Tian: Sensors **16** (2016) 156. https://doi.org/10.3390/s16101579

18 A. J. Young and D. P. Ferris: IEEE Trans. Neural Syst. Rehabil. Eng. **25** (2017) 171. https://doi.org/10.1109/TNSRE.2016.2521160

19 Y. T. Fang, A. Parri, V. R. Garate, M. Cempini, R. Ronsse, and N. Vitiello: Autom. Constr. **41** (2017) 1. https://doi.org/10.1007/s10514-016-9566-0

20 K. Ylli, D. Hoffmann, A. Willmann, P. Becker, B. Folkmer, and Y. Manoil: Smart Mater. Struct. **24** (2015) 1. https://doi.org/10.1088/0964-1726/24/2/025029

21 D. J. Hyun, H. Lim, S. Park, and K. Jung: Int. J. Precis. Eng. Manuf. **18** (2017) 803. https://doi.org/10.1007/s12541-017-0096-2

22 Y. Miao, G. Feng, and P. Da : J. Bionic. Eng. **11** (2014) 249. https://doi.org/10.1016/S1672-6529(14)60034-2

23 A. Ghebleh and M. E. Moghaddam: Multimedia Tools Appl. **77** (2018) 8237. https://doi.org/10.1007/s11042-017-4712-z

24 A. Hayfron, M. S. Nixon, and J. N. Carter: Pattern. Recogn. Lett. **24** (2003) 2175. https://doi.org/10.1007/3-540-45344-X_40

25 B. Chen, X. Zhao, H. Ma, L. Qin, and W. Liao: Smart. Mater. Struct. **26** (2017) 10. https://doi.org/10.1088/1361-665X/aa8343

26 S. Hussain, S. Q. Xie, and P. K. Jamwal: Rob. Auton. Syst. **61** (2013) 911. https://doi.org/10.1016/j.robot.2013.01.007

27 S. Bingli, Z. Zhan, X. Liu, B. Hu, and T. Zhu: Gait Posture **58** (2017) 428. https://doi.org/10.1016/j.gaitpost.2017.09.001

28 W. D. Sec, S. S. Lee, W. Y. Shin, and S. I. Choi: J. Korea Soc. Comput. Inf. **33** (2018) 17. https://doi.org/10.9708/jksci.2018.23.02.017