Mathematical Biosciences and Engineering

MBE, 18(4): 3521–3542.
DOI: 10.3934/mbe.2021177
Received: 02 February 2021
Accepted: 20 April 2021
Published: 23 April 2021

http://www.aimspress.com/journal/MBE

Research article

Multi-feature gait recognition with DNN based on sEMG signals

Ting Yao, Farong Gao*, Qizhong Zhang and Yuliang Ma

Institute of Intelligent Control and Robotics, School of Automation, Hangzhou Dianzi University, Hangzhou 310018, China

* **Correspondence:** Email: frgao@hdu.edu.cn; Tel: +8657186919108.

Abstract: This study proposed a gait recognition method based on the deep neural network of surface electromyography (sEMG) signals to improve the stability and accuracy of gait recognition using sEMG signals of the lower limbs. First, we determined the parameters of time domain features, including the mean of absolute value, root mean square, waveform length, the number of zero-crossing points of the sEMG signals after noise elimination, and the frequency domain features, including mean power frequency and median frequency. Second, the time domain feature and frequency domain feature were combined into a multi-feature combination. Then, the classifier was trained and used for gait recognition. Finally, in terms of the recognition rate, the classifier was compared with the support vector machine (SVM) and extreme learning machine (ELM). The results showed the method of deep neural network (DNN) had a better recognition rate than that of SVM and ELM. The experimental results of the participants indicated that the average recognition rate obtained with the method of DNN exceeded 95%. On the other hand, from the statistical results of standard deviation, the difference between subjects ranged from 0.46 to 0.94%, which also proved the robustness and stability of the proposed method.

Keywords: gait recognition; multi-feature fusion; deep neural network (DNN); robustness and stability

1. Introduction

Gait is a complex, continuous, and periodic movement process completed under the coordination of various tissues of the body [1]. It refers to the movement and balance of the human body when walking upright and the most basic manner of lower limb movement. The use of surface electromyography (sEMG) signals to classify and recognize different movements has been extensively studied. sEMG signals contain a wealth of muscle movement information, which can

be used to distinguish different stages of limb movements to realize gait recognition [2,3]. sEMG signals are also important for the detection muscle activity. In addition, sEMG signals reflect neuromuscular activity to a certain extent; thus, they have important use value and prospect in clinical and rehabilitation medicine [4].

Gait recognition using sEMG signals is mainly divided into two steps: feature extraction and classification recognition. Feature extraction methods of sEMG signals generally have four categories: time domain (TD), frequency domain (FD), time-frequency domain, and nonlinear features [5]. The TD feature extraction method is simple, efficient, and widely used in gait recognition [6]. However, given that sEMG signals are easily disturbed by fatigue and other factors, the TD feature suffers from severe mutations and relatively poor stability [7]. The analysis of sEMG signals by the extraction of FD features shows great vitality [8]. Sbriccoli et al. [9] used the median frequency (MF) of sEMG signals as the FD feature to study the spectral characteristics of the sEMG signals generated by biceps brachii muscle during arm movement; they achieved good results. Pancholi et al. [10] proposed the method to extract the time domain and frequency domain features of sEMG signals and used different classifiers to classify and recognize various muscles, achieving good recognition results.

The sEMG signal is a kind of weak bioelectric signal. Research on sEMG-based recognition algorithm can be used in movement gait recognition and motion intention recognition [11]. With the advancement of measurement technology and machine learning, which are mainly used for pattern recognition algorithms, such as support vector machine (SVM), linear discriminant analysis (LDA), and extreme learning machine (ELM) [12,13], Dhindsa et al. [14] proposed performance evaluation of various classifiers for predicting knee angle via electromyography signals. Wu et al. [15] proposed single-channel sEMG signals of upper limbs through SVM classification recognition. Castllini et al. [16] used Gaussian kernel SVM, neural network, and local weighted projection regression in the classification process of forearm sEMG signals and compared the actual classification effects of several methods. The results showed the better recognition of SVM classification than the other methods. Kuang et al. [17] collected sEMG signals and extracted TD features to construct feature vectors and used the ELM algorithm to classify and recognize seven kinds of human actions. The results showed that ELM has a higher recognition rate than traditional back propagation (BP) neural networks. Traditional SVM is prone to local optimal solutions, resulting in unsatisfactory classification results, whereas ELM algorithms are prone to overfitting phenomena. A large number of improved algorithms have been developed for these problems. Xi et al. [18] proposed evaluation of feature extraction and recognition for activity monitoring and fall detection based on sEMG. Deng et al. [19] proposed a continuous-motion estimation algorithm combining principal component analysis (PCA) and regularized ELM. Gao et al. [20] proposed human gait recognition based on multiple feature combination and parameter optimization algorithms. Vu et al. [21] proposed a review of gait phase detection algorithms for lower limb prostheses, compared advantages and limitations between all the proposed methods and extracted the relevant questions and recommendations about gait detection methods for future developments. These algorithms achieve good results, and with the development of deep learning [22], gait recognition has shown great improvement in accuracy and model robustness. Christian et al. [23] proposed a deep learning approach to EMG-based classification of gait phases during level ground walking. Chen et al. [24] proposed surface EMG based continuous estimation of human lower limb joint angles by using deep belief networks, the result show that the root mean square error (RMSE) between the estimated joint angles. Simao et al. [25] proposed a review on electromyography decoding and pattern recognition for human-machine interaction, the novel

classification methods and approaches for detecting non-trained gestures are discussed. Mukhopadhyay et al. [26] used the method of DNN to classify upper limb gestures. Kim et al. [27] proposed a deep-learning gesture-recognition scheme using deep convolution and recurrent neural networks. Jeong et al. [28] proposed a deep joint spatiotemporal network (DJSTN) for facial expression recognition. The extracted feature from the appearance feature-based network is fused with the geometric feature in a hierarchical structure [29]. Khushaba et al. [30] used TD features to analyze the performance of DNN for sEMG classification. Deep time-growth neural network (DTGNN) increases the complexity of the classifier, whereas the DNN requires no complex feature extraction process [31]. Guo et al. [32] summarized about 330 contributions in this field by using a survey of face recognition based on deep learning, and discussed some open challenges and directions for future research. Li et al. [33] proposed electromyography based on PCA and deep learning to master the control of prostheses. Cote-Allard et al. [34] proposed deep learning for electromyographic hand gesture signal classification using transfer learning.

To improve the stability and accuracy of gait recognition, in this study, we focused on the periodic continuity of sEMG signals, considered the gait recognition of sEMG signals as the research object, and paid attention to the selection of multiple features, recognition accuracy, and algorithm stability. The study used the DNN as a classifier to extract features, such as TD and FD, and combined them to calculate the gait recognition rate. Then, the recognition results were compared with those of traditional SVM and ELM for analysis and discussion. This study also investigated from the DNN model including five parts: 1) Comparison of loss and recognition rates. 2) Losses of DNN models in various gait phases. 3) Analysis of precision rate, recall rate, and reconciliation parameters. 4) Analysis of standard deviation and variance. 5) Cross-validation in different state phases.

2. Materials and methods

Before sEMG signals were subjected to feature extraction, the original signal was de-noised to avoid signal distortion [35]. Then, we extracted the mean absolute value (MAV), root mean square (RMS), waveform length (WL), and zero-crossing (ZC) points from the TD [36,37], the average power frequency and MF from the FD [9], and other features.

2.1. Feature selection of sEMG signals

2.1.1. Time domain feature

Based on the TD features of sEMG, in this study, four TD features were selected: MAV, RMS, ZC, and WL [38]. MAV contains the average information intensity and the concentration of sEMG signals. During exercise, RMS represents the contribution of each muscle tissue. WL reflects the complexity of the sEMG waveform and the combined effects of sEMG amplitude, frequency, and duration. TD features are easy to operate in real-time control [6,39,40], and can be calculated in accordance with the following formulas:

$$\mathrm{MAV} = \frac{1}{N}\sum_{i=1}^{N}\left|x_i\right| \tag{1}$$

$$\mathrm{RMS}=\sqrt{\frac{1}{N}\sum_{i=1}^{N}x_i^2} \tag{2}$$

$$\mathrm{ZC}=\sum_{i=1}^{N-1}\mathrm{sgn}\left(-x_i x_{i+1}\right) \tag{3}$$

$$\mathrm{WL}=\sum_{i=1}^{N-1}\left|x_{i+1}-x_i\right| \tag{4}$$

where $x_i(i=1,2,\cdots,N)$ is a time series of sEMG signals, $\mathrm{sgn}\left(x\right)=\begin{cases}1 & x>0\\ 0 & \text{other}\end{cases}$ represents the unit step function. In addition, setting a threshold can effectively avoid the signal cross counting due to noise.

2.1.2. Frequency domain feature

Given that each feature value extracted here is based on overlapping sliding windows, N is the length of the sliding window. In this article, the window size was set to 30 ms, and the sliding increment was set to 25 ms. The sEMG signals were extracted using the same window settings. The average value of the obtained data was extracted as the feature of sEMG signals.

$$\mathrm{MPF}=\frac{\int_0^{+\infty}fP\left(f\right)df}{\int_0^{+\infty}P\left(f\right)df} \tag{5}$$

$$\int_0^{MF}P\left(f\right)df=\int_{MF}^{+\infty}P\left(f\right)df=\frac{1}{2}\int_0^{+\infty}P\left(f\right)df \tag{6}$$

where $P\left(f\right)$ is the power spectral density function of the sEMG signals, f is frequency, and MF is the required median frequency.

2.2. Classification recognition algorithms

The standard artificial neural network (ANN) consists of three layers: the input, hidden, and output layers. The mathematical representation of a neural network is the propagation of functions from the input layer to the output layer. The DNN is named as such because it contains multiple hidden layers. It is also called a multi-layer perceptron. DNN is the basis of deep learning networks. A DNN is composed of the input, hidden, and output layers. In general, the first layer is the input layer, and the last layer is the output layer, and the layers in the middle are all hidden layers. The layers are fully connected, that is, any neuron in the *w* layer must be connected to any neuron in the $w+1$ layer. Figure 1 shows the schematic of the DNN structure [41].

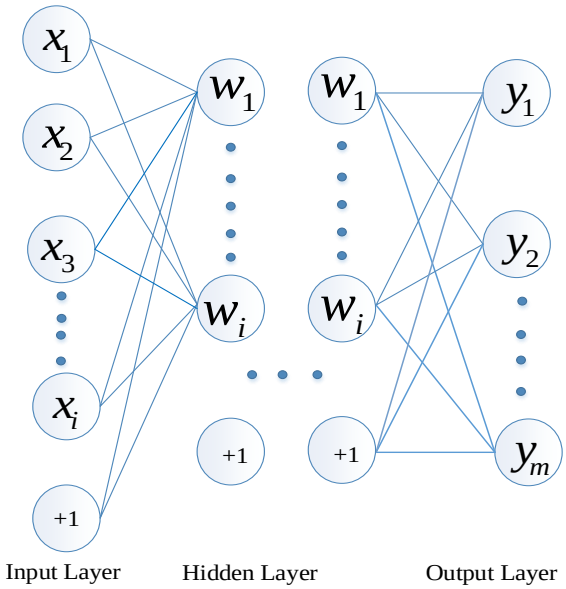


Figure 1. DNN model.

The basic structure of DNN is a perceptron with a two-layer structure. The perceptron learns the logical relationship between input x and output y and then obtains the final output through the set ReLU activation function:

$$y = \sigma(\sum_{i=1}^{n} w_i x_i + b) \tag{7}$$

where b is the activation threshold of the neuron, σ is the activation function, w_i is the connection weight between neurons, and x_i is the input value.

In the DNN training stage, the training algorithm adjusts the network parameters in reverse based on the difference between the actual output of the network and the expected output. Thus, the network parameters are optimized, and the output error is minimized. The BP algorithm is the commonly used network parameter training algorithm, and the parameter update method is as follows:

$$W' = W - \alpha \frac{\partial L}{\partial W} \tag{8}$$

$$b' = b - \alpha \frac{\partial L}{\partial b} \tag{9}$$

where W' is the neuron connection weight after parameter update, b' is the neuron bias after parameter update, W is the neuron weight, α is the learning rate of network training, b is the neuron bias, and L is the loss function.

2.3. Selection of DNN structural parameters

In the current research, DNN, which is an ANN with multiple hidden layers, was used to classify gait. Compared with traditional ANN (with a hidden layer), DNN can represent complex nonlinear

models. In this study, a neural network structure was used for comparative experiments. The ReLU activation function was used between layers, and the dropout layer was set to reduce overfitting. The number of training was set to 1000 times, and training sample size was set to 100. The learning rate was set to 0.01, and the input layer neuron was used to extract the feature values (TD and FD features). The number of hidden layer neurons was set to 128, and the output layer was five (five different gait stages). Normalized exponential function was used for the output layer to calculate the probability of each category. The BP algorithm was utilized to calculate cross entropy as a loss function, and the amount of data in the training and test sets accounted for 70 and 30% of the data setting, respectively.

The training process consisted of two steps: 1) the weights of hidden layers were trained using unlabeled data through a greedy layer-wise approach, and the normalized exponential function was trained using labeled data to map the inputs to assigned levels. 2) The network was fine-tuned using labeled data through BP.

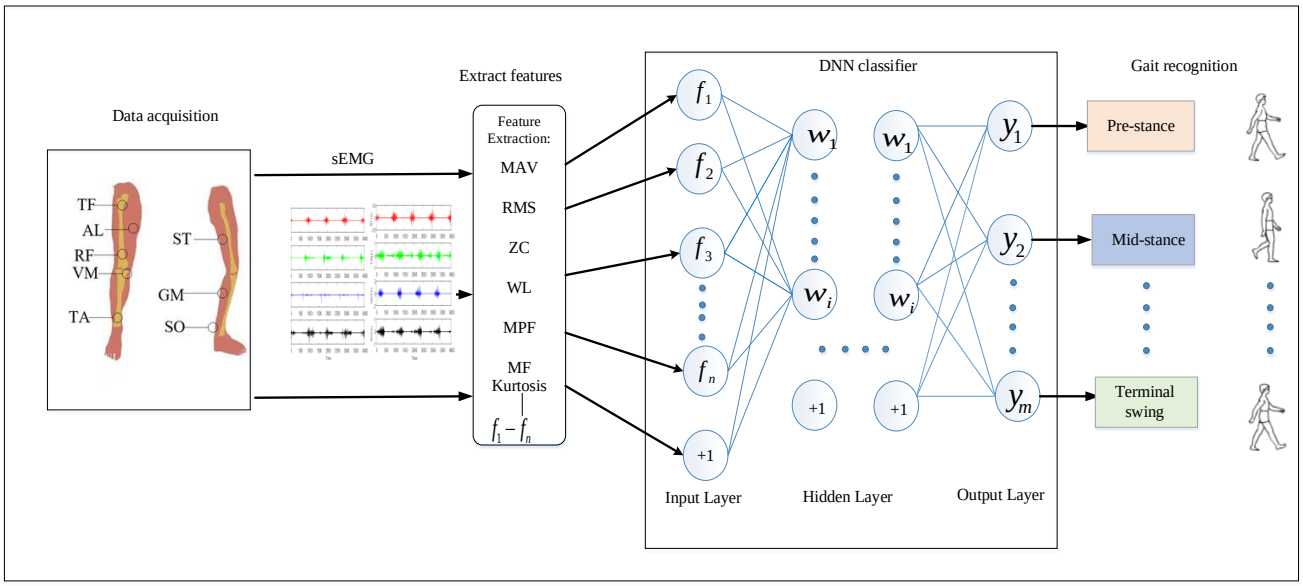


Figure 2. Flow chart of gait recognition.

The method of this paper mainly includes feature extraction, classification with the acquired gait data. The flow chart is shown in Figure 2.

3. Results and discussion

3.1. sEMG signal acquisition

Six healthy male subjects were selected to participate in the sEMG signals collection of this experiment. In the experiment, the subjects walked on a level ground at normal speed. The acquisition equipment used Delsys's Trigno wireless sensor, the sampling frequency of electromyographic signal was 1000 Hz, and the tester used an electromyographic acquisition instrument to collect the sEMG signals data on the above-mentioned eight muscle movements during periodic gait walking (Figure 3).

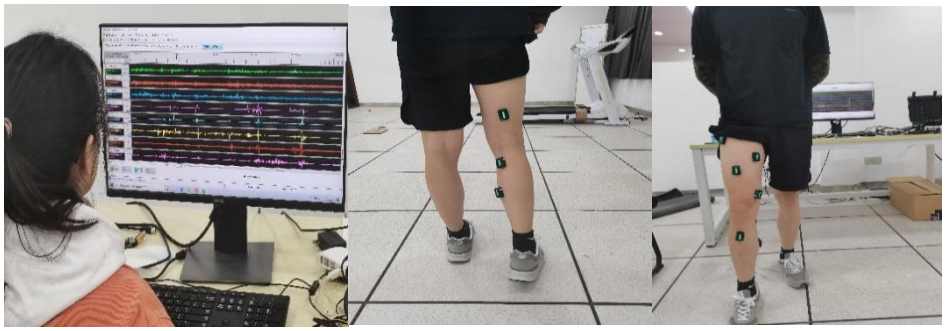


Figure 3. sEMG signal acquisition.

We selected eight muscles of the human body during the movement of lower limbs: vastus medialis (VM), adductor longus (AL), tensor fascia lata (TF), semitendinosus (ST), rectus femoris (RF), tibialis anterior (TA), gastrocnemius (GM), and soleus (SO) (Figure 4).

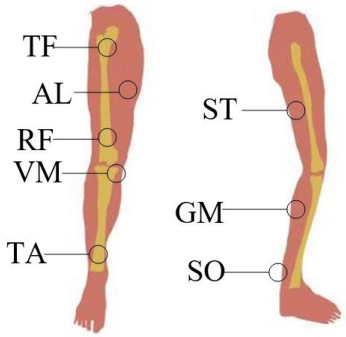


Figure 4. Muscle position diagram.

The gait cycle of a normal person is composed of support phase and swing phase that can be further divided into five stages: pre-stance, mid-stance, terminal-stance, pre-swing, and terminal-swing. The testers used the sEMG acquisition instrument to collect the sEMG signals data on the above-mentioned eight-way muscle movement during periodic gait walking.

At the beginning of the experiment, the experimenter walked on the level ground at normal speed (1.5 m/s). The members of each group continued to walk until 70 gaits at a constant speed. Figure 5 showed the collected sEMG signals. The sEMG signals of each channel (different muscles) and the activation degrees of the same muscle in the asynchronous state exhibited differences. Therefore, the sEMG signals can reflect information of the asynchronous state.

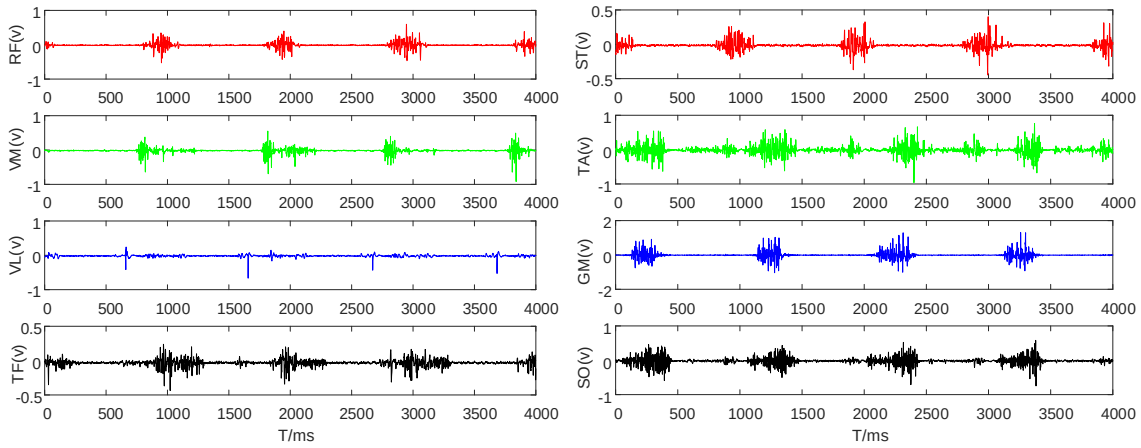


Figure 5. Collection of muscle sEMG signals.

3.2. sEMG signal data processing

Given that the collected signal will produce noise, whether artificial or natural, under a certain environmental background, to ensure the non-distortion of the signal, and to improve signal fidelity, for the extracted ST muscle signals, we examined three denoising algorithms: wavelet modulus maximum method, wavelet threshold method, and wavelet packet threshold method [42–44].

We analyzed three kinds of wavelet decomposition indicators: root-squared error (RMS) and signal-to-noise ratio (SNR)

Table 1. SNR and RMS of the three denoising methods.

Indicator analysis	Wavelet modulus maximum	wavelet packet threshold	wavelet threshold
RMS	0.0026	0.0027	0.0036
SNR	97.9229	93.4560	83.8056

From Table 1, the denoising performance of the RMS was small, and that of the SNR was large. Thus, excellent denoising of sEMG signals was attained. The values of SNR were 97.9229 (maximal value), 93.4560 and 83.8056, respectively, with wavelet modulus maximum, wavelet packet threshold, and wavelet threshold. In addition, for the RMS index, the wavelet modulus maximum method achieved the lowest value (0.0026). The results shows that the wavelet modulus maximum method is better than the other two methods in denoising sEMG signals. Therefore, we used the wavelet modulus maximum value for denoising [42].

3.3. Hidden layer number setting and corresponding evaluation index

Given that the multiple hidden layers in DNN can effectively compensate the performance difference caused by the varying numbers of neurons in each layer [45], The DNN model established in this study was only selected for the number of hidden layers. The number of selected hidden layers was within 1–4 [45], and the number of neurons in each hidden layer was 128.

Appropriately increasing the number of network layers can improve the performance of the model. However, in the case of insufficient data, the more complex the network structure, the more likely is overfitting to occur, resulting in a decreased generalization capability of the model. Therefore, we used the evaluation index of DNN for different numbers of network layers.

Table 2. Recognition rate and time of different hidden layers.

Hidden Layer	Accuracy (%)	Time (min)
1	92.39	1.23
2	95.52	1.96
3	96.79	2.34
4	95.77	3.87

The data in Table 2 showed that when the number of hidden layers was 1–3, the recognition rate of the network gradually increased. But, when four hidden layers were selected, the recognition rate decreased. The network training time was prolonged as the number of hidden layers increased. However, the increase was negligible. Thus, three hidden layers were selected in this study to ensure the accuracy of the network.

3.4. Influence of feature types on recognition results

1) Single-type feature recognition with SVM

First, the two types of features, namely, TD and FD, were inputted into the SVM for classification and recognition.

Table 3. Classification results of various features with SVM (%).

Sample	Pre-stance	Mid-stance	Terminal-stance	Pre-swing	Terminal-swing	Average accuracy
TD	94.69	93.16	95.01	95.82	95.78	94.89
FD	94.39	94.16	95.03	95.82	95.60	95.00

Table 3 showed the good recognition results for the two types of features when classified by SVM. Meanwhile, the FD feature recognition effect was the best, with a recognition rate of 95.00%, followed by that of the TD feature with a recognition rate of 94.89%. In addition, the recognition rate of the FD and TD features in the swing stance was significantly higher than those of other supporting stances. The swing recognition rate in the pre-swing was the highest at 95.82%.

2) Single-type feature recognition with ELM

The TD and FD features were inputted into the ELM for classification and recognition. Table 4 showed the recognition results.

Table 4. Classification results of various features with ELM (%).

Sample	Pre-stance	Mid-stance	Terminal-stance	Pre-swing	Terminal-swing	Average accuracy
TD	95.31	95.42	94.69	96.36	94.38	95.23
FD	94.06	95.67	94.87	95.69	95.71	95.20

Table 4 showed that the TD feature recognition rate in the ELM was the highest (95.23%). The FD feature samples exhibited a recognition rate of 95.20%. In addition, compared with the different periods, the TD features revealed a significantly higher recognition rate in the pre-swing, whereas the FD features showed a significantly higher recognition rate in the terminal-swing of gait period.

3) Single-type feature recognition with DNN.

The TD and FD features were inputted into the DNN for classification and recognition. Table 5 provided the recognition results.

Table 5. Classification results of various features with DNN (%).

Sample	Pre-stance	Mid-stance	Terminal-stance	Pre-swing	Terminal-swing	Average accuracy
TD	95.18	96.79	95.01	94.76	94.73	95.29
FD	94.91	96.50	94.62	94.66	94.54	95.08

Table 5 showed that the recognition results for the various types of features differed after being inputted to the DNN. The recognition rates with TD and FD features were 95.29 and 95.08%, respectively.

Tables 3–5 revealed that the classification results of each type of feature in each gait phase is different when the two types of features underwent gait pattern recognition through various classification models. This finding indicated that each type of feature varied during exercise. The sEMG signal information were analyzed from different perspectives.

3.5. Comparison of recognition results of different classifiers

The two types of features, (TD and FD), and combined TD + FD features were inputted into the DNN, ELM, and SVM models for classification and recognition. Tables 6 and 7 provided the recognition results.

Table 6. Classification results of TD features with various classifiers (%).

Classifier	Pre-stance	Mid-stance	Terminal-stance	Pre-swing	Terminal-swing	Average accuracy
DNN	95.18	96.79	95.01	94.76	94.73	95.29
ELM	95.31	95.42	94.69	96.36	94.38	95.23
SVM	94.69	93.16	95.01	95.82	95.78	94.89

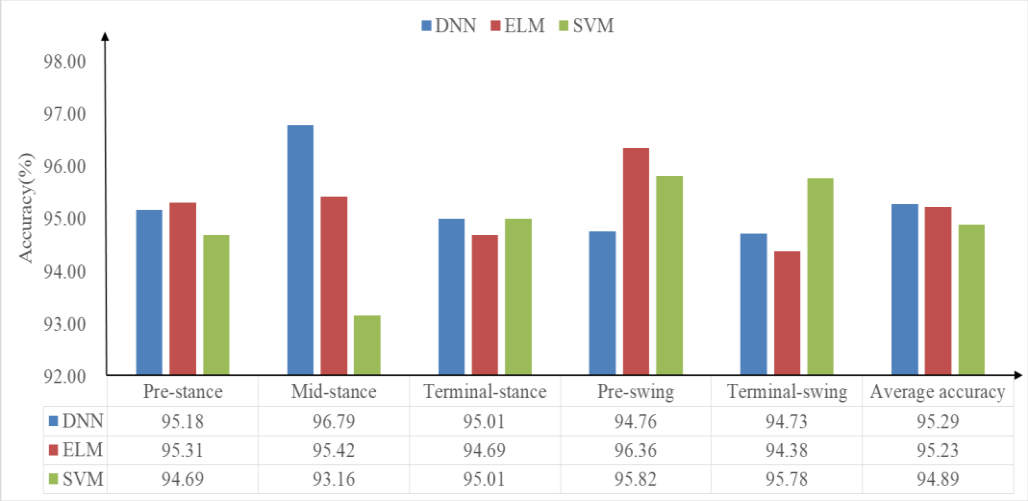


Figure 6. Classification results for the TD features.

The data analysis (Figure 6) showed that the DNN achieved the best classification and recognition results for the TD features (95.29%). Meanwhile, the results of ELM and SVM were comparable. In addition, the recognition rate of mid-stance under the DNN classifier (96.79%) was significantly higher than those obtained with the other two models. The classification results for the other four stages were similar.

Table 7. Classification results of FD features with various classifiers (%).

Classifier	Pre-stance	Mid-stance	Terminal-stance	Pre-swing	Terminal-swing	Average accuracy
DNN	94.91	96.50	94.62	94.66	94.54	95.08
ELM	94.06	95.67	94.87	95.69	95.71	95.20
SVM	94.39	94.16	95.03	95.82	95.60	95.00

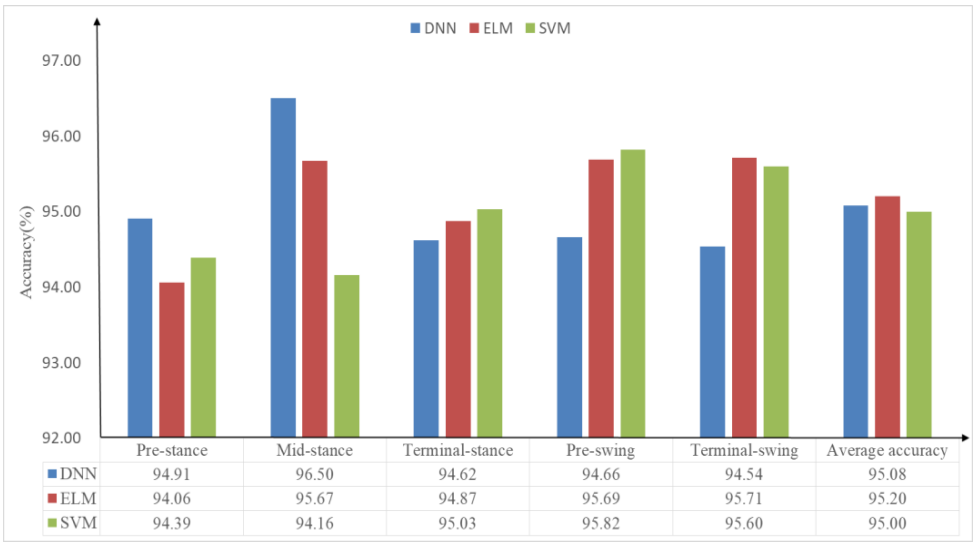


Figure 7. Classification results for FD features.

For the FD features, data analysis (Figure 7) showed that given the limited amount of extracted feature data, the DNN failed to achieve the best recognition result. The recognition rates of the five stages showed no considerable difference. The DNN yielded slightly poorer result than the other the two classifiers but displayed no considerable difference. Thus, the two feature types were merged to form the largest feature and attain different results.

In this study, the TD and FD features (TD + FD) were combined to form the largest features, which increased the data volume of the feature sample and improved the recognition effect after iteration. Table 8 showed the recognition effect of the combined features.

Table 8. Classification results of combined features with various classifiers (%).

Classifier	Pre-stance	Mid-stance	Terminal-stance	Pre-swing	Terminal-swing	Average accuracy
DNN	98.33	95.74	95.92	97.61	96.34	96.79
ELM	95.26	96.46	95.13	98.59	95.29	96.14
SVM	95.61	96.48	94.23	97.28	98.32	96.38

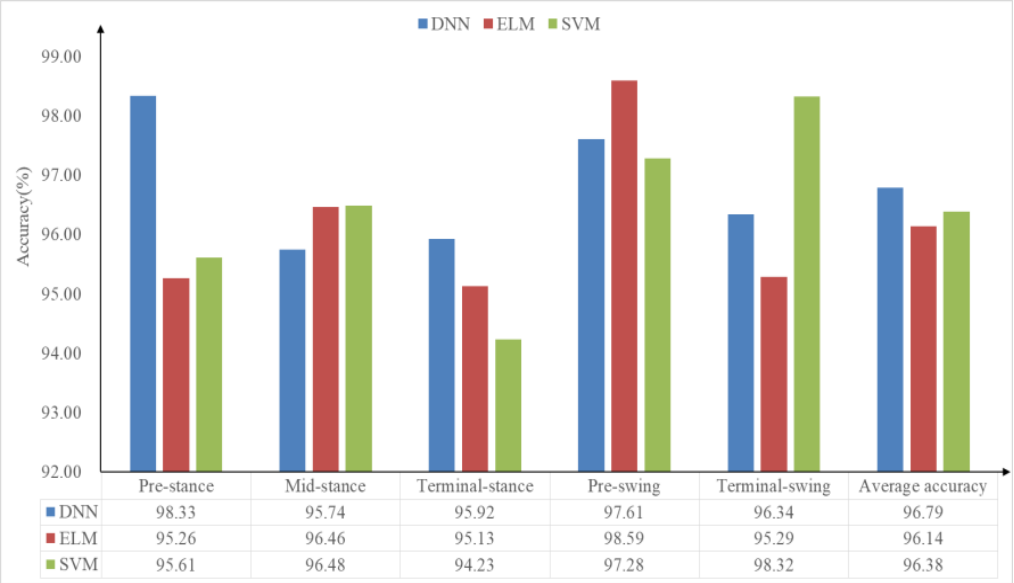
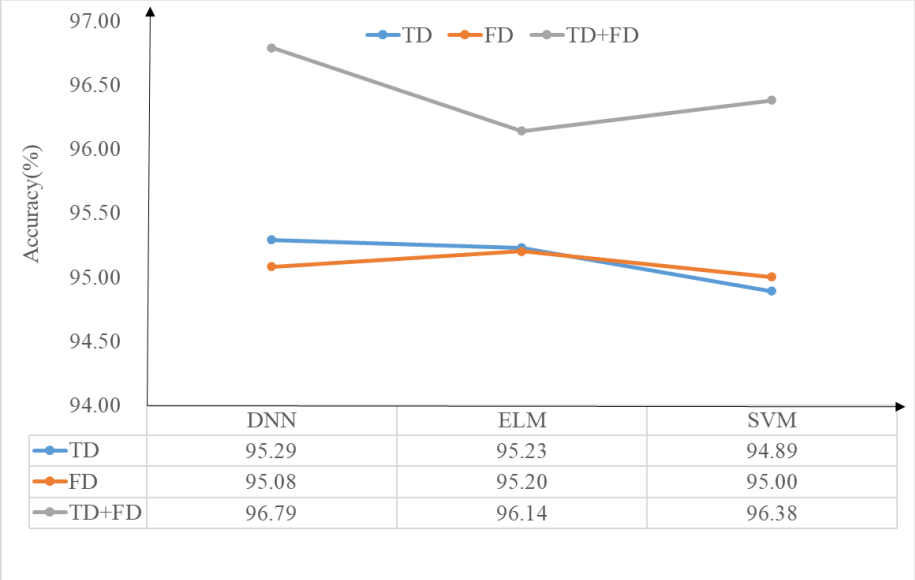


Figure 8. Classification results for combined features.

The data in Figure 8 revealed that after the combined features formed the largest feature, the recognition result of DNN was the highest (96.79%), whereas those of ELM and SVM were comparable. The recognition rate of pre-swing in ELM was the highest at 98.59%, whereas in SVM, the recognition rate at the terminal-swing was the highest at 98.32%. The recognition rate under pre-stance in the DNN classifier (98.33%) was higher than those observed with the other two classifiers.

From Figures 6–8, the recognition rates of the three types of classifiers for each feature sample varied. In general, the DNN exhibited better classification performance than ELM and SVM.



	DNN	ELM	SVM
TD	95.29	95.23	94.89
FD	95.08	95.20	95.00
TD+FD	96.79	96.14	96.38

Figure 9. Recognition results for three different features.

Figure 9 showed that the average recognition rate of the DNN method presented a different trend compared with the SVM and ELM methods. The three features (TD, FD, and TD + FD) were compared using the DNN classifier. The combination features increased the sample size, and the recognition accuracy was also increasing. The recognition results revealed that after combination, the best recognition rate of the features was observed.

3.6. Analysis of different indicators

In this section, gait recognition rates were discussed with the DNN classifier under the combined features, where the time domain features and frequency domain features were combined into the multi-feature combination (TD + FD).

3.6.1. Comparison of loss and recognition rates

We used the pre-stance, mid-stance, and pre-swing as examples (Figures 10–12, respectively).

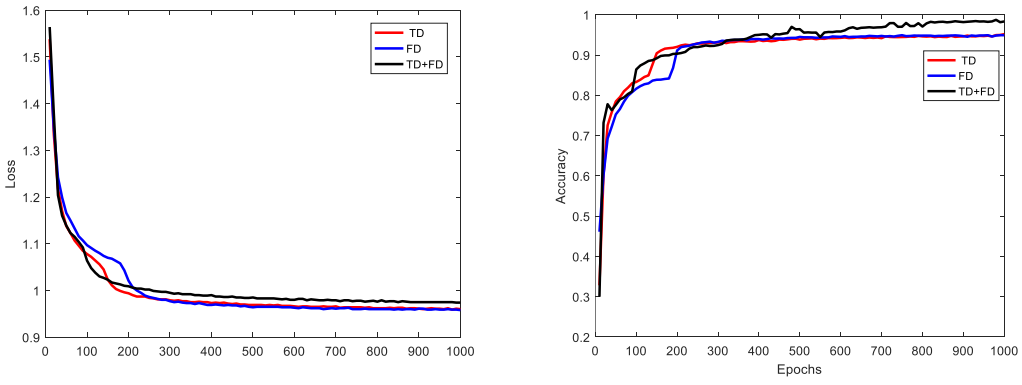


Figure 10. Loss and recognition rate of the DNN classifier in pre-stance.

As shown in Figure 10, for the combined features, the loss and recognition rate of pre-stance under the DNN classifier were the largest at the initial stage. The losses in the TD and FD were in a stable state, and the loss after fusion was the smallest. At the beginning, the recognition rate for the combined features was the lowest among the three feature types. However, as the number of training increased, the final recognition rate reached a stable state, and the recognition rate of the combined features was the highest at 98.33%.

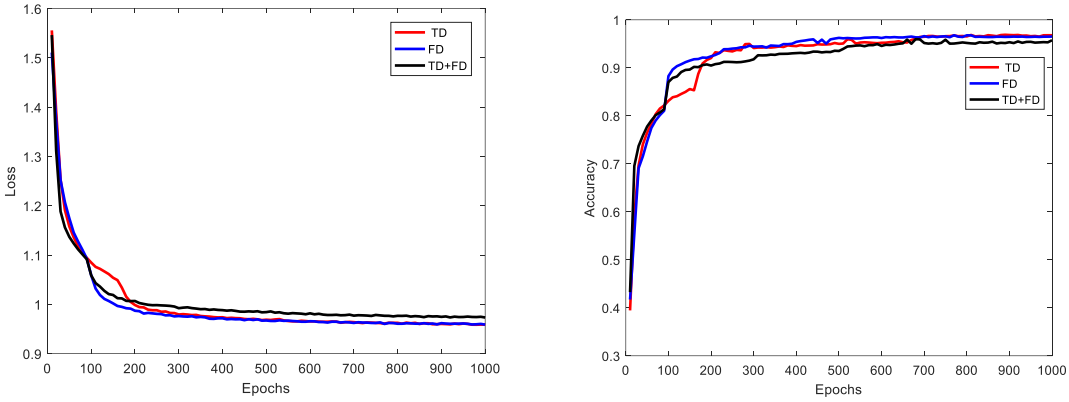


Figure 11. Loss and recognition rate of the DNN classifier in mid-stance.

Figure 11 revealed the loss and recognition rate of the mid-stance under the DNN classifier. The initial loss values of the three feature types showed no considerable difference. The loss of the three features reached a stable state as the number of training increased. The loss of the combined features was the smallest, and those of the other two single-type features were the same. At the beginning, the recognition rate for the TD features was the lowest among the three feature types. As the number of training increased, the final recognition rate stabilized, and the recognition rate for the features after fusion reached the highest at 95.74%.

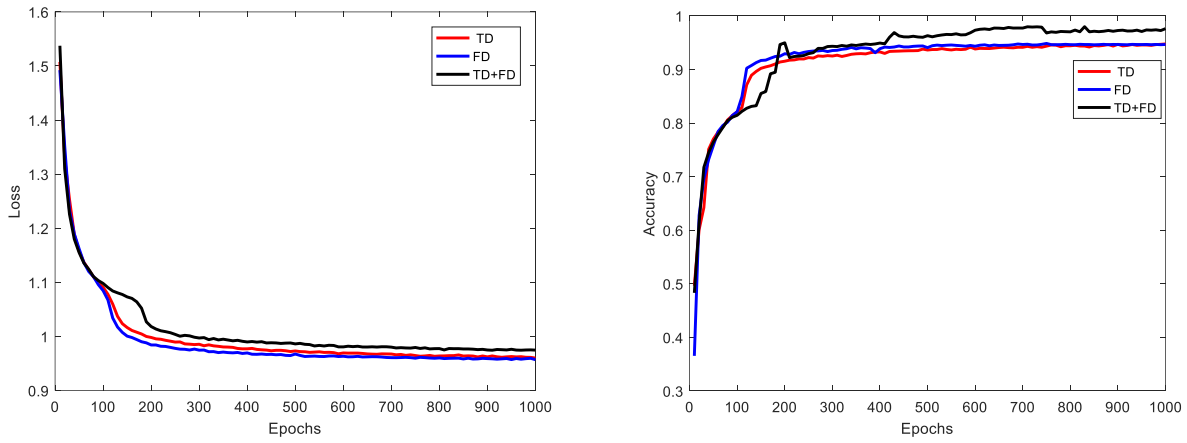


Figure 12. Loss and recognition rate of the DNN classifier in pre-swing.

Figure 12 presented the loss and recognition rate of pre-swing under the DNN classifier. The loss value of the combined features was the largest at the beginning, whereas those of TD and FD features were the same. With the increase in the number of training, the loss of the three feature types reached a stable state, with a minimal loss observed in the combined features. Meanwhile, the losses in TD and FD were the same. Initially, the recognition rate for FD features was the lowest among the three feature types. As the number of training increased, the final recognition rate arrived at a stable state, and the recognition rate of the combined features reached the highest at 97.61%.

Figures 10–12 showed that under the same number of iterations, the loss and recognition performance in different periods varied. The loss of pre-stance was the smallest, and the recognition effect was good. However, regardless of the period, the combined feature type achieved the smallest quantity loss and the best recognition rate. Thus, increasing the amount of data can alleviate the problem of model overfitting. The disadvantage of DNN is that if the number of training times is extremely small, then the model cannot converge; thus, considerable iterative training is required, and the total calculation time is notably longer than those of the other two shallow-network methods [46]. In addition, as the amount of training data increased, the recognition rate obtained by the DNN method also increased.

3.6.2. Losses of DNN models in various gait phases

In this experiment, we calculated the verification loss and recognition rate of different models of the DNN classifier, which is a fully connected feedforward architecture including three hidden layers. The number of units in the input layer was equal to the extracted sample features, and the size of the output layer was equal to the number of classes (five different gait phases). We used the ReLU activation function because the nonlinearity in the hidden layer generated by ReLU can desirably optimize the deep network, and the learning speed is fast. To prevent overfitting, we used the dropout layer in the terminal swing stage as an example (Figure 13).

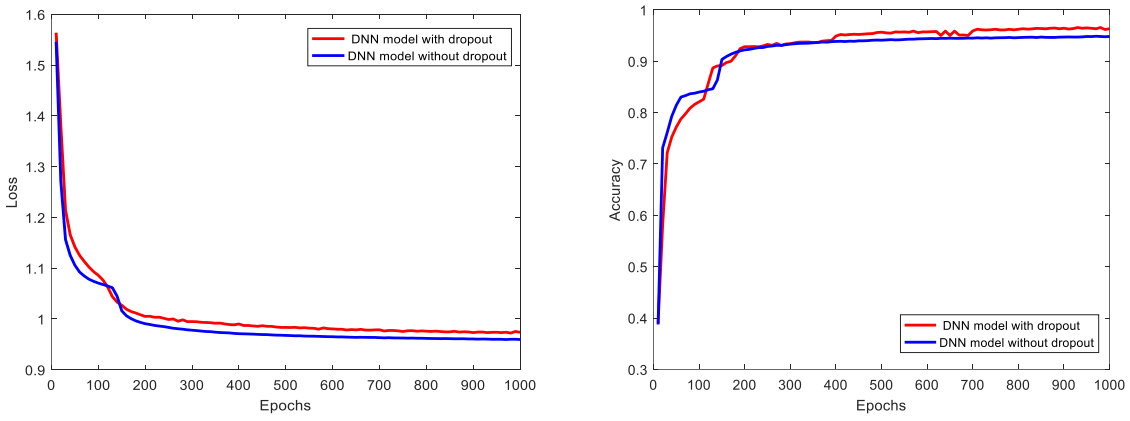


Figure 13. Different losses and recognition rates obtained with the DNN model.

Dropout is a technique used to prevent model overfitting. This technique is simple but practical. The basic idea is to randomly dropout certain activated neurons during training, which can improve the model robustness because it will become independent of specific local features

(given that local features may be discarded). Figure 13 showed that in the first 400 iterations, the recognition rates obtained with and without the dropout layers were the same and showed no difference. However, as the number of iterations increases, the model becomes prone to overfitting. As the training progresses, the learner's fitting ability is gradually enhanced, and the variation is gradually reduced. Therefore, adding a dropout layer when a certain number of iterations is reached can avoid overfitting and improve the recognition rate. Figure 13 revealed that when dropout was not used, overfitting occurred when the training data were small, and the recognition rate was 94.83%. After using dropout, the recognition rate reached 96.34%, which implies that dropout can alleviate the problem of overfitting. Meanwhile, the loss span was relatively large in the first 200 iterations, and the loss reached a stable state after 200–1000 iterations. The final loss with a dropout layer was less than that without a dropout layer.

3.6.3. Analysis of precision rate, recall rate, and reconciliation parameters

Table 9 compared the precision rate, recall rate, and harmonic average of the precision and recall rates of the five gait phases. Precision and recall rates can be influenced each other. Ideally, both parameters should be high. However, in general, when the accuracy rate is high, the recall rate is low. In this experiment, the precision rate was calculated as the recognition rate. The harmonic mean is equally important for correctly classifying each category, and we also considered the importance of accuracy and recall. Therefore, it can be used as a standardized measure of performance indicators.

Table 9. Experimental results of the combination feature (%).

Gait stage	Precision	Recall	F1 score
Pre-stance	0.9833	0.9860	0.9846
Mid-stance	0.9574	0.9224	0.9396
Terminal-stance	0.9592	0.9506	0.9549
Pre-swing	0.9761	0.9845	0.9803
Terminal-swing	0.9634	0.9569	0.9601
Average accuracy	0.9679	0.9601	0.9639

From Table 9, the accuracy and recall rates of the five gait phases showed no considerable difference, and the harmonic average of support in the pre-swing reached 0.9846, which is 0.0207 point higher than the overall average of the mid-stance. The lowest harmonic mean was 0.9396. The height of the harmonic mean depends on precision and recall. The higher the precision and recall, the larger the harmonic mean.

3.6.4. Analysis of standard deviation and variance

Table 10. Standard deviation of combination feature under the DNN classifier.

Gait stage	Standard deviation	Variance
Pre-stance	0.67	0.45
Mid-stance	0.94	0.88
Terminal-stance	0.92	0.84
Pre-swing	0.61	0.37
Terminal-swing	0.46	0.22
Average value	0.72	0.55

From the analysis of data results in Table 10, the standard deviation in the terminal-swing stage was the smallest (0.46), whereas that in the mid-stance was the largest, reaching 0.94. Thus, the smaller the standard deviation, the more stable is the swing. Thus, the recognition rate in the later stage of the swing was the most stable, and that of the mid-stance stage was the largest compared with the other four periods. A large variance will dominate the generalization capability of the model. If training is carried out further, the model may learn unique characteristics of the data set, and these characteristics are inapplicable to other data. At this point, an overfitting phenomenon has occurred. In this experiment, from the statistical results, the standard deviations ranged from 0.46 to 0.94%, which prove the robustness and stability of the proposed method.

3.6.5. Cross-validation in different state phases

Cross-validation is a statistical analysis method used to verify the performance of classifiers. *k*-fold cross-validation is the most widely used cross-validation method, and it can effectively avoid over-fitting and under-fitting. After analysis, *k*-fold cross-validation can be used to determine the validity and reliability of our results. This method can effectively avoid overfitting. Thus, the final result must be persuaded. In this experiment, we used *k*-fold cross-validation. The specific principle is to randomly divide all samples into *k* subsets of approximately equal size. *k*-1 of the *k* subsets is used for training, and the rest are used for training a test subset. Copies are continually obtained as new test samples, and the remaining *k*-1 copies are used as new training samples. The process is continued until all *k* samples are recognized as test samples; then, the average of all recognition results is obtained as the true generalized performance of the classifier [47]. *k* is generally greater than or equal to 2 and generally starts from 3 in actual operation; a *k* equal to 2 is attempted only when the amount of data in the original data set is small. Based on the characteristics of sEMG signals, we used 3-, 4-, 5-, and 10-fold cross-validation for calculation. For the 10-fold cross-validation, we randomly divided the data set into 10 parts, used 9 of them for training, and used the others as a test set. Then, we

randomly used different division methods for 10 calculations. Figure 14 showed the recognition rate of cross-validation.

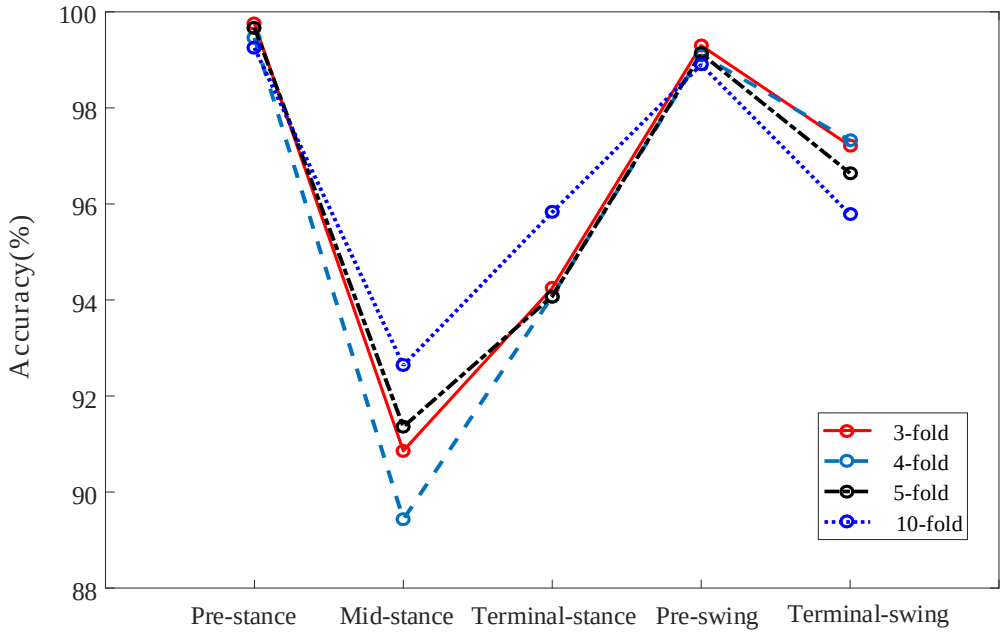


Figure 14. Cross validation in different gait phases.

As shown in Figure 14, for all *k*-folds, the accuracy verification of the five gait phases showed a difference. In these gait phases, the accuracy of pre-stance was the highest, and those for the mid-stance and terminal-stance were the lowest. In addition, during the terminal-swing, the three-, four-, and 5-fold cross validation exhibited a large transition from the pre-stance to mid-stance. For the 5-fold cross-validation, a large jump also occurred in the terminal-stance to the pre-swing. Judging from the recognition results, the recognition result of 4-fold cross-validation was the poorest and the most unstable, whereas that of the 10-fold cross-validation was the best compared with the other three. Therefore, overall, the 10-fold cross validation method is stable, reliable, and suitable for the identification method proposed in this study.

In summary, we arrived at the following conclusions:

1) Gait recognition through sEMG signals presented two problems. First, the walking gait is a continuous periodic process. For the classifier, the different stages of this continuous process need to be correctly divided. Second, sEMG, as a bioelectric signal generated by muscle contraction activity, contains a wealth of human motion and control information, from which the information required for gait recognition can be obtained. However, sEMG is non-linear. The stability prevents the conventional feature extraction method from fully decoding the sEMG signals. Therefore, classifier performance and feature sample selection are the main influencing factors of sEMG signals feature extraction.

2) In terms of classifier performance, compared with the SVM and ELM methods, the recognition rate of the DNN method showed a different trend. The recognition rate increased with the increase in the number of features. However, several shortcomings emerged. First, when a limited number features was used, the recognition rate of the DNN method was lower than that of the ELM method. Second,

its recognition calculation time was higher than those of the other two traditional algorithms. The main reason for this problem is the small amount of data, which is unsuitable for deep network training, used in this study. However, when the amount of training sample size increased, the accuracy with the method of DNN also improved, which has evident advantages in processing big data and is a direction worth studying in the future.

3) In terms of DNN index performance, different models affect the DNN recognition results. Therefore, a good model must be developed to achieve excellent recognition results. To prevent overfitting, we used the dropout layer. The recognition efficiency was greatly improved compared with the model without the dropout layer. The cross-validation method can also avoid overfitting. For the proposed method, the effect of 10-fold cross-validation is stable and reliable. From the calculation results of standard deviation, the DNN classifier has a certain degree of robustness and stability.

4. Conclusions

In this study, the problem of that used the combination of feature samples and the recognition algorithm of the DNN to improve the recognition accuracy and stability of recognition was studied. First, the recognition result with the combination of features showed better recognition performance than single-type features of TD and FD, presenting less loss and better recognition rate. However, when the number of layers was extremely high, the recognition result was poor. Appropriately increasing the number of network layers can improve the performance of the model. Overfitting will occur if the number of network layers is excessively high. On the other hand, the DNN algorithm in deep learning was used for comparison, and it showed a better result than the ELM and SVM. The results have certain reference value for human motion assessment, pedestrian gait detection, and intelligent security monitoring. However, the DNN also has its shortcomings. If the number of training times is extremely small, the model cannot converge and thus requires an intensive iterative training. Thus, the total calculation time is notably longer than those of the other two shallow-network methods. Finally, as the amount of training data increases, the accuracy obtained by the DNN method also increases, which has remarkable advantages in processing big data and is a direction worth studying in the future.

Acknowledgments

This work was supported by the Zhejiang Provincial Natural Science Foundation of China (No. LY20E050011), and the National Natural Science Foundation of China (No. 62071161).

Conflict of interest

The authors declare no conflicts of interest.

References

1. M. Perc, The dynamics of human gait, *Eur. J. Phy.*, **26** (2012), 525–534.
2. C. Mei, F. R. Gao, Y. Li, A determination method for gait event based on acceleration sensors, *Sensors*, **19** (2019), 5499.

3. J. Ryu, B. Lee, D. H. Kim, sEMG signal-based lower limb human motion detection using a top and slope feature extraction algorithm, *IEEE Signal Process. Lett.*, **24** (2017), 929–932.
4. P. Connor, A. Ross, Biometric recognition by gait: a survey of modalities and features, *Comput. Vis. Image Underst.*, **167** (2018), 1–27.
5. J. S. Richman, M. J. Randall, Physiological time-series analysis using approximate entropy and sample entropy, *Am. J. Physiol-Heart C.*, **278** (2000), 2039–2049.
6. D. Karabulut, F. Ortes, Y. Z. Arslan, M. A. Adli, Comparative evaluation of EMG signal features for myoelectric controlled human arm prosthetics, *Biocybern. Biomed. Eng.*, **37** (2017), 326–335.
7. A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, Y. Laurillau, EMG feature evaluation for improving myoelectric pattern recognition robustness, *Expert Syst. Appl.*, **40** (2013), 4832–4840.
8. J. J. Wang, F. R. Gao, Y. Sun, Z. Z. Luo, Non-uniform characteristics and its recognition effects for walking gait based on sEMG, *Chinese J. Sensor. Actuat.*, **29** (2016), 384–389.
9. P. Sbriccoli, I. Bazzucchi, A. Rosponi, M. Bernardi, G. D. Vito, F. Felici, Amplitude and spectral characteristics of biceps brachii sEMG depend upon speed of isometric force generation, *J. Electromyogr. Kinesiol.*, **13** (2003), 139–147.
10. S. Pancholi, A. M. Joshi, Portable EMG data acquisition module for upper limb prosthesis application, *IEEE Sens. J.*, **18** (2018), 3436–3443.
11. L. Zhang, G. Liu, B. Han, Z. Wang, T. Zhang, sEMG based human motion intention recognition, *J. Robot.*, **2019** (2019), 3679174.
12. C. Cortes, V. N. Vapnik, Support-vector networks, *Mach. Learn.*, **20** (1995), 273–297.
13. G. B. Huang, Q. Y. Zhu, C. K. Siew, Extreme learning machine: a new learning scheme of feedforward neural networks, *IEEE IJCNN*, (2004), 985–990.
14. I. S. Dhindsa, R. Agarwal, H. S. Ryait, Performance evaluation of various classifiers for predicting knee angle from electromyography signals, *Expert Syst.*, **36** (2019), 1–14.
15. Y. S. Wu, S. Liang, L. Zhang, Z. Q. Chai, C. C. Cao, S. W. Wang, Gesture recognition method based on a single-channel sEMG envelope signal, *EURASIP J. Wirel. Commun. Netw.*, **35** (2018), 1–8.
16. C. Castellini, P. Smagt, Surface EMG in advanced hand prosthetics, *Biol. Cybern.*, **100** (2009), 35–47.
17. Y. X. Kuang, Q. Wu, J. K. Shao, J. F. Wu, X. H. Wu, Extreme learning machine classification method for lower limb movement recognition, *Cluster Comput.*, **20** (2017), 3051–3059.
18. X. G. Xi, M. Y. Tang, S. M. Miran, Z. Z. Luo, Evaluation of feature extraction and recognition for activity monitoring and fall detection based on wearable sEMG sensors, *Sensors*, **17** (2017), 1229.
19. Y. X. Deng, F. R. Gao, H. H. Chen, Angle estimation for knee joint movement based on PCA-RELM algorithm, *Symmetry*, **12** (2020), 130.
20. F. R. Gao, T. X. Tian, T. Yao, Q. Z. Zhang, Human gait recognition based on multiple feature combination and parameter optimization algorithms, *Comput. Intell. Neurosci.*, **2021** (2021), 6693206.
21. H. T. Vu, D. B. Dong, H. L. Cao, T. Verstraten, D. Lefeber, B. Vanderborght, et al., A review of gait phase detection algorithms for lower limb prostheses, *Sensors*, **2019** (2020), 3972.
22. Y. L. Cun, Y. Bengio, G. Hinton, Deep learning, *Nature*, **521** (2015), 436–444.
23. C. Morbidoni, A. Cucchiarelli, S. Fioretti, F. D. Nardo, A deep learning approach to EMG-based classification of gait phases during level ground walking, *Electronics*, **8** (2019), 894.

24. J. C. Chen, X. D. Zhang, Y. Cheng, N. Xi, Surface EMG based continuous estimation of human lower limb joint angles by using deep belief networks, *Biomed. Signal Process. Control*, **40** (2018), 335–342.
25. M. Simao, N. Mendes, O. Gibaru, P. Neto, A review on electromyography decoding and pattern recognition for human-machine interaction, *IEEE Access*, **7** (2019), 39564–39582.
26. A. K. Mukhopadhyay, S. Samui, An experimental study on upper limb position invariant EMG signal classification based on deep neural network, *Biomed. Signal Process. Control*, **55** (2020), 101669.
27. J. H. Kim, G. S. Hong, B. G. Kim, D. P. Dogra, Deepgesture: deep learning-based gesture recognition scheme using motion sensors, *Displays*, **55** (2018), 38–45.
28. D. Jeong, B. G. Kim, S. Y. Dong, Deep joint spatiotemporal network (DJSTN) for efficient facial expression recognition, *Sensors*, **20** (2020), 1963.
29. J. H. Kim, B. G. Kima, P. P. Roy, D. M. Jeong, Efficient facial expression recognition algorithm based on hierarchical deep neural network structure, *IEEE Access*, **7** (2019), 41273–41285.
30. R. N. Khushaba, A. A. Ani, A. A. Timemy, A. A. Jumaily, A fusion of time-domain descriptors for improved myoelectric hand control, *IEEE SSCI*, (2016), 1–6.
31. A. Gharehbaghi, M. Linden, A deep machine learning method for classifying cyclic time series of biological signals using time-growing neural network, *IEEE Trans. Neur. Netw. Learn. Syst.*, **29** (2018), 4102–4115.
32. G. D. Guo, N. Zhang, A survey on deep learning based face recognition, *Comput. Vis. Image Underst.*, **189** (2019), 1–37.
33. C. J. Li, J. Ren, H. Q. Huang, B. Wang, Y. F. Zhu, H. S. Hu, PCA and deep learning based myoelectric grasping control of a prosthetic hand, *Biomed. Eng. Online*, **17** (2018), 107.
34. U. C. Allard, C. L. Fall, A. Drouin, A. C. Lecours, C. Gosselin, K. Glette, et al., Deep learning for electromyographic hand gesture signal classification using transfer learning, *IEEE Trans. Neur. Syst. Rehabil. Eng.*, **27**(2019), 760–771.
35. A. Phinyomark, P. Phukpattaranont, C. Limsakul, Feature reduction and selection for EMG signal classification, *Expert Syst. Appl.*, **39** (2012), 7420–7431.
36. M. Zardoshti-Kermani, B. C. Wheeler, K. Badie, R. M. Hashemi, EMG feature evaluation for movement control of upper extremity prostheses, *IEEE Trans. Rehabil. Eng.*, **3** (1995), 324–333.
37. R. Boostani, M. H. Moradi, Evaluation of the forearm EMG signal features for the control of a prosthetic hand, *Physiol. Meas.*, **24** (2003), 309–319.
38. Y. Y. Cao, F. R. Gao, L. Yu, Q. S. She, Gait recognition based on EMG information with multiple features, *IFIP Adv. Inf. Commun. Technol.*, (2018), 402–411.
39. J. H. Li, G. F. Li, Y. Sun, G. Z. Jiang, B. Tao, S. Xu, Hand motions recognition based on sEMG nonlinear feature and time domain feature fusion, *Int. J. Innovat. Comput. Appl.*, **10** (2019), 43–50.
40. Y. Narayan, Hb vsEMG signal classification with time domain and Frequency domain features using LDA and ANN classifier, *Mater. Today*, **37** (2021), 3226–3230.
41. G. E. Hinton, R. R. Salakhutdinov, Reducing the dimensionality of data with neural networks, *Science*, **313** (2006), 504–507.
42. R. L. Ortolan, R. N. Mori, R. R. Pereira, C. M. N. Cabral, J. C. Pereira, A. Cliquet, Evaluation of adaptive/nonadaptive filtering and wavelet transform techniques for noise reduction in EMG mobile acquisition equipment, *IEEE Trans. Neur. Syst. Rehabil. Eng.*, **11** (2003), 60–69.

43. S. G. Chang, B. Yu, M. Vetterli, Adaptive wavelet thresholding for image denoising and compression, *IEEE Trans. Image Process.*, **9** (2000), 1532–1546.
44. A. Fathi, A. R. Naghsh-Nilchi, Efficient image denoising method based on a new adaptive wavelet packet thresholding function, *IEEE Trans. Image Process.*, **21** (2012), 3981–3990.
45. G. E. Hinton, S. Osindero, Y. W. Teh, A fast learning algorithm for deep belief nets, *Neur. Comput.*, **18** (2006), 1527–1554.
46. C. Szegedy, W. Liu, Y. Q. Jia, P. Sermanet, S. Reed, D. Anguelov, et al., Going deeper with convolutions, *IEEE CVPR, Boston, MA, USA*, (2015), 1–9.
47. R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in *Proceedings of the 14th International Joint Conference on Artificial intelligence*, **2** (1995), 1137–1143.

AIMS Press