Small-Data-Driven Temporal Convolutional Capsule Network for Locomotion Mode Recognition of Robotic Prostheses

Yanggang Feng, Member, IEEE, Dinghao Xue, Linhang Ju, Graduate Student Member, IEEE, Wuxiang Zhang, Member, IEEE, and Xilun Ding, Member, IEEE

Abstract—Locomotion mode recognition has been shown to substantially contribute to the precise control of robotic lower-limb prostheses under different walking conditions. In this study, we proposed a temporal convolutional capsule network (TCCN) which integrates the spatial-temporal-based, dilation-convolution-based, dynamic routing and vector-based features for recognizing locomotion mode recognition with small data rather than big-data-based neural networks for robotic prostheses. TCCN proposed in this study has four characteristics, which extracts the (1) spatial-temporal information in the data and then makes (2) dilated convolution to deal with small data, and uses (3) dynamic routing, which produces some similarities to the human brain to process the data as a (4) vector, which is different from other scalar-based networks, such as convolutional neural network (CNN). By comparison with a traditional machine learning, e.g., support vector machine (SVM) and big-data-driven neural networks, e.g., CNN, recurrent neural network (RNN), temporal convolutional network (TCN) and capsule network (CN). The accuracy of TCCN is 4.1% higher than CNN under 5-fold cross-validation of three-locomotion-mode and 5.2% higher under the 5-fold cross-validation of five-locomotion modes. The main confusion we found appears in the transition state. The results indicate that TCCN can handle small data balancing global and local information which is closer to the way how the human brain works, and the capsule layer allows for better processing vector information and retains not only magnitude information, but also direction information.

Index Terms—Locomotion mode recognition, small data, robotic prosthesis.

I. INTRODUCTION

Robotic prostheses can assist people with lower-limb injuries in performing basic movements [1]. Human locomotion mode recognition plays a key role in robotic prostheses control under different terrains [2], [3]. To realize locomotion mode recognition, biological signals e.g. surface electromyography (sEMG), or position/force signals e.g. inertial measurement unit (IMU), multi-axis force/torque, from humans or robotic prostheses are utilized [4], [5], [6], and based on these signals, versatile algorithms including model-based computational methods, e.g. fuzzy logic [7], a slope gradient estimator [8], and model-free computational methods, traditional machine learning, e.g. support vector machines (SVMs) [9], [10], and big-data-driven neural network, e.g., convolutional neural network (CNN) [6], [11], are proposed to obtain more accurate recognition.

Model-based computational methods are proposed to address the issue on accurate locomotion mode recognition [7], [8]. With one inertial measurement unit (IMU) in the backpack and encoders to measure lower-limb joint angles, a slope gradient estimator based on the sensor data fusion is proposed to construct an adaptive gait planning approach for sloped terrains. The performance of the approach is limited by the sensor accuracy, tracking errors of the controllers and kinematics computation [8]. The fuzzy logic-based method could apply fuzzy sets and fuzzy rules for reasoning about transitional boundaries or qualitative knowledge experiences for describing systems which is suitable for terrain identification for robotic prostheses, and the average identification accuracy could be 98.74%, and the average identification delay could be 9.06% of one gait cycle [7]. If the human locomotion modes are infinite, which may be established accurately and completely, human locomotion may be more easily and rapidly recognized. However, the varied and individualized differences in human movement increase the challenge to recognize accurate human movement modes [7].

Model-free methods gradually attract wide attention with the development of machine learning and neural network. Strides could be divided into three separate feature sets including sensed, translational, and expanded, then cross-validation is performed using linear discriminant analysis to enhance walking task prediction in robotic prostheses [12]. Support vector machine (SVM) classifier is also used to recognize the locomotion mode [10]. Raw data is collected from onboard sensors to calculate feature values and concatenated together to be feature vectors with their special locomotion mode labels to train the classifier. After trained, SVM classifier is the input of the next real-time recognition [10]. To cope with the time-varying problems of surface electromyography (sEMG) signals, adaptive intent recognition algorithms are
proposed which were evaluated on transfemoral amputees across multiple days. The algorithm incorporated 96.31% of neural information across multiple experimental sessions, and outperformed non-adaptive versions of the algorithm with a 6.66% relative decrease in error rate [13] and experiment sessions [14]. To realize independent intent recognition for transfemoral amputees, a mode-specific classification method is proposed which allows each locomotion transition to be statistically considered its own class, and the method could reduce error on transitional steps by up to 50% without affecting steady-state classification [15]. To optimize the interaction between humans and robotics and improve sufficient synchronization of the robotic prosthesis with human movements, gait recognition and the prediction model is proposed called the gait neural network (GNN), which is based on the temporal convolutional network (TCN) [16]. Performance of the GNN was evaluated based on the publicly available HuGaDB dataset, and by evaluating the GNN on the collected data, GNN achieves substantial gait-prediction performance even without strong periodicity, and validated on data collected by an inertial-based wearable motion capture device and compare GNN with other computational methods, e.g., back propagation (BP), SVM, CNN, long short-term memory (LSTM) and Light Gradient Boosting Machine (LightGBM) [16].

Since using neural networks based on big data has been shown to substantially improve training accuracy, for instance, based on the ImageNet database (3.2 million images) [17], the facial recognition using the neural networks provides an insight of 100% accuracy under different conditions [18]. In the field of wearable robots, image-based [19], [20], [21], [22] and kinematic-based databases [23] on humans gait are established, but the amount of these databases is relatively smaller than ImageNet database (3.2 million images) [17]. Especially, for robotic prostheses, insufficient specific subjects, e.g., amputees with lower-limb loss, increased the difficulty of achieving big data on long-term walking gait, and therefore, relatively smaller data increases the requirement for algorithms.

The main contributions of the paper are as follows. Most neural networks were originally designed for big data recognition, and in the field of robotic prosthetics, the data is relatively small. In the study, we proposed a temporal convolutional capsule network (TCCN) which integrates the spatial-temporal-based, dilation-convolution-based, dynamic routing and vector-based features for recognizing locomotion mode with small data rather than big-data-based neural networks for robotic prostheses.

II. METHODS

Three subjects with transtibial amputation participated in the experiment as same in [11]. We collected the data of the strain gauge signal when they walked after wearing the robotic prosthesis. We proposed a temporal convolutional capsule network (TCCN) for locomotion mode recognition. To compare with TCCN, we investigated convolutional neural network(CNN), recurrent neural network(RNN), temporal convolutional network(TCN), capsule network(CN) and SVM algorithms. The overview of this study is shown in Fig. 1. In this section, the evaluation methods and some model conditions of training are also shown.

A. Prosthesis and Locomotion Mode

The experiments were approved by the Local Ethics Committee of Peking University. The raw data in this study are from one strain gauge inside a robotic prosthesis, as same in [11]. The prosthesis in this study comes from a previous study [11]. The detailed parameters of the prosthetic model can be referred to [11]. The prosthesis uses two kinds of sensors(including one full bridge of strain gauge and one angle sensor). One angle sensor was used to measure ankle angle. The strain gauge which reflects the deformation information due to contact force between carbon-fibre footplate and ground contains motion mode information while walking on different terrains. To reduce the weight and volume, the battery installation is designed to be embedded. In this study, five locomotion modes (i.e. level ground, ramp ascent, ramp descent, stair ascent and stair descent) were studied and analyzed. Each subject was asked to perform level ground, ramp ascent, level ground and stair descent (forward walking), turn around, and perform stair ascent, level ground, ramp descent and level ground (reverse walking) in order, as seen in Fig. 1. There were 35 repetitions of forward/reserve walking for each subject. All subjects volunteered to participate in the experiments and signed informed consent. The possible existing risks were also explained to them in advance. More details could refer to [11].

B. SVM and Neural Network Algorithms

The data used in this study was derived from the previous study on locomotion mode recognition [11]. Participants, data collection, and data pre-processing could refer to [11] in detail. The original signals from the strain gauge bridge were amplified 100 times and processed by a low-pass digital filter with a 10 Hz cutoff frequency [11]. The raw signal of each trial was divided into 10 strides at the heel strike time [11]. Linear interpolation is performed on the signal of each step, and the signal length of each step is transformed into a fixed length (1000) [11]. All data was max-min normalized [11]. In this study, several different neural networks and SVM algorithms were used to identify locomotion patterns. Different types of networks, accordingly, have different data processing methods. CNN processes data with spatial characteristics, and RNN processes data with temporal characteristics, TCN, TCCN, and CN process data with spatial-temporal characteristics. After pre-processing, the data format from the strain gauge is 1000 × 1 where 1000 represents the length of the signal of each stride. CNN and SVM can directly apply this data, while other neural networks dealing with data with temporal characteristics or data with spatial-temporal characteristics can not. Therefore, the data was further processed to convert the data of each stride (1000 × 1) into (20 × 50), as shown in Fig. 2. The purpose of dimension transformation is to divide a gait cycle into 20 time steps, where each time step has 50 data points, and the post-processed data retains the spatial-temporal characteristics.

1) TCCN: Inspired by TCN and the capsule network’s data processing as a vector, and by referring to the relevant study [24], we knew that dilated convolution may improve the performance of the deep neural network in small data sets. Combined with the characteristics of data, TCCN was
proposed to apply to the mode recognition of robotic prostheses, which extracts the spatial-temporal information in the data and could operate the data as a vector by which TCCN may extract more information from data, such as pose information. Applying to locomotion mode recognition, the features may improve the recognition accuracy. TCN is a neural network based on dilated convolution, causal convolution and residual connection. CN is a new network proposed by Hinton et al. For a more detailed introduction to the capsule network, please refer to [25]. In the proposed TCCN model, there are 8 layers. TCCN consists of three parts. The first part includes two residual blocks, the second part is the capsule layer, and the last part is the fully connected layer. A capsule layer is connected behind the residual blocks. Then, the output of the capsule layer is transmitted to the fully connected layer to obtain the prediction probability of each classification. The residual block contains three convolutional layers, as shown in Fig. 3. The first convolutional layer in each residual block uses rectified linear unit(ReLU) as the active function. Using the residual block, we can directly propagate the input from the lower layer to the higher layer and add the input, which may solve the problem of network degradation to a certain extent. In particular, the convolutional layer used in the residual block has dilated convolution. By sampling the input data exponentially layer by layer, the dilated convolution can obtain a larger receptive field with fewer network layers, which can reduce the depth of the network. Dilated convolution demonstrates the capability to improve the performance of small data sets in deep convolutional neural networks [24]. Therefore, we try to use dilated convolution in the proposed TCCN model. The TCCN model is shown in Fig. 4. The details of the structure are shown in Table I.
The dilation rate used for dilated convolution in residual blocks (1) and (2) are 1 and 4, respectively. The specific operation in the capsule layer could refer to [25].

As shown in Fig. 4, the proposed TCCN model. The abbreviations LG, RA, SA, RD, and SD stand for Level Ground, Ramp Ascent, Stair Ascent, Ramp Descent, and Stair Descent respectively. AS includes ramp ascent terrain and stair ascent terrain. DS includes ramp descent terrain and stair descent terrain.

C. Evaluation Method

5-fold cross-validation and corresponding confusing matrix were used to evaluate the efficacy of corresponding computational methods.

1) 5-Fold Cross-Validation: In 5-fold cross-validation, the initial sampling is divided into 5 subsamples, one single subsample is retained as the data of the validation model, and the other 4 samples are used for training. The cross-validation is repeated 5 times, and each subsample is verified once. The results of 5 times on average are used to finally obtain a single estimation. The advantage of this method is that the randomly generated subsamples are repeatedly used for training and verification.

2) Confusion Matrix: To show the specific details of misclassification between locomotion pattern recognition, the confusion matrix is computed. The confusion matrix of 5-locomotion modes is defined as follows:

\[
C = \begin{pmatrix}
C_{00} & C_{01} & C_{02} & C_{03} & C_{04} \\
C_{10} & C_{11} & C_{12} & C_{13} & C_{14} \\
C_{20} & C_{21} & C_{22} & C_{23} & C_{24} \\
C_{30} & C_{31} & C_{32} & C_{33} & C_{34} \\
C_{40} & C_{41} & C_{42} & C_{43} & C_{44}
\end{pmatrix}
\]

\(c_{ij}\) is defined as follows:

\[
c_{ij} = \frac{a_{ij}}{a_i} \times 100\%
\]

(1)

where ‘0’ represents the Level Ground, ‘1’ represents Ramp Ascent, ‘2’ represents Ramp Descent, ‘3’ represents Stairs Ascent, and ‘4’ represents Stairs Descent. \(a_{ij}\) represents the misclassification of mode \(i\) into mode \(j\), and \(a_i\) represents the total number of mode \(i\).

D. Model Training

In this study, we identified three motion patterns and five motion patterns. Because the control parameters of the prosthesis are similar in similar movement terrains, such as ramp ascent and stair ascent [11]. The misclassification occurring between similar states shows a relatively small impact on the control of robotic prostheses. Therefore, we classified similar terrains into the same category. Stair ascent terrain and ramp ascent terrain were classified as the ascent mode and stair descent terrain and ramp descent terrain were classified as the descent mode. Among the three locomotion modes, the ascent mode includes stair ascent and ramp ascent, and the descent mode includes stair descent and ramp descent. We randomly divided the data of each subject into three groups, one for training, one for verification, and one for testing. The size of the training set is 60%, and the size of the verification set and test set is 20%.

A loss function is used to calculate the difference between the predicted value of the neural network and the real value. Through the size of the loss value, the coefficients of the neural network are optimized through specific algorithms (e.g., backpropagation algorithm). In CNN, RNN, and TCN models, we used the cross-entropy loss function, while in CN and TCCN, we used the margin loss function. The margin loss function could refer to [25]. The cross-entropy loss function used in this study is defined as follows:

\[
Loss = \frac{1}{S} \sum_{i} \sum_{j=1}^{S} y_{ij} \log(\hat{y}_{ij}),
\]

where \(Loss\) represents the total loss, \(S\) is the total number of samples, \(i\) represents the sample number, \(j\) represents the locomotion mode, \(\hat{y}_{ij}\) represents the probability that the \(i^{th}\) sample is predicted to be class \(j\) \((j = 1, 2, \ldots, 5)\). If the

---

**TABLE I**

| Layers | Type     | Convolution kernel size | Dilation rate | Output Shape |
|--------|----------|-------------------------|--------------|--------------|
| 0      | Input Layer | -                       | -            | (Note,20,50) |
| 1      | Convolution Layer | 3                       | 1            | (Note,20,32) |
| 2      | Convolution Layer | 3                       | 1            | (Note,20,32) |
| 3      | Convolution Layer | 1                       | 1            | (Note,20,32) |
| 4      | Convolution Layer | 3                       | 4            | (Note,20,128) |
| 5      | Convolution Layer | 3                       | 4            | (Note,20,128) |
| 6      | Convolution Layer | 1                       | 1            | (Note,20,128) |
| 7      | Capsule Layer | -                       | -            | (Note,5,10)  |
| 8      | Output Layer  | -                       | -            | (Note,5)     |
ith sample belongs to class \( k (k = 1, 2, \ldots, 5) \), then,

\[
y_{ij} = \begin{cases} 
1, & j = k \\
0, & j \neq k 
\end{cases}
\] 

(3)

In the convolutional neural network (CNN), recurrent neural network (RNN), temporal convolutional network (TCN), capsule network (CN), and temporal convolutional capsule network (TCCN), adaptive momentum estimation (Adam) optimizer was used to train our network model. In the CNN model, the keeping proportion for the first fully connected layer was 0.8. The dropout rate used for the fully connected layer was 0.2 in the RNN model. In our TCN model, this dropout rate used for the flatten layer was also 0.2. In the capsule network model and the TCCN model, the rate of the dropout layer used ahead of the capsule layer was 0.2. The number of dynamic routes used in the capsule layer was 3. The learning rate used by the adam optimizer in the CNN model was 0.001 and the learning rate used by the adam optimizer in RNN, TCN, CN, and TCCN models was 0.001. In the training process, the maximum number of training rounds we allow was 500. In the study, the convergence conditions of used networks were consistent. When the loss value on the validation set had not been improved in 20 epochs or the training reached the maximum number of training epochs, our training process stopped and the model with the least loss on the validation set was selected.

### III. Results

Two partition methods were applied to the data set. One is to train the data of three subjects who participated in the experiment separately and calculate the average value of the results, called “separate”. The other is to mix the data of three subjects together as the total and then train them, called “total”.

#### A. Recognition Accuracy

From Fig. 5 and Fig. 6, we can see that the mode recognition accuracy of TCCN is higher than that of other neural networks and SVM algorithms, whether in three locomotion mode recognition or five locomotion mode recognition. The results mean that whether in three locomotion mode recognition or five locomotion mode recognition, TCCN has better performance in locomotion mode recognition than other neural networks and SVM algorithms.

1) 5-Fold Cross-Validation (Separate): Fig. 5 shows the results with different algorithms in 5-fold cross-validation. In the 5-fold cross-validation (separate), the data of three subjects were trained separately and the average accuracy was taken as the final result. The accuracy of CNN is the lowest and the accuracy of TCCN is the highest among these neural networks and SVM algorithms. The results about accuracy indicated that the TCCN did yield a better result than other networks and SVM algorithms. The accuracy of the TCCN in 3 classification in 5-fold cross-validation for three subjects is 96.6\(\pm\)1.2\%. The average accuracy in 5 classification is 94.1\(\pm\)1.7\%. The accuracy of the CNN in 3 classification in 5-fold cross-validation for three subjects is 92.5\(\pm\)1.6\%. The accuracy in 5 classification is 88.9\(\pm\)4.2\%. Compared with CNN, the accuracy of TCCN in 5 classification increases by 5.2\%, and the accuracy of TCCN in 3 classification increases by 4.1\%.

#### 2) 5-Fold Cross-Validation (Total): To better illustrate the performance of TCCN, we also mixed the data of three subjects to evaluate these algorithms. Fig. 6 shows the results with different neural networks and SVM algorithm in 5-fold cross validation (total). The accuracy of TCCN is the highest among these neural networks and SVM algorithms, whether in three locomotion mode recognition or five locomotion mode recognition. The average accuracy of the TCCN in 3 classification using 5-fold cross-validation (total) is 95.8\%. The average accuracy in 5 classification is 93.6\%. The average accuracy of the CNN in 3 classification using 5-fold cross-validation (total) is 95.8\%. The accuracy in 5 classification is 91.0\%. In the five classification using 5-fold cross-validation (total), the average accuracy of the TCCN is 2.6\% higher than that of the CNN. In the three classification, the average accuracy of the TCCN is 1.4\% higher than the accuracy of the CNN.

#### B. Confusion Matrix in 5-Fold Cross-Validation

To better understand the recognition results, we also provided the confusion matrix of each neural network and SVM algorithms. Fig. 7 is the confusion matrix of neural network and SVM algorithms in 5-fold cross-validation (separate) for 3 locomotion mode recognition. The confusion matrix of neural network and SVM algorithms in 5-fold cross-validation (separate) for 5 locomotion mode recognition is shown in Fig. 8. Fig. 9 and Fig. 10 show the confusion matrix of neural networks and SVM algorithms in 5-fold cross-validation (total) for three locomotion and five locomotion modes, respectively.
In the TCCN model, the results are similar to the CNN model. From Fig. 8, the results are similar to those obtained from Fig. 7. The misclassification mainly occurred in the transition state. The least misclassification occurred between ascent mode and descent mode. In CNN and TCCN model, the misclassification rate between ascendant and descendant terrain is close to zero. In RNN, TCN, CN and SVM models, the misclassification rate between ascendant and descendant terrain is zero.

2) 5-Fold Cross-Validation (Total): In the 5-fold cross-validation (total), the data of three subjects were trained by mixing them for locomotion mode recognition.

a) 5-fold cross-validation (total) for three-locomotion modes: The confusion matrix of neural networks and SVM algorithms in the 5-fold cross-validation (total) is shown in Fig. 9. The main misclassification occurred between level ground and ascendant environment, and between level ground and descendant environment.

b) 5-fold cross-validation (total) for five-locomotion modes: The confusion matrix of neural networks and SVM algorithms in the 5-fold cross-validation (total) is shown in Fig. 10. The main misclassification occurred between similar motion trends, e.g. stair ascent and ramp ascent. The misclassification between ascendant terrain and descendant terrain is relatively small, compared with the misclassification between level ground and other locomotion modes.

IV. DISCUSSION

The main contribution of this study is that we proposed a temporal convolutional capsule network (TCCN) and compared the performance of TCCN and other algorithms using the same small dataset. We used several different methods to compare and evaluate the performance of the algorithms. We divided the data into three dimensions: original data dimension, temporal dimension, and spatial-temporal dimension to be fed into neural networks with different characteristics. The results suggested that the performance of TCCN was better than CNN, RNN, TCN, CN and SVM.

A. Dilated Convolution

Limited by the number of subjects, in the field of the robotic prosthesis, getting big data based on locomotion patterns is not realistic, and how to extract sufficient information from small data and recognise multiple locomotion modes is a challenge.

The prominent batch normalization and dilated convolution demonstrate the capability to improve the performance of small data sets in deep convolutional neural networks [24]. Taking these features to the proposed TCCN model, dilated convolution was used and the data was max-min normalized in data pre-processing. Several neural networks used in this study used the same data, and therefore, normalization maybe not be a factor to improve the performance of locomotion mode recognition.

In the CNN model and TCCN model, the significant differences between them are as follows: the CNN model only uses regular convolution, while the TCCN model uses dilated convolution; the CNN model does not use the capsule layer, and the TCCN model uses the capsule layer. Our experimental results show that the recognition accuracy of the TCCN model is higher than that of the CNN model. Dilated convolution demonstrates the capability to improve the performance of
Fig. 8. Matrix confusion using 5-fold cross-validation (separate) in five locomotion modes. The data of three subjects were trained separately for locomotion mode recognition. Then, the confusion matrix obtained by three subjects was averaged as the final result. The abbreviations LG, RA, SA, RD, and SD stand for Level Ground, Ramp Ascent, Stair Ascent, Ramp Descent, and Stair Descent respectively.

|        | (A) SVM | (B) CNN | (C) RNN |
|--------|---------|---------|---------|
| LG     | 97.0%  | 90.8%  | 94.8%  |
| RA     | 8.1%   | 8.5%   | 6.7%   |
| RD     | 2.0%   | 5.9%   | 3.4%   |
| SA     | 6.9%   | 10.6%  | 4.5%   |
| SD     | 0.0%   | 0.7%   | 0.0%   |

Fig. 9. Matrix confusion using 5-fold cross-validation (total) in three locomotion modes. The data of three subjects were trained by mixing them together for locomotion mode recognition. Then, the confusion matrix was obtained. The abbreviations LG, AS, and DS stand for Level Ground, Ascent, and Descent respectively. AS includes ramp ascent terrain and stair ascent terrain. DS includes ramp descent terrain and stair descent terrain.

|        | (D) TCN | (E) CN | (F) TCCN |
|--------|---------|--------|----------|
| LG     | 95.1%  | 95.2%  | 96.5%  |
| RA     | 6.3%   | 5.5%   | 3.9%   |
| RD     | 4.2%   | 3.4%   | 3.8%   |
| SA     | 4.0%   | 2.4%   | 2.1%   |
| SD     | 1.0%   | 0.3%   | 0.7%   |

Both TCN and TCCN were used in this study. The difference between the TCCN model and the TCN model is that the capsule layer was used in the TCCN model. The recognition results show that the performance of TCCN is relatively better than TCN, which could indicate that the capsule layer may improve the recognition performance on our small data set to a certain extent.

B. Dynamic Routing

The dynamic routing algorithm in the capsule layer is used to update the weight coefficient. The specific introduction of the algorithm can be found in [25]. The human brain tends to classify objects according to the global features, while CNN, RNN, and TCN classify objects according to the local features [26]. The proposed temporal convolutional capsule network (TCCN) in this study and the human brain have certain similarities in object recognition. The eye receives the object’s information and transmits it to the brain. The brain analyzes the hierarchical relationship in the information and tries to match the relationship already stored in the brain. When recognizing objects, the hierarchical pose relationship between object components is important. In the TCCN model, by using the capsule layer, the pose information in the data can be extracted, and then different objects can be recognized. From the point of view, TCCN and the human brain have certain similarities in object recognition. Each neuron in the human brain has a special function. When recognizing a feature, the feature must be transmitted to the neuron that is best at processing the feature. A similar idea of the capsule layer is reflected in dynamic routing. If the similarity between low-level features and high-level features is high, the coefficient between them is large, on the contrary, the coefficient is relatively small. By dynamic routing, the lower capsule is connected with a certain higher capsule. Through
feature extraction of layers, features extracted from high-level capsules are often more abstract, such as the relative positions of features. Since the capsule layer processes data as a vector form, the extracted features may include global information such as location and direction. Other neural networks that process data in the scalar form may extract only abstract features, excluding the relative position information between features. Consequently, this may result in the performance of TCCN being better than other networks and SVM algorithms.

C. Vector-Based Spatial-Temporal Feature

The TCCN model has a capsule layer, and the TCN model has no capsule layer. One feature of the capsule layer is to treat information as vectors [25]. The recognition accuracy of TCCN with the capsule layer is improved by about 1% compared with TCN without the capsule layer, which suggests that treating data as a vector could improve the performance in recognizing motion patterns. The capsule layer processes the data as vectors and retains the hierarchical relationship between object components, which is important for correctly classifying objects. On the contrary, other networks, e.g., CNN, RNN, TCN, and SVM in this study process the data as a scalar and ignore the “direction” information between features, which may not conducive to identifying different locomotion modes. In addition, TCCN regards data as spatial-temporal information rather than not only spatial or temporal information, which may be the cause of the higher recognition accuracy of TCCN compared with CNN and RNN. Fang et al. compared the recognition results of different methods on the HuGaDB data [16], and this study suggests that GNN yields the highest accuracy rate of 98.04% compared with CNN 79.24% and LSTM 92.78% [16]. GNN extracts spatial-temporal information from data, which also indicates that spatial-temporal information from small data is conducive to locomotion pattern recognition.

D. Limitation

The strain gauge sensor and other sensors are fused together for motion pattern recognition, which may reduce the number of sensors and classification errors in both transition and static states [27], [28]. In this study, we aim to investigate what features of algorithms may be conducive to locomotion mode recognition and only one sensor was utilized. Future research could focus on the experimental data obtained by fusing the strain gauge sensor and other sensors, and then apply the TCCN model to the fused data. Due to the limitation of experimental data, we only verified the performance of the TCCN proposed in this study based on our small data set and discovered that TCCN could improve the accuracy of motion pattern recognition. More small datasets from more experiments and sensors are needed to enhance the persuasion that TCCN can improve the performance of motion recognition on small datasets. Future research is needed to visualize what features the different dimensions of vectors represent and what role these features play in locomotion mode recognition. The input signal, namely the strain gauge signal, may be reconstructed from the capsule layer values in the TCCN model. Through the output value of the capsule layer, the output value of the previous layer is calculated backward layer by layer, and finally, the value of the input signal is obtained.
Then, the image of the input signal can be reconstructed. Control the change of the value of a dimension of the vector in the capsule layer, and obtain the specific meaning of the dimension according to the change of the reconstructed signal. When we know what each dimension of the vector represents about the input signal, we can better tune the hyperparameters.

V. CONCLUSION

In this study, we proposed a temporal convolutional capsule network (TCCN) based on dilated convolution derived from the capsule network. To show the performance of TCCN in locomotion mode recognition, we investigated other neural networks and SVM algorithms. We investigated the recognition accuracy of each network and SVM under 5-fold cross-validation and the confusion matrix under 5-fold cross-validation is also shown. Under the 5-fold cross-validation of three-locomotion modes, the accuracy of the TCCN model is 4.1% higher than that of the CNN model. Under the 5-fold cross-validation of five-locomotion modes, the accuracy of the TCCN model is 5.2% higher than that of the CNN model. This indicates that using TCCN could improve the performance of motion pattern recognition. Herein the different motion modes, the confusion between ascent mode and descent mode occurs the least and the misclassification rate is almost 0%. In the 5-fold cross-validation of the three-locomotion-mode, the misclassification between the ascent mode and the descent mode in the TCCN model is 0% and the misclassification between the ascent mode and the descent mode in the CNN model is around 0.5%. The misclassification mainly occurring between transition states shows a relatively small impact on the control of robotic prostheses, which indicates the efficacy and feasibility of TCCN under real terrains. The feature of TCCN-the dilated convolution and capsule layer(dynamic routing and vector-based), may be conducive to improving the locomotion mode recognition accuracy. TCCN proposed in this study extracts the spatial-temporal information in the data and processes the data as a vector, which is different from other scalar-based networks, e.g., CNN, which may improve the accuracy of motion pattern recognition to a certain extent.

In the future study, more focus should be on what information is represented by each dimension of the vector obtained when processing the data as a vector, and what information is key to locomotion mode recognition.

REFERENCES

[1] P. Cherelle, V. Grosu, A. Matthys, B. Vanderborght, and D. Lefeber, “Design and validation of the ankle mimicking prosthetic (AMP-) foot 2.0,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 22, no. 1, pp. 138–148, Jan. 2014.
[2] M. R. Tucker et al., “Control strategies for active lower extremity prosthetics and orthotics: A review,” J. Neuroeng. Rehabil., vol. 12, no. 1, p. 1, 2015.
[3] N. E. Krausz, T. Lenzi, and L. J. Hargrove, “Depth sensing for improved control of lower limb prostheses,” IEEE Trans. Biomed. Eng., vol. 62, no. 11, pp. 2576–2587, Nov. 2015.
[4] H. A. Varol, F. Sup, and M. Goldfarb, “Multiclass real-time intent recognition of a powered lower limb prosthesis,” IEEE Trans. Biomed. Eng., vol. 57, no. 3, pp. 542–551, Mar. 2010.
[5] H. Huang, F. Zhang, L. J. Hargrove, Z. Dou, D. R. Rogers, and K. B. Englehart, “Continuous locomotion-mode identification for prosthetic legs based on neuromuscular-mechanical fusion,” IEEE Trans. Biomed. Eng., vol. 58, no. 10, pp. 2867–2875, Oct. 2011.
[6] D. Xu, Y. Feng, J. Mai, and Q. Wang, “Real-time on-board recognition of continuous locomotion modes for amputees with robotic transibial prostheses,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 26, no. 10, pp. 2015–2025, Oct. 2018.
[7] K. Yuan, Q. Wang, and L. Wang, “Fuzzy-logic-based terrain identification with multisensor fusion for transibial amputees,” IEEE/ASME Trans. Mechatronics, vol. 20, no. 2, pp. 618–630, Apr. 2015.
[8] C. Zou, R. Huang, J. Qiu, Q. Chen, and H. Cheng, “Slope gradient adaptive gait planning for walking assistance lower limb exoskeletons,” IEEE Trans. Autom. Sci. Eng., vol. 18, no. 2, pp. 405–413, Apr. 2020.
[9] V. Papapicco et al., “A classification approach based on directed acyclic graph to predict locomotion activities with one inertial sensor on the thigh,” IEEE Trans. Med. Robot. Biomech., vol. 3, no. 2, pp. 436–445, May 2021.
[10] X. Liu and Q. Wang, “Real-time locomotion mode recognition and assistive torque control for unilateral knee exoskeleton on different terrains,” IEEE/ASME Trans. Mechatronics, vol. 25, no. 6, pp. 2722–2732, Dec. 2020.
[11] Y. Feng, W. Chen, and Q. Wang, “A strain gauge based locomotion mode recognition method using convolutional neural network,” Adv. Robot., vol. 33, no. 5, pp. 254–263, Mar. 2019.
[12] R. Stolyarov, G. Burnett, and H. Herr, “Translational motion tracking of leg joints for enhanced prediction of walking tasks,” IEEE Trans. Biomed. Eng., vol. 65, no. 4, pp. 763–769, Apr. 2018.
[13] J. A. Spanias, A. M. Simon, S. B. Finucane, E. J. Perreault, and L. J. Hargrove, “Online adaptive neural control of a robotic lower limb prosthesis,” J. Neural Eng., vol. 15, no. 1, Feb. 2018, Art no. 016015.
[14] M. Liu, F. Zhang, and H. H. Huang, “An adaptive classification strategy for reliable locomotion mode recognition,” Sensors, vol. 17, no. 9, p. 2020, 2017.
[15] A. J. Young and L. J. Hargrove, “A classification method for user-independent intent recognition for transfemoral amputees using powered lower limb prostheses,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 24, no. 2, pp. 217–225, Feb. 2016.
[16] B. Fang et al., “Gait neural network for human-exoskeleton interaction,” Front. Neurorobot., vol. 14, p. 58, Oct. 2020.
[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 248–255.
[18] M. M. Ahsan, Y. Li, J. Zhang, M. T. Ahad, and K. D. Gupta, “Evaluating the performance of Eigenface, Fisherface, and local binary pattern histogram-based facial recognition methods under various weather conditions,” Technol., vol. 9, no. 31, pp. 1–12, 2021.
[19] J. Shotton et al., “Real-time human pose recognition in parts from single depth images,” in Proc. CVPR, Jun. 2011, pp. 1297–1304.
[20] R. Gross and J. Shi, “The CMU motion of body (MoBo) database,” Robotics Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-RI-TR-01-18, Jun. 2001.
[21] S. Zheng, J. Zhang, K. Huang, R. He, and T. Tan, “Robust view transformation model for gait recognition,” in Proc. 18th IEEE Int. Conf. Image Process., Sep. 2011, pp. 2077–2080.
[22] H. Iwama, M. Okumura, Y. Makihara, and Y. Yagi, “The OU-ISIR gait database comprising the large population dataset and performance evaluation of gait recognition,” IEEE Trans. Inf. Forensics Security, vol. 7, no. 5, pp. 1511–1521, Oct. 2012.
[23] R. Chereshnev and A. Kertész-Farkas, “HuGaDB: Human gait database for activity recognition from wearable inertial sensor networks,” in Proc. Int. Conf. Anal. Images, Social Netw. Techs. Cham, Switzerland: Springer, 2017, pp. 131–141.
[24] S. Zhou, C. Chen, G. Han, and X. Hou, “Deep convolutional neural network with dilated convolution using small size dataset,” in Proc. Chin. Control Conf. (CCC), Jul. 2019, pp. 8568–8572.
[25] S. Sabour, N. Frosst, and G. E. Hinton, “Dynamic routing between capsules,” 2017, arXiv:1710.09829.
[26] N. Baker, H. Lu, G. Erliklihan, and P. J. Kellman, “Deep convolutional networks do not classify based on global object shape,” PLOS Comput. Biol., vol. 14, no. 12, pp. 1–43, 2018.
[27] A. J. Young, T. A. Kuiken, and L. J. Hargrove, “Analysis of using EMG and mechanical sensors to enhance intent recognition in powered lower limb prostheses,” J. Neural Eng., vol. 11, no. 5, Oct. 2014, Art no. 056021.
[28] J. A. Spanias, A. M. Simon, K. A. Ingraham, and L. J. Hargrove, “Effect of additional mechanical sensor data on an EMG-based pattern recognition system for a powered leg prosthesis,” in Proc. 7th Int. IEEE/EMBS Conf. Neural Eng. (NER), Apr. 2015, pp. 639–642.