Construction of Myoelectric Signal Classifier Using LSTM and Efficacy of Prior Learning and Relearning Processes

Kazuya Kishida, Kei Hasegawa, and Kiyotaka Kamata

Abstract—**The use of myoelectric signals to control myoelectric prosthetic hands has been the object of numerous studies. Recently, machine learning techniques have been proposed as suitable methods to construct a classifier that sorts myoelectric signals. However, machine learning needs a lot of learning data from the user in order to generate an accurately functioning classifier, thus placing a considerable burden on the user. In this study, we discuss the use of myoelectric signals from non-users to build the classifiers and we introduce a relearning process to the construction protocol. We consider feedforward neural networks and long short-term memory to construct the classifiers using non-users' and user's myoelectric signals. In order to show the validity of our method, we discuss the results of experiments performed to test the classification of the myoelectric signals for accuracy rate, precision rate, and recall rate.**

Index Terms—**Long Short-Term Memory, Recurrent Neural Networks, Feedforward Neural Networks, Support Vector Machine, Myoelectric Signals, Myoelectric Prosthetic Hand.**

I. Introduction

Many studies on myoelectric prosthetic hands for people who lost their movement function due to an accident or illness have been published in recent years [1]-[3]. A myoelectric prosthetic hand refers to an electric artificial hand that controls movements by estimating the user's intention from the weak electrical signals (hereinafter the myoelectric signals) generated by the activity of the remaining muscles. Although it is necessary to practice in order to move the myoelectric prosthetic hand according to the user's intention, its use can improve their quality of life. Therefore, we think that it is worth using the myoelectric prosthetic hand. In recent studies, machine learning has been proposed as a method to construct the classifier that sorts the myoelectric signals [3]-[5]. However, myoelectric signals have individual differences, noises are loud, and slight changes happen even in the same repeated action of the user. Furthermore, a lot of learning data from the user are needed to construct a high accuracy classifier system. So, we think that the traditional method places a large burden on the user for constructing the classifier. Therefore, we considered the method constructing classifier using artificial neural networks (ANNs) as one of the methods reducing user's burden[6].

K. Kishida is with the National Institute of Technology, Kagoshima College, 1460-1 Shinko, Hayato, Kirishima, Kagoshima, 899-5102, Japan. e-mail: kishida@kagoshima-ct.ac.jp.

K. Hasegawa is with the Development of Human and Engineered Environment Studies, Frontier Sciences, The University of Tokyo. e-mail: 5skhasegawa32@gmail.com.

K. Kamata is with the National Institute of Technology, Kagoshima College. e-mail: kamata@kagoshima-ct.ac.jp.

In this paper, in order to reduce this burden, we proceed with the discussion of our considered method further through outcomes of additional experiments and introduction of prior learning and relearning process. We consider methods that construct the classifiers using both user's and non-users' myoelectric signals. By adding non-users' myoelectric signals, we think that the classifier can be constructed using a smaller number of user's learning data than those the traditional method needs, because non-users' myoelectric signals provide a sufficient number of learning data. Furthermore, we think that our method increases the generality of the classifier by compensating for the user's differences of myoelectric signals. This is because the inclusion of non-users' myoelectric signals widens the input space of learning data compared with the method using only user's myoelectric signals.

The methods we considered apply feedforward neural networks (FFNNs) and long short-term memory (LSTM) in artificial neural networks. We think that time series data of myoelectric signals can be effective to construct the classifier, so the use of LSTM, which handles long time series data well, is expected to improve the classification performance. The use of ANNs ensures the strongest point of our method, i.e., that the constructed classifiers can be relearned using new learning data and be adjusted for the user through the relearned process. For comparison, we also construct a classifier which uses support vector machine (SVM) in additional to the above two methods using ANNs.

We performed some experiments in order to show the validity of the methods we took into consideration. We evaluate the constructed classifiers in point of accuracy rate, precision rate and recall rate, and discuss our findings.

II. Myoelectric Signals and Measurement

A. *Myoelectric Signals*

Myoelectric signals are generated when the brain transmits command signals to muscle fibers. Generally, the inside of a cell membrane of muscle fibers has an electrical potential of -80 mV compared with the outside. This electrical potential reverses as a result of depolarization that occurs by receiving the command signals from the brain. The reversal electrical potential, called action potential, propagates along muscle fibers interactively. This action potential is called electromyogram (EMG) [7].

There are two ways to measure the myoelectric signal. One is the needle EMG method. This method is applied to clinical uses [7] because it enables to recognize changes in the myoelectric signal with high spatial resolution. However, the needle EMG, as an invasive procedure, hurts the myoelectric prosthetic user physically. The other option is the surface EMG method, which measures the myoelectric signals through electrodes placed on the skin surface. This method has low invasiveness. Moreover, attachment and detachment of the electrode are easy. The frequency range of surface EMG is about 5∼500 Hz [8].

B. *Measurement of Myoelectric Signals*

We applied the surface EMG method in consideration of its limited physical burden. 10 healthy adult participants (9 males, 1 female, aged 21 and 22 years) were asked to perform six hand motions (relaxation, grasping, opening, palmer flexion, dorsal flexion and ulnar flexion) as shown in Fig. 1. Then, we obtained the myoelectric signals from four measurement positions (Fig. 2), which were flexor digitorum superficialis muscle (FDS), flexor carpi ulnaris muscle (FCU), extensor carpi radialis longus muscle (ECRL) and extensor digitorum communis muscle (EDC). Participants began performing a hand motion at the same time as measurement was signaled to start, and we measured the myoelectric signals by bipolar measurement, which uses two disposable electrodes arranged at each measurement position. In addition, body earth is arranged on an elbow that is not affected by electrical potential of measurement position. This method enables noises to decrease, since there is mutual noise cancellation between the two electrodes. The distance between the two electrodes of each pair was 2 cm. The myoelectric signals were measured at a sampling frequency of 6000 Hz for 500 ms. Hence, myoelectric signals data with time-series length of 3000 were obtained. In consideration of the frequency range of the myoelectric signals and of the utility frequency range, myoelectric signals with frequency range of less than 5 Hz, 59.5∼60.5 Hz, and more than 1000 Hz were eliminated by digital filtering through MATLAB after obtaining the data. Participants were asked to reduce the electric impedance of the skin to less than 5 kΩ by skin treatment before the measurement. Figure 3 and TABLE I show the measurement system and measurement conditions, respectively.

Fig. 1. Classification of six hand motions

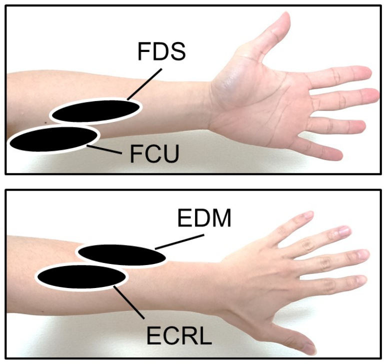

Fig. 2. Measurement positions

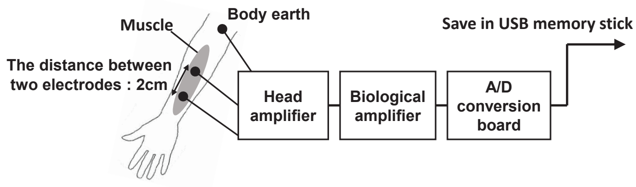

Fig. 3. Measurement system

TABLE I
MEASUREMENT CONDITIONS

Head amplifier	BA-U001
Biological amplifier	BA-1008 Electrodes : Disposable electrode Gain : 74 dB
A/D conversion board	ADA16-32/2(CB)F
Sampling frequency	6000 Hz
Sampling time	500 ms
Filter	High-pass filter : 5 Hz Low-pass filter : 1000 Hz Notch filter : 50.5∼60.5 Hz
Electric impedance of the skin	less than 5 kΩ
Participants	Four 21-year-old males Five 22-year-old males One 22-year-old female

III. CLASSIFIER AND THE LEARNING METHOD

A. *Feedforward Neural Networks*

FFNNs are composed of layers, each having some neurons. Each neuron connects to the neurons of the following layer. A weighted signal moves in one direction from the input layer to the output layer. FFNNs having three layers are shown in Fig. 4.

The equations for the outputs of each hidden layer neuron and output layer neuron are, respectively, given as

$$z_j = f\left(\sum_{i=0}^{n} w_{ji} x_i\right) \tag{1}$$

$$y_k = f\left(\sum_{j=0}^{m} v_{kj} z_j\right) \tag{2}$$

where x_i is the input variable from the i-th neuron in the input layer, z_j is the output variable of the j-th neuron in the hidden layer, y_k is the output variable of the k-th neuron in the output layer, w_{ji} is the weight between the i-th neuron in the input layer and j-th neuron in the hidden layer, v_{kj} is the weight between the j-th neuron in the hidden layer and the k-th neuron in the output layer, n and m are constants corresponding to the numbers of input layer neurons and

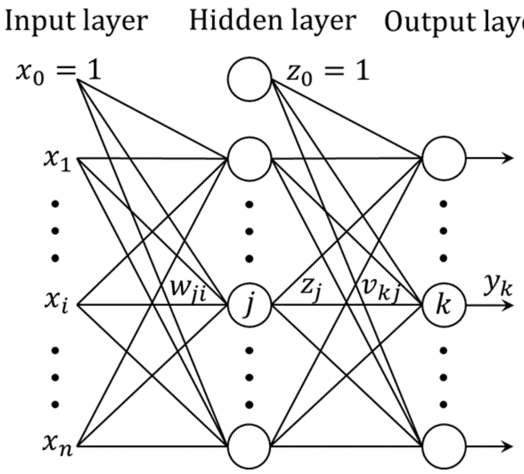

Fig. 4. Structure of FFNNs

hidden layer neurons, respectively. Moreover, $f(u)$ is the activation function given by the following equation:

$$f(u) = \frac{1}{1 + e^{-u}} \tag{3}$$

$$E = \sum_k (d_k - y_k)^2 \tag{4}$$

Backpropagation (BP) is applied as a learning algorithm. BP adjusts the weights of the network so that the error function E (Eq. (4)) is minimized. The error function indicates the error between the calculated output and the supervised signal. All parameters are updated according to the BP algorithm.

The classifier constructed by FFNNs consisted of three layers: an input layer, a hidden layer and an output layer. The input, hidden, output layers had 4, 50, 6 neurons, respectively.

B. Long Short-Term Memory

LSTM is a kind of recurrent neural network (RNN) architecture that is able to handle long time-series data. It was proposed by S. Hochreiter and J. Schmidhuber [9] and has feedback connections. Unlike FFNNs, the input values to the hidden layer are not only weighted signals from the input layer, but also weighted signals from the previous hidden layer (Fig. 5). This structure allows the past input to influence the output and enables LSTM to capture the feature of time-series variations. Moreover, unlike RNNs, LSTM has a structure that replaces each neuron in a recurrent hidden layer with an LSTM block. Each LSTM block contains a memory cell, an input gate, a forget cell and an output gate (Fig. 6). The memory cell plays the role of storing the internal state. Simple RNNs are not able to handle long time-series data because of the vanishing gradient problem. However, LSTM is able to do so if each gate switches conveniently.

The equation for the internal state of a memory cell is given by Eq. (5). In addition, the equation for the output of an input gate, a forget gate and an output gate are, respectively, given by Eqs. (6) - (8):

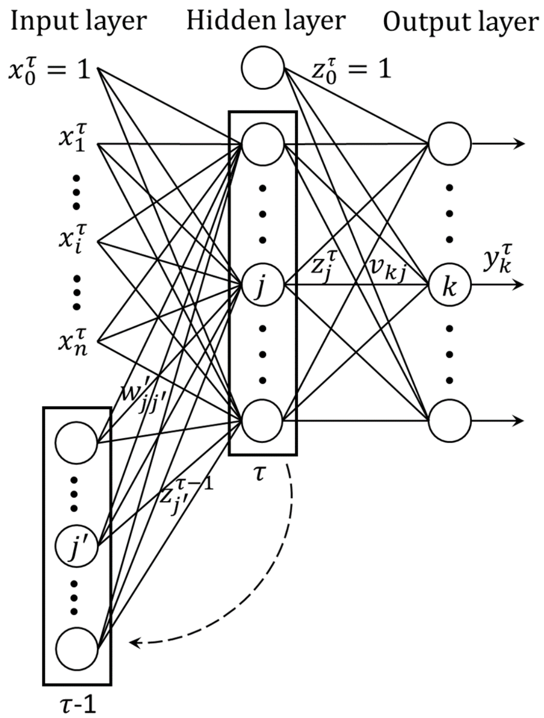

Fig. 5. Structure of LSTM

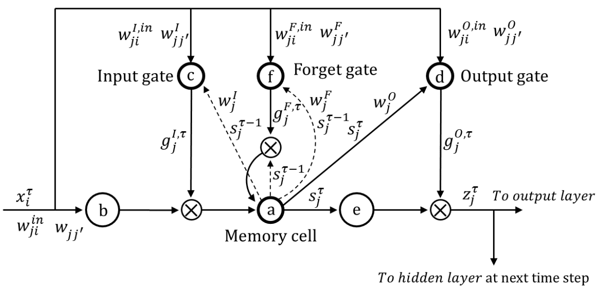

Fig. 6. LSTM block in the recurrent hidden layer

$$s_j^\tau = g_j^{F,\tau} s_j^{\tau-1} + g_j^{I,\tau} f\left(\sum_{i=0}^{n} w_{ji}^{in} x_i^\tau + \sum_{j'=0}^{m} w_{jj'} z_{j'}^{\tau-1}\right) \tag{5}$$

$$g_j^{I,\tau} = f\left(\sum_{i=0}^{n} w_{ji}^{I,in} x_i^\tau + \sum_{j'=0}^{m} w_{jj'}^{I} z_{j'}^{\tau-1} + w_j^I s_j^{\tau-1}\right) \tag{6}$$

$$g_j^{F,\tau} = f\left(\sum_{i=0}^{n} w_{ji}^{F,in} x_i^\tau + \sum_{j'=0}^{m} w_{jj'}^{F} z_{j'}^{\tau-1} + w_j^F s_j^{\tau-1}\right) \tag{7}$$

$$g_j^{O,\tau} = f\left(\sum_{i=0}^{n} w_{ji}^{O,in} x_i^\tau + \sum_{j'=0}^{m} w_{jj'}^{O} z_{j'}^{\tau-1} + w_j^O s_j^{\tau}\right) \tag{8}$$

where x_i^τ is the input variable from the i-th neuron at time step τ in the input layer, z_j^τ is the output variable of the j-th LSTM block at time step τ, s_j^τ is the internal state of the j-th LSTM block at time step τ, $g_j^{I,\tau}$ is the output variable of the input gate of the j-th LSTM block at time step τ, $g_j^{F,\tau}$ is the output variable of the forget gate of the j-th LSTM block at time step τ, $g_j^{O,\tau}$ is the output variable of the output gate of the j-th LSTM block at time step τ, w_{ji}^{in} is the weight between the i-th neuron in the input layer and the j-th LSTM block, $w_{jj'}$ is the weight between the j-th LSTM block in the hidden layer at time step $(\tau-1)$ and the j-th LSTM block at time step τ, $w_{ji}^{I,in}$ is the weight between the i-th neuron in the input layer and input gate of the j-th LSTM block, $w_{jj'}^{I}$ is the weight between the j-th LSTM block at time step $(\tau-1)$ and the input gate of the j-th LSTM block at time step $(\tau-1)$, $w_{ji}^{F,in}$ is the weight between the i-th neuron in the input layer and the forget gate of the j-th LSTM block, $w_{jj'}^{F}$ is the weight between the j-th LSTM block at time step $(\tau-1)$ and the forget gate of the j-th LSTM block at time step $(\tau-1)$, $w_{ji}^{O,in}$ is the weight between the i-th neuron in the input layer and output gate of j-th LSTM block, $w_{jj'}^{O}$ is the weight between the j-th LSTM block at time step $(\tau-1)$ and the output gate of the j-th LSTM block at time step $(\tau-1)$, w_j^I is the weight of the peephole connection of the input gate, w_j^F is the weight of the peephole connection of the forget gate, w_j^O is the weight of the peephole connection of the output gate, n and m are constants that correspond to the numbers of input layer neurons and hidden layer neurons, respectively. The output of LSTM block, the output of neuron in hidden layer, is given by the following equation.

$$z_j^\tau = g_j^{O,\tau} f(s_j^\tau) \tag{9}$$

The equation for the outputs of each output layer neuron is given as:

$$y_k^\tau = f\left(\sum_{j=0}^{m} v_{kj} z_j^\tau\right) \tag{10}$$

where y_k^τ is the output variable of the k-th neuron at time step τ in the output layer. In this study, the error function E is defined as Eq. (11)

Backpropagation through time (BPTT) [10][11] is applied as the learning algorithm for LSTM. BPTT adjusts the weights of the network so that the error function E is minimized. The error function indicates the error between the calculated output and the supervised signal. All parameters are updated according to the BPTT algorithm.

$$E = \frac{1}{TN}\sum_{\tau=0}^{T}\sum_{k=0}^{N}(d_k^\tau - y_k^\tau)^2 \tag{11}$$

where d_k^τ is the supervised signal of the k-th neuron at time step τ in the output layer, N is a constant that corresponds to the number of output layer neurons and T is the constant that indicates the length of a time-series input.

The classifier constructed by LSTM consisted of three layers: an input layer, a hidden layer and an output layer. The input, hidden, output layers had 4, 50, 6 neurons, respectively.

C. Support Vector Machine

SVM is a learning model for pattern recognition. SVM is applied as a solution of a binary classification problem. Support vectors are feature vectors chosen from the learning data to define a decision function. The margin is the distance between the support vectors and the classification boundary that classifies into two classes. SVM establishes a classification boundary so that the margin is maximized (Fig. 7).

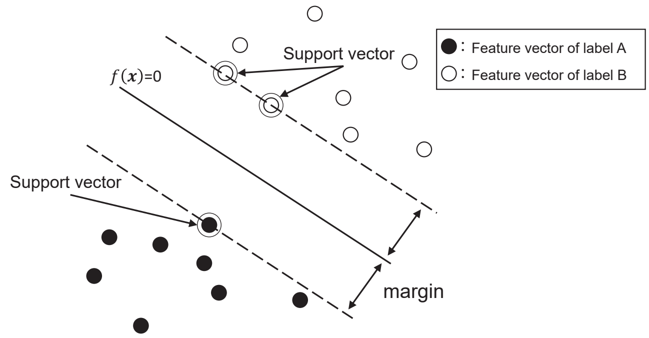

Fig. 7. Classification boundary and margin in SVM

The equation for the decision function of a linear SVM is defined as

$$f(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b \tag{12}$$

where $\boldsymbol{x}$ is the input vector, $\boldsymbol{w}$ is the normal vector of the classification boundary and b is an intercept of the decision function. $\boldsymbol{w}$ and b are parameters to shape the decision function. These parameters are found through the Lagrange multipliers method. However, there is a limit to what linear SVM classifies as linearly inseparable input. Therefore, nonlinear SVM with the Kernel function is introduced. The kernel function makes it possible to recognize linearly inseparable input by converting a linearly inseparable input distribution into a linearly separable input distribution. The Radial Basis Function (Eq. (13)) is applied as the Kernel function in this study. The decision function of nonlinear SVM is defined as Eq. (14). The dual variable $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ is found by the optimization problem, which is called dual problem (Eq. (15)), to define the decision function.

$$K(\boldsymbol{x_i}, \boldsymbol{x_j}) = exp(-\gamma||\boldsymbol{x_i} - \boldsymbol{x_j}||^2) \tag{13}$$

$$f(x) = \sum_{i\in[n]} \alpha_i y_i K(x_i, x) + b \tag{14}$$

$$\begin{gathered}\max_{\boldsymbol{\alpha}} -\frac{1}{2}\sum_{i,j\in[n]} \alpha_i\alpha_j y_i y_j K(x_i, x_j) + \sum_{i\in[n]} \alpha_i \\ subject \quad to \quad \sum_{i\in[n]} \alpha_i y_i = 0 \\ 0 \le \alpha_i \le C, i \in [n]\end{gathered} \tag{15}$$

where $\boldsymbol{x_i}$ is the input vector of the i-th learning data, y_i is a label of the i-th learning data which consists of $y_i \in \{-1, 1\}$, C is the regularization parameter to permit misclassification, γ is the parameter to decide the gradient of Kernel and n is a constant that represents the number of learning data. In this study, C and γ were empirically determined.

The multi-class classification is made by combining some two-class classifiers. This study applied One-Versus-One as a method of multi-class classification. For six-class classification, ${}_6\mathrm{C}_2$ (= 15) of two-class classifiers were prepared.

IV. Experiments to Classify Myoelectric Signals

A. Making of Input-Output Data

The amplitude of the myoelectric signal changes depending on the force of the squeezed muscle. Maximal voluntary contraction (MVC) is the muscle strength which a human maximizes by contracting one's muscle intendedly and voluntarily. The amplitude and wave density of the myoelectric signals increase with an increase of the muscle load. There is a linear relationship between the muscle activity and the muscle load within 10∼90% of MVC [7]. In this study, time-integration was applied as feature variables.

The integration value was applied as input data of FFNNs and SVM. To derive this value, we summed up the absolute values of each myoelectric signal data obtained from four measurement positions within 500 ms (3000 points).

The equation for the input value x_i^k obtained from a measurement position i (i=1,. . . ,4) (Fig. 2) in a hand motion k (k=a,. . . ,f) (Fig. 1) was given as:

$$x_i^k = \sum_{n=1}^{3000} \left| Data_i^k(n) \right| \tag{16}$$

where $Data_i^k(n)$ is the value of the n-th point from a measurement position i in a hand motion k.

When deriving the input data for LSTM, firstly, we divided the myoelectric signals of 500 ms (3000 points) into 20 parts; each part contained the myoelectric signals of 25 ms (150 points). Then, each part was time-integrated within 25 ms (150 points).

The equation for the input value $x_i^k(\tau)$ of input i and time step τ (τ = 0, . . . , 19) in a hand motion k was given as:

$$x_i^k(\tau) = \sum_{n=1}^{150} \left| Data_i^k(n + 150 \cdot \tau) \right| \tag{17}$$

TABLE II shows each supervised signal, for FFNNs and LSTM, of six neurons in the output layer to each hand motion.

TABLE II
Supervised Signals of FFNNs and LSTM

Hand Motion	Output of k-th neuron in the output layer					
	1	2	3	4	5	6
(a) Relaxation	0	0	0	0	0	1
(b) Grasping	0	0	0	0	1	0
(c) Opening	0	0	0	1	0	0
(d) Palmar Flexion	0	0	1	0	0	0
(e) Dorsal Flexion	0	1	0	0	0	0
(f) Ulnar Flexion	1	0	0	0	0	0

In this study, we obtained input-output data from 10 participants. We obtained 50 input-output data for each hand motion from each participant, so that the number of each participant's input-output data was 300. Therefore, 3000 input-output data were obtained from 10 participants.

B. Experiment I

In Experiment I, we organized participants' input-output data into five data sets. Using four data sets as learning data, we constructed classifiers based on FFNNs, LSTM and SVM. Then, we evaluated each constructed classifier using the remaining data set as evaluation data. The processing flow to build five data sets was as follows:

[Step 1] Extract 10 input-output data for each hand motion from each participant's 50 input-output data to make data set 1, which stores a total of 600 input-output data.

[Step 2] Extract 10 input-output data, which is not used in [Step 1], for each hand motion from each participant's remaining 40 input-output data to make data set 2, which stores a total of 600 input-output data.

[Step3] Repeat [Step 2] to make data set 3∼5.

Figure 8 shows the method to build data set 1.

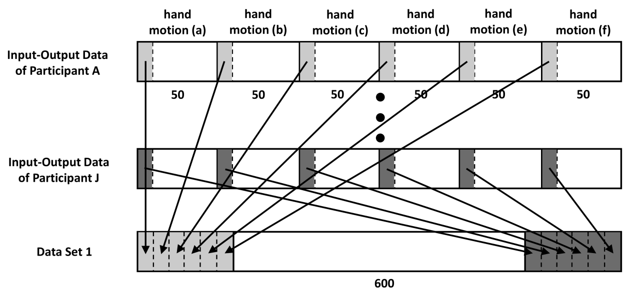

Fig. 8. Method to build data set 1

5-fold-validation was applied as an evaluation method. In FFNNs and LSTM for Experiment I, the epoch was 3000, and the learning rate was 0.005. Also, the learning rate of LSTM was changed automatically by applying the Adam algorithm [12]. The parameters of SVM were C=10 and γ=10.

TABLE III
Results of Experiment I

Evaluation Items	Accuracy[%]	Precision[%]	Recall[%]
FFNNs	89.6	89.8	89.8
LSTM	93.8	93.8	93.8
SVM	91.1	91.2	91.2

TABLE III shows the average results of five folds for Experiment I. in terms of accuracy rate, precision rate and recall rate. The corresponding values were more than 89% for all three classifiers: FFNNs, LSTM and SVM. In particular, the classifier obtained using LSTM achieved high discrimination performance. We considered that the LSTM model could acquire time-series variations of the myoelectric signals as a feature variable from input data by handling the locally divided myoelectric signals data as time-series data.

C. Experiment II

Experiment II was performed to discuss the generality of the classifier constructed using non-users' input-output data. In this experiment, we assumed 9 participants over 10 to be non-users, with the remaining one participant considered as a user. We used 9 non-users' input-output data as learning data to construct each classifier. Then, we evaluated each constructed classifier using the remaining one user's input-output data as evaluation data. In FFNNs and LSTM for

TABLE IV
RESULTS OF EXPERIMENT II

Acuracy rate[%]

Learning Data	B-J	A,C-J	A,B,D-J	A-C,E-J	A-D,F-J	A-E,G-J	A-F,H-J	A-G,I,J	A-H,J	A-I
Evaluation Data	A	B	C	D	E	F	G	H	I	J
FFNNs	84.7	85.0	79.7	86.3	88.3	61.0	83.7	85.0	64.0	91.3
LSTM	76.7	82.3	76.0	86.3	89.7	82.7	76.3	86.0	69.3	83.3
SVM	87.3	83.3	85.7	86.0	87.7	71.3	79.3	89.0	68.0	77.7

Precision rate[%]

Learning Data	B-J	A,C-J	A,B,D-J	A-C,E-J	A-D,F-J	A-E,G-J	A-F,H-J	A-G,I,J	A-H,J	A-I
Evaluation Data	A	B	C	D	E	F	G	H	I	J
FFNNs	86.0	86.0	87.0	90.0	89.0	72.0	90.0	89.0	53.0	92.0
LSTM	72.0	84.0	86.0	88.0	90.0	85.0	86.0	89.0	75.0	84.0
SVM	90.0	83.0	88.0	90.0	90.0	78.0	90.0	92.0	57.0	76.0

Recall rate[%]

Learning Data	B-J	A,C-J	A,B,D-J	A-C,E-J	A-D,F-J	A-E,G-J	A-F,H-J	A-G,I,J	A-H,J	A-I
Evaluation Data	A	B	C	D	E	F	G	H	I	J
FFNNs	85.0	85.0	80.0	86.0	88.0	61.0	84.0	85.0	64.0	91.0
LSTM	77.0	82.0	76.0	86.0	90.0	83.0	76.0	86.0	69.0	83.0
SVM	87.0	83.0	86.0	86.0	88.0	71.0	79.0	89.0	68.0	78.0

Experiment II, the epoch was 3000, and the learning rate was 0.005. Also, the learning rate of LSTM was changed automatically by applying the Adam algorithm [12]. The parameters of SVM were C=1 and γ=1. The experiment was performed ten times for 10 participants.

TABLE IV shows the average results in Experiment II. The majority of results for accuracy rate, precision rate and recall rate were more than 70%. However, in the case of evaluation data F, the accuracy rate and recall rate were 60% units in FFNNs. In the case of evaluation data I, the accuracy rate and recall rate were also 60% units in all three classifiers. In addition, in the case of evaluation data I, the precision rate was 50% units in FFNNs and SVM. According to the results of Experiment II, we found that it is possible to construct a classifier having discrimination performance by only using non-users' data. It is assumed that differences between participant F's and I's generating factors of myoelectric signals and others' ones such as muscle strength, caused low discrimination performance. As a consequence, vast amounts of non-users' input-output data are needed for constructing classifiers having high generality.

D. Experiment III

Experiment III was performed to assess the method where a classifier built using non-users' data undergoes a relearning process using user's data.

The experimental protocol consists of the following steps: firstly, we constructed classifiers with FFNNs and LSTM using 9 participants' input-output data (2700 input-output data), assuming them to be non-users' data. This step is called prior learning. Secondly, we made five data sets, with each data set comprising 60 input-output data from the remaining one participant's 300 input-output data, assuming these to be user's data. Thirdly, the classifier relearned using four data sets. This step is called relearning process. Lastly, we evaluated the constructed classifier using the remaining data set. By adopting this protocol, the classifier constructed through the relearning process is updated to fit the user and is expected to get a higher discrimination performance. 5-fold validation was applied as an evaluation method. Figure 9 shows the method for building a data set from the user. In FFNNs and LSTM, the epochs in prior learning and relearning were 3000 respectively, and the learning rate was 0.005. Also, the learning rate of LSTM was changed automatically by applying the Adam algorithm [12]. SVM is not able to relearn, so we only took into consideration only FFNNs and LSTM in the experiment with prior learning. TABLE V shows the related average results of 5 folds.

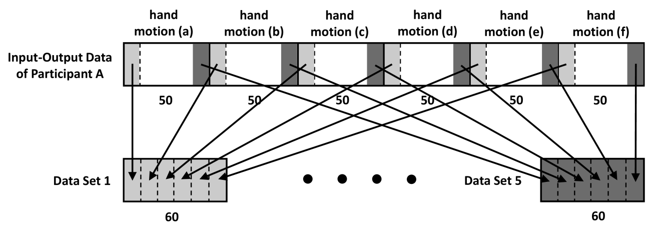

Fig. 9. Method for building a data set from participant A

TABLE V shows LSTM achieved higher discrimination performance (more than 94.0%) than FFNNs, and the discrimination performance of LSTM was the best in this paper. The differences are significant at $p < 0.01$ measured using the two-tailed t-test for accuracy rate (t-stat = 3.62, p-value = 0.0056), precision rate (t-stat = 3.81, p-value = 0.0041) and recall rate (t-stat = 3.62, p-value = 0.0056).

Also, we constructed classifiers without prior learning using only the user to compare with the discrimination performance of the experiment with prior learning. In FFNNs and LSTM, the epochs were 6000, and the learning rate was 0.005. Also, the learning rate of LSTM was changed automatically by applying the Adam algorithm [12]. The

TABLE V
RESULTS OF EXPERIMENT III WITH PRIOR LEARNING

Accuracy rate[%]

Prior Learning Data	B-J	A,C-J	A,B,D-J	A-C,E-J	A-D,F-J	A-E,G-J	A-F,H-J	A-G,I,J	A-H,J	A-I
Additional Learning and Evaluation Data	A	B	C	D	E	F	G	H	I	J
FFNNs	100.0	91.7	93.3	93.0	91.7	92.3	97.3	99.7	89.0	97.3
LSTM	99.7	94.0	94.7	94.7	96.3	96.0	99.7	99.7	96.0	99.3

Precision rate[%]

Prior Learning Data	B-J	A,C-J	A,B,D-J	A-C,E-J	A-D,F-J	A-E,G-J	A-F,H-J	A-G,I,J	A-H,J	A-I
Additional Learning and Evaluation Data	A	B	C	D	E	F	G	H	I	J
FFNNs	100.0	92.4	93.6	93.0	92.0	93.4	97.2	99.6	90.2	97.2
LSTM	99.6	94.4	95.2	95.0	97.2	96.4	99.6	99.6	96.2	99.4

Recall rate[%]

Prior Learning Data	B-J	A,C-J	A,B,D-J	A-C,E-J	A-D,F-J	A-E,G-J	A-F,H-J	A-G,I,J	A-H,J	A-I
Additional Learning and Evaluation Data	A	B	C	D	E	F	G	H	I	J
FFNNs	100.0	91.8	93.4	93.0	91.6	92.2	97.2	99.6	89.0	97.2
LSTM	99.6	94.0	94.8	94.8	96.2	96.0	99.6	99.6	96.0	99.4

TABLE VI
RESULTS OF EXPERIMENT III WITHOUT PRIOR LEARNING

Accuracy rate[%]

Additional Learning and Evaluation Data	A	B	C	D	E	F	G	H	I	J
FFNNs	99.7	91.0	93.3	92.3	90.0	91.7	96.7	99.3	86.7	96.7
LSTM	97.7	85.0	91.0	83.0	92.3	92.7	96.0	93.0	87.0	94.0
SVM	100.0	89.0	92.0	93.7	90.0	93.3	96.7	99.7	85.0	97.0

Precision rate[%]

Additional Learning and Evaluation Data	A	B	C	D	E	F	G	H	I	J
FFNNs	99.6	91.8	93.6	92.8	90.2	93.0	96.8	99.2	87.6	97.0
LSTM	97.6	83.2	93.2	80.2	90.8	93.4	96.2	93.8	89.0	94.6
SVM	100.0	89.0	92.2	93.8	90.4	94.0	96.6	99.6	86.2	97.2

Recall rate[%]

Additional Learning and Evaluation Data	A	B	C	D	E	F	G	H	I	J
FFNNs	99.6	91.0	93.2	92.2	90.0	91.6	96.6	99.2	86.6	96.8
LSTM	97.6	83.2	93.2	80.2	90.8	93.4	96.2	93.8	89.0	94.6
SVM	100.0	89.0	91.8	93.6	89.8	93.4	96.6	99.6	84.8	97.0

TABLE VII
T-TEST RESULTS WITH AND WITHOUT PRIOR LEARNING

	FFNNs			LSTM		
	accuracy	precision	recall	accuracy	precision	recall
t-stat	3.64	2.67	3.83	5.87	4.61	5.89
p-value	0.0054	0.0256	0.0040	0.0002	0.0013	0.0002

parameters of SVM were C=10 and γ=10. TABLE VI shows the results for the constructed classifiers, which were evaluated by 5-fold-validation. According to TABLE V, the classifiers constructed without prior learning did not show good performances in point of accuracy rate, precision rate and recall rate. We think that the classifier constructed using only the user's myoelectric signals cannot achieve high performance because insufficient input data for learning exist in the input space.

The results shown in TABLE V and TABLE VI, make clear that the classifier with prior learning achieved better discrimination performance than the classifier without prior learning. This indicates the validity of using non-users' data for constructing the classifier and an effectiveness of prior learning. We think that using multiple non-users' input-output data for learning, the lack of input-output data for learning using only the user's data are compensated. We found that the method we took into consideration, LSTM with prior learning, had better discrimination performance than SVM. The results of Experiment III proved that the method with prior learning is able to lighten the user's burden because the use of non-users' data enables to reduce the number of user's data required for learning. Therefore, we achieved the goal of this study. Additionally, the differences between with and without prior learning are significant at $p < 0.05$ measured using a two-tailed t-test in accuracy rate, precision rate and recall rate. This is shown in TABLE VII. The differences between our proposed method and SVM are significant at $p < 0.05$ measured by two-tailed t-test in accuracy rate (t-stat = 3.16, p-value = 0.0115), precision rate (t-stat = 3.31, p-value = 0.0091) and recall rate (t-stat = 3.18, p-value = 0.0111). However, there were merely 10 participants in this experiment. Future studies will need to involve a much larger number of participants in order to improve the significance of the results.

V. Conclusion

In this paper, we considered a strategy to reduce the user's burden when constructing a classifier for the control of a myoelectric prosthetic hand. To this end, we developed a method that uses both user's and non-users' myoelectric signals and introduced a relearning process to obtain a classifier that both fits the user and has high discrimination performance. The introduction of non-users' myoelectric signals ensures that a sufficient number of learning data is provided and that learning data are widely distributed in the input space. We took into consideration both FFNNs and LSTM as methods to construct the classifiers. The classifiers built with FFNNs and LSTM can be reconstructed using extra learning data after they have been constructed. Furthermore, in order to show the validity of our methods, we performed experiments to classify the myoelectric signals. As shown by our results, LSTM performs better in point of accuracy rate, precision rate and recall rate. In Experiment III, we constructed classifiers by two steps. In the first step, the classifiers were constructed using non-users' learning data. In the following step, the classifiers ware reconstructed by relearning user's learning data, that is by performing a relearning process. We showed that the classifiers based on LSTM which had been constructed through these two steps had the best discrimination performance.

Future studies will focus on using more participants' input-output data in the classification to improve the significance of the results.

References

[1] S. Mallik and M. Dutta, "A Study on Control of Myoelectric Prosthetic Hand Based on Surface EMG Pattern Recognition," International Journal of Advance Research in Science and Engineering, vol. No. 6, Issue No. 07, pp635-646, Jul. 2017

[2] D. R. Damodar, U. V. Suthar and H. D.Solanki, "Myo-Electric Hand : Prosthetic Hand Replication Using EMG Based Approach," International Journal of Engineering Development and Research, vol. 6, Issue. 3, pp658-662, 2018

[3] T. Tsuji, O. Fukuda, M. Kaneko and K. Ito, "Pattern classification of time-series EMG signals using neural networks," International Journal of Adaptive Control and Signal Processing, pp829-848, 2000

[4] N. Bu, O. Fukuda and T. Tsuji, "EMG-Based Motion Discrimination Using a Novel Recurrent Neural Network", International Journal of Intelligent Information Systems, vol. 21, no. 2, pp113-126, 2013

[5] M. A. Oskoei and H. Hu, "Evaluation of Support Vector Machines in Upper Limb Motion Using Myoelectric Signal," published as a conference paper at International Conference on Biomedicals and Bioengineering 2008

[6] K.Kishida, K.Hasegawa and K.Kamata, "Construction of Classifier of Myoelectric Signals by using ANNs," Lecture Notes in Engineering and Computer Science:Proceedings of The World Congress on Engineering and Computer Science 2019,22-24 October, 2019, San Francisco, USA, pp356-361

[7] M. S. Stock, A. S. Drusch, and B. J. Thompson, "The minimum number of contractions required to examine the EMG amplitude versus isometric force relationship for the vastus lateralis and vastus medialis," International Journal of Electromyography and Kinesiology 24, pp827-834, 2014

[8] Y. Makino, A. Okada and H. Shinoda, "Measuring Myoelectric Potential Patterns Based on Two-Dimensional Signal Transmission Technology," published as a conference paper at International Council of Associations for Science Education 2006

[9] S. Hochreiter and J. Schmidhuber, "LONG SHORT-TERM MEMORY," Journal of Neural Computation, vol. 9, no. 8, pp1735-1780, 1997

[10] P. J. Werbos, "Backpropagation Through Time: What It Does and How to Do It," Proceedings of the IEEE, vol. 78, no. 10, pp1550-1560, 1990

[11] R. J. Williams and D. Zisper, "Gradient-Based Learning Algorithms for Recurrent Networks and Their Computational Complexity," In Y.Chauvin and D. E. Rumelhart, editors, "Theory, Architectures and Applications," pp433-486. Lawrence Erlbaum Publishers, 1995

[12] D. P. Kingma and J. L. Ba, "Adam: A Method for Stochastic Optimization," published as a conference paper at International Conference on Learning Representations 2015

[13] D. Inui, S. Ito and M. Sakaki, "Experimental Considerations on Signal Feature and Kernel/Parameters of SVM in Hand Motion Classification from sEMG," Journal of the Japan Society of Mechanical Engineering, vol. 79, no. 808, pp221-231, Dec. 2013