Journal of Signal Processing, Vol.18, No.4, pp.225-228, July 2014

SELECTED PAPER AT NCSP'14

Direct Estimation of Hand Motion Speed from Surface Electromyograms Using a Selective Desensitization Neural Network

Kazumasa Horie[1], Atsuo Suemitsu[2] and Masahiko Morita[1]

[1] University of Tsukuba
1–1–1 Tennodai, Tsukuba, Ibaraki 305-0006, Japan
Phone/FAX:+81-29-853-6554
E-mail: horie@bcl.esys.tsukuba.ac.jp
mor@bcl.esys.tsukuba.ac.jp

[2] Japan Advanced Institute of Science and Technology
1–1 Asahidai, Nomi, Ishikawa 923-1211, Japan
Phone/FAX:+81-761-51-1227
E-mail: sue@jaist.ac.jp

Abstract

During the application of surface electromyograms (EMGs) in human-machine interfaces, direct estimates of multiple hand motion speeds are required to facilitate operations that fully reflect the user's intentions. However, no practical methods are available for this purpose because conventional function approximators cannot learn the complex relationships between the motion speeds and surface EMGs within a practical period of time. By contrast, it has been shown that a selective desensitization neural network (SDNN) can learn complex input-output relationships with low computational costs. In this study, we propose a method for the direct estimation of hand motion speeds from surface EMGs using a SDNN. We estimated the motion speed in practice to assess the efficacy of this proposed method. Our experimental results show that the proposed method can estimate the approximate speeds of six basic motions in real time.

1. Introduction

Recently, methods have been proposed for classifying multiple types of hand motions from surface electromyograms (EMGs) [1, 2] to facilitate the development of EMG-based human-machine interfaces. However, estimates of hand motion speed are also required to allow these interfaces to fully reflect the user's intentions. Unfortunately, practical methods are not available for the direct estimation of hand motion speeds from surface EMGs.

The estimation of motion speeds is regarded as a function approximation problem, which describes the relationship between surface EMGs and motion speeds. However, this function is readily affected by individual differences in surface EMGs. Thus, we consider that a function approximator is required to approximate individual functions.

However, most function approximators are not suitable for application to EMG-based interfaces. For example, a multilayer perceptron (MLP) cannot learn complex relationships within a practical period of time; radial basis function networks are not suitable for practical applications since they have excessively high computational costs and require numerous training samples to learn relationships. Suitable function approximators for EMG-based devices should have the following characteristics.

- High learning ability
- High robustness against redundant inputs
- High generalizability
- Low computational costs for learning

Recent studies have shown that a selective desensitization neural network (SDNN) may fulfill these requirements. For example, Nonaka et al. applied a SDNN to the approximation of a two-variable function and showed that the SDNN had a high learning ability and high generalizability [3]. Based on previous studies, we consider that a SDNN may be highly suited for application to EMG-based devices.

Thus, the aim of the present study was to develop a method for the direct estimation of the speed of multiple hand motions from surface EMGs using SDNNs. The following sections describe the SDNN and our proposed estimation method. We also experimentally verified the efficacy of the proposed method.

2. Selective Desensitization Neural Network

SDNNs are known to provide good approximations of a wide range of functions using a small set of training data samples. In this section, we explain how a function $y = f(\boldsymbol{x})$ is approximated based on a SDNN given a continuous-valued input vector $\boldsymbol{x} = (x_1, \ldots, x_m)$ $(m > 2)$.

The input layer comprises m neural groups $(G^1, \ldots, G^m)$. Each group has n units that represent an input signal x_i, i.e., the input signal is represented in a distributed manner by the activity patterns of the units. If the range of possible values for input signals is divided into q ranges, each range corresponds to different activity patterns, which are configured using equal numbers, 1 and -1. In addition, similar patterns are encoded based on the proximity between the ranges of the input signals.

The middle layer comprises $m(m-1)$ neural groups $G'^{\mu,\nu}(\mu, \nu = 1, \ldots, m;\ \mu \neq \nu)$. The units in $G'^{\mu,\nu}$ are connected with both of the units in G^μ and G^ν, which are located in the input layer. This realizes a procedure called desensitization, which neutralizes the outputs of the units, regardless of their inputs and inner potential. For example, if the i-th unit of G^{μ} is desensitized by the j-th unit of G^{ν}, the output of the i-th unit of $G'^{\mu,\nu}$ is given by

$$g_i'^{\mu,\nu} = \frac{g_i^{\mu}(1+g_j^{\nu})}{2} \tag{1}$$

where g_i^{μ} is the output of the i-th unit of G^{μ} and g_j^{ν} is the output of the j-th unit of G^{ν}.

The output layer of the SDNN comprises l ($l > 2$) units, each of which is connected to all of the units in the middle layer. The output of the i-th unit in the output layer is calculated by

$$y_i = H\Big(\sum_{\mu,\nu(\mu\neq\nu)}\sum_{j=1}^{n} w_{ij}^{\mu,\nu} g_j'^{\mu,\nu} - h_i\Big) \tag{2}$$

$$H(u) = \begin{cases} 1 \ (u \geq 0) \\ 0 \ (otherwise) \end{cases} \tag{3}$$

where $w_{ij}^{\mu,\nu}$ is a synaptic weight from the j-th unit of $G'^{\mu,\nu}$ in the middle layer and h_i is a threshold. The final output y of the SDNN is determined based on the number of units with outputs of 1 in the output layer. Learning is achieved by error correction training in this network.

3. Motion Speed Estimation Method

Our proposed method has three components: surface EMG measurement, signal preprocessing, and function approximation. Figure 1 shows a flowchart that illustrates this method.

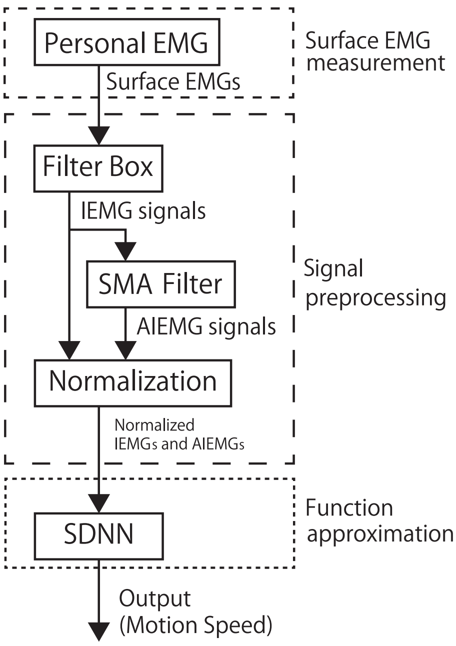


Figure 1: Flowchart of the proposed method

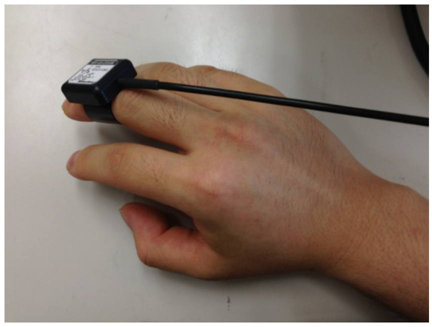
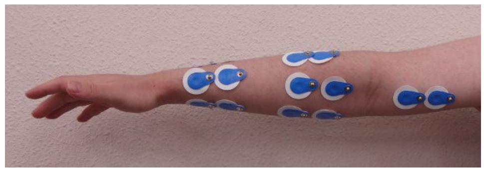
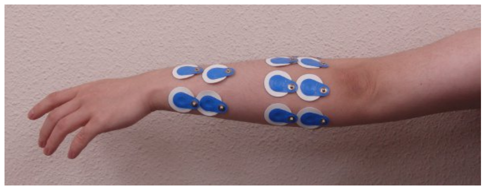


Figure 2: Installed gyroscope



(a) Thumb side



(b) Little finger side

Figure 3: Arrangement of EMG sensors

3.1 Measurement

A three-axis gyroscope (MP-G3-2000B, MicroStone Co. Ltd) is used to measure the motion speeds. The gyroscope is mounted on the middle finger (Figure 2) and it measures the angular velocity. $V(t)$ is calculated by

$$V(t) = \sqrt{V_x(t)^2 + V_y(t)^2 + V_z(t)^2} \tag{4}$$

where $V_x(t), V_y(t)$, and $V_z(t)$ are the angular velocities around the x-axis, y-axis, and z-axis of the gyroscope at time t, respectively. Finally, $V(t)$ is normalized against the maximum value during each time step and defined as the actual motion speed at time t.

The surface EMGs are measured at 11 points on the forearms (Figure 3). A relatively large number of sensors are required but it is not necessary to position the sensors accurately on specific muscles. The surface EMG is measured and sampled at 1 kHz using Personal-EMG (Oisaka Electronic Device Ltd) equipment and a 12-bit A/D converter.

3.2 Signal preprocessing

Surface EMGs are not suitable for use as input signals because of their fluctuations. Thus, it is necessary to extract features by signal preprocessing. However, we cannot employ

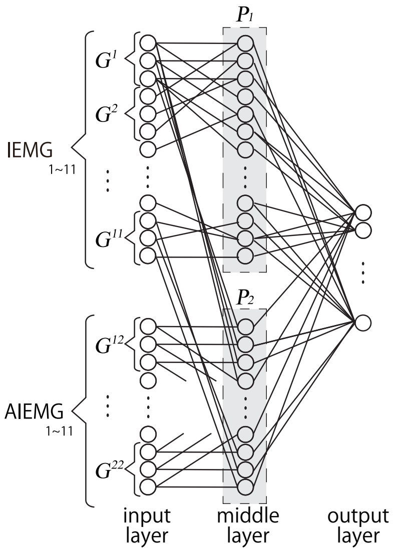


Figure 4: Structure of the SDNN

complex methods because they are not applicable to multiple motions. Using simple preprocessing, we obtain the integral of the surface EMG (IEMG) and its mean over the previous 200 ms (average IEMG: AIEMG) as features.

The IEMG can be obtained with little time lag but it is related to the forces generated by muscles rather than the motion speed. In contrast, the AIEMG is related more directly to the motion speed than the IEMG, but there is a large time lag between the AIEMG and the actual motion itself. Therefore, the combined use of both of these features can be expected to enhance the accuracy of motion speed estimation.

In the proposed method, a filter box (Oisaka Electronic Device Ltd) is used to convert the surface EMGs into IEMGs and a simple moving average (SMA) filter converts the IEMGs into AIEMGs. The IEMGs and AIEMGs are normalized against the maximum value during each time step before being used as inputs for the SDNN.

3.3 Function approximation

The SDNN is constructed as shown in Figure 4 and it is used as a function approximator. Note that the SDNN is prepared for each motion, but all SDNNs have the same structure and parameters.

The input layer comprises 11 neural groups ($G^1, \ldots, G^{11}$) and 11 other neural groups ($G^{12}, \ldots, G^{22}$), which represent the IEMG signals ($\mathrm{IEMG}_1, \ldots, \mathrm{IEMG}_{11}$) and AIEMG signals ($\mathrm{AIEMG}_1, \ldots, \mathrm{AIEMG}_{11}$), respectively. The activity patterns of the units are determined from the input signals using the parameters $n = 32$ and $q = 96$.

The middle layer has two parts (P_1 and P_2). P_1 and P_2 comprise 110 and 22 neural groups, respectively. The units of the neural groups in P_1 are connected with the units of two neural groups from $G^1, \ldots, G^{11}$. In P_2, the units of the neural groups are connected with the units in the corresponding pairs of the neural groups $(G^1, G^{12}), \ldots, (G^{11}, G^{22})$. In the middle layer, half of the units are desensitized by the corresponding units in the input layer.

The output layer comprises 70 units. The final output of the SDNN is calculated by $0.02k - 0.2$, where k is the number of units with outputs of 1 in the output layer. Thus, the motion speed is represented in the range $[-0.2, 1.2]$.

According to the error correction training algorithm, the SDNN repeats the training process until the mean squared error is sufficiently low or a specific number of iterations have been completed.

4. Experiment

4.1 Methods

To obtain measurements of motion speeds and surface EMGs, five adult male subjects were asked to execute six basic motions (flexion, extension, grasp, open, pronation, and supination) repeatedly at different speeds, each for 60 s. In total, 25–50 data samples were collected for each motion type and 25 of these data samples were selected randomly for use in the experiment. Next, we performed the following cross-validation (similar to a sixfold cross-validation): (1) for each motion type, 21 data samples including the fastest motion were selected to train the corresponding SDNN, whereas the remaining four data samples were used as test data; (2) the process was repeated by changing the training data samples, except the fastest motion.

4.2 Results

Figure 5 shows an example of the estimated motion speeds, which demonstrates that our proposed method could estimate the approximate motion speed. In addition, this method did not show a time lag between the measured and estimated motion speeds. Thus, we consider that the proposed method can be applied to EMG-based interfaces.

5. Discussion

To investigate the effect of the SDNN on the performance of this method, we compared the estimation accuracy with conventional function approximators: MLP and logistic regression (LR).

The MLP had one hidden layer (20 units) and it was trained by back-propagation learning. The training process was repeated until the mean squared error was sufficiently low or a specific number of iterations had been completed. The parameters of the LR model were determined using the steepest descent method.

The experimental conditions were same as those used in the previous experiment. To evaluate the estimation accuracy, the average correlation coefficients (CCs) between the actual

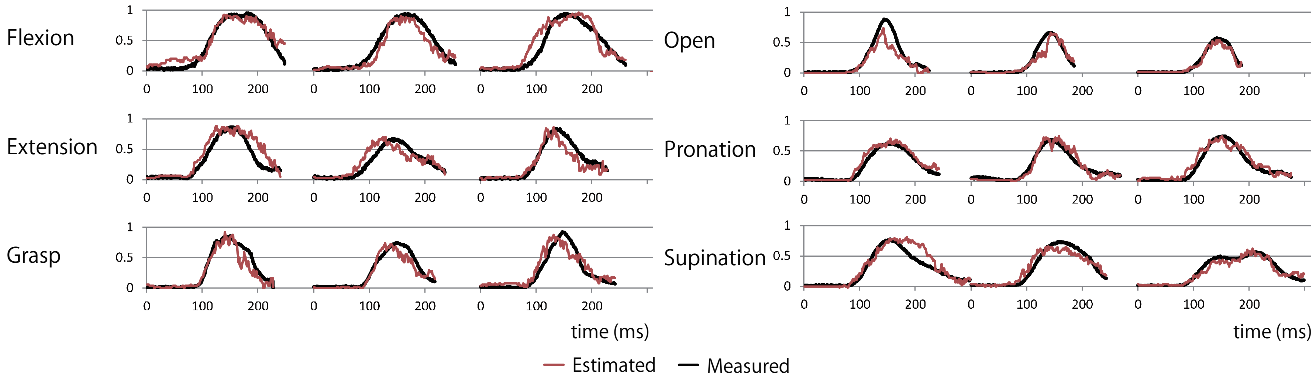


Figure 5: Examples of estimated motion speeds

Table 1: Comparison of different function approximators

(a) Test data

	Motion						
	F	E	G	O	P	S	Ave.
SDNN	0.90	0.88	0.85	0.89	0.90	0.91	0.89
MLP	0.84	0.82	0.69	0.81	0.83	0.83	0.80
LR	0.87	0.85	0.70	0.85	0.85	0.86	0.83

(b) Training data

	Motion						
	F	E	G	O	P	S	Ave.
SDNN	0.98	0.97	0.96	0.97	0.97	0.97	0.97
MLP	0.90	0.89	0.80	0.91	0.91	0.92	0.89
LR	0.91	0.90	0.80	0.92	0.90	0.91	0.89

F: Flexion E: Extension G: Grasp
O: Open P: Pronation S: Supination

and estimated motion speeds were calculated for each motion type.

The experimental results are shown in Table 1(a). The average CCs with the MLP and LR models were 0.80 and 0.83, respectively, which were 0.09 and 0.06 lower than that obtained with the SDNN (0.89). Furthermore, the average CCs with both the MLP and LR models were much lower for the grasp motion. These results indicate that the accuracy of the results obtained using the MLP and LR models were not suitable for practical use.

To understand the sources of the estimation accuracy differences, the learning performance was compared using the SDNN, MLP, and LR models. Table 1(b) shows the average CCs for the actual and estimated motion speeds using the training data samples, where the CCs obtained with SDNN were greater than 0.96 for all of the motions, whereas those obtained with the MLP and LR models were much lower, especially for the grasp motion. This suggests that the SDNN has a greater learning capacity than the MLP and LR models, and this difference caused the variation in the estimation accuracy.

6. Conclusion

In this study, we proposed a method for directly estimating the speed of multiple hand motions from multichannel surface EMGs using SDNNs. The experimental results showed that the proposed method could estimate the approximate motion speed and there was no time lag between the actual and estimated motion speeds.

Given its practical advantages, such as the lack of a requirement for adjustment sensors and the low computational costs, the proposed method may be suitable for application to EMG-based devices. Thus, combining this method with a practical motion classification method may facilitate the development of user-friendly EMG-based devices.

In future research, we aim to improve the estimation accuracy and to implement an EMG-based device using the proposed method.

Acknowledgment

This work was supported partly by JSPS KAKENHI Grant Numbers 22300079 and 24700593.

References

[1] M. Yoshikawa, M. Mikawa and K. Tanaka: Real-time hand motion classification using EMG signals with support vector machines, IEICE Trans. Inf. & Syst. (Japanese Edition), Vol. J92-D, No. 1, pp. 93–103, 2009.

[2] H. Kawata, F. Takana, A. Suemitsu and M. Morita: Practical surface EMG pattern classification by using a selective desensitization neural network, Neural Information Processing (Part II), Lecture Notes in Computer Science, Vol. 6444, pp. 42–49, 2010.

[3] K. Nonaka, F. Tanaka and M. Morita: Empirical comparison of feedforward neural networks on two-variable function approximation, IEICE Trans. Inf. & Syst. (Japanese Edition), Vol. J94-D-II, No. 12, pp. 2114–2125, 2011.