

## Research Article

# Estimation of Finger Joint Angles from sEMG Using a Neural Network Including Time Delay Factor and Recurrent Structure

**Masaaki Hioki and Haruhisa Kawasaki**

*Faculty of Engineering, Gifu University, 1-1 Yanagido, Gifu 501-1193, Japan*

Correspondence should be addressed to Masaaki Hioki, kabaduki@hotmail.com

Received 10 November 2011; Accepted 22 December 2011

Academic Editor: M. Pääsuke

Copyright © 2012 M. Hioki and H. Kawasaki. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Background.** The surface electromyogram (sEMG) is strongly related to human motion and is useful as a human interface in robotics and rehabilitation. **The purpose of this study was to establish a new system for estimating finger joint angles using few sEMG channels.** **Methods.** To deal with a dynamic system, the proposed method adopts time delay factors and a feedback stream into a neural network (NN) with 6 system parameters. The 2 target motion patterns were each tested with 5 subjects. 1000 combinations of system parameter sets were tested. **Results.** A system with only 4 channels can estimate angles with 7.1–11.8% root mean square (RMS) error, which is approximately the same level of accuracy achieved by other systems using 15 channels. **Conclusions.** The use of so few channels is a great advantage in an sEMG system because it provides a convenient interface system. This advantage is conferred by the proposed NN system.

## 1. Background

It is hoped that biological signals may be used as a new type of human-machine interface. The electroencephalogram (EEG) and surface electromyogram (sEMG) have been used in many studies. The EEG provides a great variety of information on humans and can be used as a brain-machine interface (BMI); applications currently being studied include interface systems for computers [1] and robot hands [2].

While the EEG examines internal human information (i.e., emotion, thinking), the sEMG is used to study human motion [3–6] because it is generated while voluntary muscle movements. It is highly useful as a machine interface, especially for prosthetics and robotic devices that assist rehabilitation. The human hand has many joints and performs important functions; therefore, injury to or loss of fingers is a serious problem. Prosthetic hands [7–12] and robotic devices that assist rehabilitation [13–26] are being developed in many institutes, and some prosthetic hands are available in the world market [7–9]. Robotic devices that assist rehabilitation are helpful in achieving an

early recovery for injured patients. Devices that target the upper limb (shoulder or elbow, without hand) are currently being researched [13–19], while other work focuses on hand and finger function [20–26]. While research on prosthetic hardware has achieved a certain level, the next step must focus on how to control these devices. One solution is to use sEMG, which in fact has already been applied to controlling a device that assists rehabilitation [19, 27] and to estimating the degree of recovery [28].

In research that targets estimating hand state from sEMG, many studies have focused on discriminating hand patterns [29–34]; that is, they deal with discrete clustering problems using sEMG as input information. The resultant systems can recognize hand shape (e.g., “open”, “fist”, or “peace sign”), and the main aim now is to apply such systems to the problem of controlling a prosthetic hand. Some studies have achieved high estimating accuracy; however, this type of system can estimate only predetermined patterns. Thus, if such a system was trained to recognize only “open hand” and “fist”, it will be unable to recognize “peace sign.” A system that estimates continuous finger joint angles, however, would

be able to recognize every hand state and therefore has high versatility. Some studies have begun to examine the estimation of finger joint angle [35–37].

Shrirao et al. [35] establish an index finger joint angle estimation system that uses neural network (NN) committees and that can interpret 3 flexion-extension speeds (low-, middle- and high-speed flexion and extension). It uses 1 sEMG channel at the extensor digitorum superficialis muscle. Kitamura et al. [36] report on an estimation system for base flexion/extension of the joint of the index finger. Their system adopts an analysis of the autoregressive relationship. It applies 4 channels at forearm, but no strict attach locations are determined. Their target is only a single joint of a single finger; however, since human hand skill involves multiple fingers and joints, this system is inadequate as a human-machine interface for understanding hand state.

Smith et al. [37] estimate multiple finger joint angles from many sEMGs. They used an NN to estimate the metacarpophalangeal (MP) joint angles of five fingers during the individual movement of each joint. Their system uses 15 channels with indefinite attach locations. They report an estimating precision of ~11%. However, certain estimated joint angles vary even if the joints are not moving. For example, even when the thumb joint is not moving, the joint angle is estimated as a moving joint. The reason for this is that the dynamics of the estimation system cannot be trained because the system uses a simple feedforward NN. Furthermore, in order to apply sEMG to a prosthetic hand system or a device to assist rehabilitation, the number of sEMG electrodes should be small. Fifteen channels are too many for this purpose.

Our goal was to establish a new type of finger joint angle estimation system, specifically, one that can effectively interpret a dynamic system using very few sEMG electrodes. Such a system would be useful not only for operating robot hands but also for controlling a prosthetic hand by teleoperation.

In the present study, a perceptron with three layers (one type of NN) was adopted as a multi-input and multioutput system that can estimate finger joint angles. In order to deal with the complicated relationship between sEMG and multifinger joint angles, a recurrent structure using a feedback stream and time delay factors was integrated into the perceptron system. In this way, the input vector into the perceptron includes information about current and past sEMG and feedback, allowing the estimate system to train the dynamics of the target system. Note that while the NN has a recurrent structure, the network can train a method for a feedforward network, but not for a recurrent one. This system requires many system parameters, and the present paper shows the influence that such parameters exert on the estimated results.

The target finger motion groups in the present study were the following:

- (1) individual finger flexion and extension,
- (2) finger flexion and extension both individually and together.

After training the network with the appropriate data set, finger joint angles were estimated using an untrained data

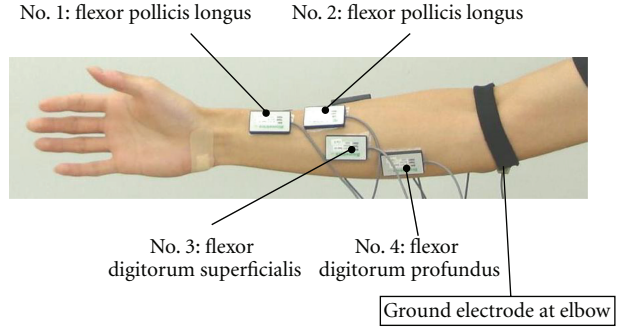


FIGURE 1: Location of the sEMG measuring electrodes. A total of 4 electrodes are used in this system. Many muscles related to finger motion are located in the forearm, and the muscles on the flexional side of the forearm in particular control finger flexion.

TABLE 1: Target muscles and their main functions. On the flexion side of the forearm, there are certain muscles that relate to finger flexion. Note particularly that the muscles of the thumb are different from those of the other fingers so electrodes 1 and 2 specifically target the flexor pollicis longus to obtain information about the thumb.

Electrode index	Target muscle	Muscle's main function
1	Flexor pollicis longus	Thumb PIP and DIP flexion
2	Flexor pollicis longus	Thumb PIP and DIP flexion
3	Flexor digitorum superficialis	2nd 5th finger PIP flexion
4	Flexor digitorum profundus	2nd 5th finger DIP flexion

set. The estimation system was then evaluated by comparing the results with measured finger joint angles.

## 2. Methods

### 2.1. Data Acquisition

**2.1.1. sEMG.** The sEMG signals were measured at the right forearm of each subject at the 4 locations shown in Figure 1 and Table 1. In order to configure the sEMG system with the lowest possible number of electrodes, all electrodes in the proposed system are placed on the flexor side of the arm and none are placed on the extensor side. Therefore, for the purposes of the present study, we focused on the flexor muscles, using 2 channels for thumb motion and 2 for the 4 fingers (a total of 4 channels).

The locations were determined by palpation for each subject such that channels 1 and 2 would record thumb flexion, and channels 3 and 4 would record finger flexion. Because displacement of the electrode affects the precision of the estimate [3], all data must be remeasured if for any reason the electrodes must be reattached.

The sEMG sampling was performed at above 4 ch, 1 ms sampling frequency, 12 bit resolution of the AD converter,

and  $-5 \sim +5$  V dynamic range with the appropriate equipment (sEMG amplifier system: PTS-137, DKH Co., Ltd., Tokyo, Japan).

**2.1.2. Finger Joint Angles.** The present system adopts a three-layered perceptron whose core is an NN that can train the desired input-output relation; a teacher signal vector is necessary to train the NN. Because the system aims to estimate finger joint angles, the teacher vector should be extracted from the data glove measurements. Finger joint angles were sampled at 30 Hz using a CyberGlove (Immersion Co., Ltd., San Jose, California, USA) on the right hand at the time of sEMG measurement. The measured finger joints were the metacarpophalangeal (MP) joint, the proximal interphalangeal (PIP) joint, the distal interphalangeal (DIP) joint, and the abduction-adduction (Abd) joint. Therefore, the total number of measured finger joints per subject was 20.

## 2.2. Signal Processing

**2.2.1. Feature Vector from sEMG.** Measured raw sEMG data was processed using the segmentation technique [4] after processing with a band-pass filter (first order, pass band: 10–350 Hz). This signal processing is useful and powerful for extracting the feature vector from raw sEMG data. The present system adopted the following segmentation parameters: segment length, 1024; shift length, 32. Every 32 ms, waveform length [3, 4] (WL) was adopted as the feature value for each segmentation of raw sEMG. This type of feature value includes both a time domain feature and a frequency domain. The raw sEMG vector had 4 elements, as did the feature vector.

After the feature vector was extracted, it was normalized in the range between 0 and 1. The parameters necessary for normalizing (maximum and minimum values) were obtained and calculated from advanced measuring data, which takes the relaxed state of the muscle as the minimum value and the contracted state as the maximum value.

**2.2.2. Teacher Vector from Finger Joint Angles.** After the measurement of the 20 finger joint angles is available, however, there will be redundant information. For example, because of the structure of the human muscle-tendon-joint system, it is difficult to flex only the DIP joint during single-finger flexion. Thus, the number of joints tested can be reduced to streamline the estimation system. With respect to single-finger flexion, the PIP joint has a large range of motion and it is possible to flex only the PIP without also flexing the MP or DIP. And PIP can move in the range of around 120 degrees, while MP can in around 90 degrees. It can therefore be said that the PIP joint shows finger flexion. In processing for joints, we therefore extracted 5 finger joints as the teacher vector, that is, the dimension is reduced from 20 to 5.

Similarly, the parameters necessary for normalization were obtained and calculated from advanced measurement data. The value of the teacher vector indicates the flexion rate of the joint (0, full extension; 1, full flexion). In other

words, this value also means flexion degree, so this can be represented with unit of percentage.

**2.3. Estimation System with an NN.** The present estimation system for finger joint angles was developed from feature values obtained from sEMG by integration with a three-layered perceptron. This is a feedforward type of NN with the following useful characteristics that are effective in estimating finger joint angles from sEMG.

- (i) The network does not need a strict system model. With training, it can approximate the input-output nonlinear function for training data. This is important for dealing with personal differences.
- (ii) It is possible to apply this network to a multi-input multioutput system. Modifying the dimensions of the input or output vectors is therefore quite easy.
- (iii) By using the sigmoid function as a unit output function in the output layer of the NN, it is possible to limit the output values from the network to (0, 1), which is an easy range to relate to finger joint angles. Additionally, the limitation of output from the network system can also contribute to system stability.

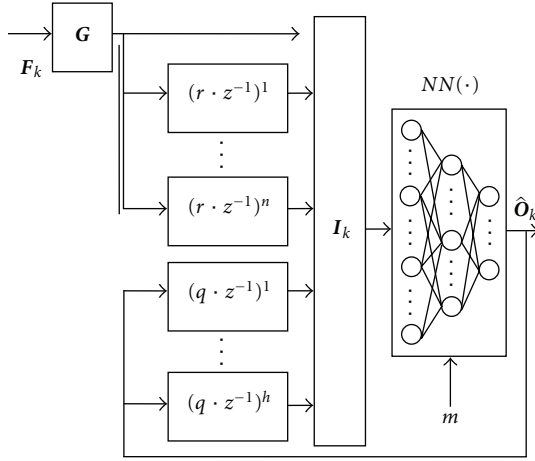
Furthermore, in order to train the dynamics of the estimation system, we introduced to the three-layered perceptron a recurrent structure using a feedback stream of past output signals from the network and a time-delay factor to input to the network. This makes it possible for the estimation system to train the system dynamics. The estimation system is shown in Figure 2. In this system, the time delay parameters for feature vector  $n$  and feedback vector  $h$  indicate the extent to which previous feature vectors and precious feedback vectors, respectively, are considered.

In this figure, NN ( $\cdot$ ) is a three-layered perceptron. This type of NN can train any input-output relation if the number of the middle layer unit  $m$  is adequate. The present system adopts the standard sigmoid function as the activating function of each unit in the NN, and the steepest descent and backpropagation methods as the training method. This training method changes the coefficients of the internal unit and it can be shown simply as follows:

$$\Delta w = \alpha \cdot \delta \cdot \mathbf{x}, \quad (1)$$

where  $\Delta w$  is the amount of change of a certain coefficient of the unit,  $\alpha$  is the training rate and adequate small value,  $\delta$  is a backpropagation term, and  $\mathbf{x}$  is the input into this unit. Under this method, a coefficient that takes large input values will change largely. During network training, modifying the value of the input vector into the network is also important for highly precise functioning.

The gain  $G$  is a constant value matrix whose value is 1 or more than 1. The  $G$  makes the value of the feature vector components rather than the feedback vector components large because the NN trains the system depending on the sEMG information rather than on feedback information. The forgetting rates  $r$  for the feature vectors and  $q$  for



$F_k$ : feature vector extracted from sEMG measurements. Range is (0, 1)

$\hat{O}_k$ : estimated vector. Range is (0, 1)

$I_k$ : estimated vector. Range is (0, 1)

$z^{-i}$ :  $i$ -step time delay factor

$n$ : time delay parameter for feature vector

$h$ : time delay parameter for feedback vector

$m$ : number of middle layer units of the NN

$G$ : gain for feature vector

$r$ : forgetting rate for feature vector

$q$ : forgetting rate for feedback vector

FIGURE 2: Structure of the estimating system. Subscript  $k$  indicates  $k$ th-step data. The main part of this estimating system is the neural network (NN). The input vector into the NN has feedforward and feedback components.

the feedback vectors have a value of 1 or less than 1. These parameters make the estimation system weigh newer information as more important than older information.

The system can be shown as follows (without gain or forgetting rates):

$$\begin{aligned} \hat{O}_k &= NN(I_k) \\ &= NN(F_k, F_{k-1}, \dots, F_{k-n}, O_{k-1}, \dots, O_{k-h}). \end{aligned} \quad (2)$$

The input vector into the NN is composed of the following factors:

- (1) newest feature vector:  $F_k$ ,
- (2) past feature vectors:  $F_{k-1}, \dots, F_{k-n}$ ,
- (3) past output vectors from the NN, that is, feedback components:  $O_{k-1}, \dots, O_{k-h}$ .

Thus, the estimation system can train the target system dynamics. If parameters  $n$  and  $h$  are both zero, the estimation

system becomes ordinary [22], and it is clear that the proposed system is the expansion of the ordinary

$$\hat{O}_k = NN(I_k) = NN(F_k), \quad n = 0, \quad h = 0. \quad (3)$$

As described above and shown in Figure 2, the number of units at the input layer of NN will change depending on the values of  $n$  and  $h$  because the dimensions of  $F_k$  and  $O_k$  are constant at 4 and 5, respectively. The number of units at the input layer NumIn can be represented as follows:

$$\text{NumIn} = 4 + 4n + 5h. \quad (4)$$

On the right-hand side of this equation, the first term shows the dimension of the newest feature vector ( $F_k$ ), the second term shows the dimensions of past feature vectors ( $F_{k-1}, \dots, F_{k-n}$ ) and the third term shows the dimensions of feedback components ( $O_{k-1}, \dots, O_{k-h}$ ). For example, if  $n$  is 0 and  $h$  are both 0 (i.e., an ordinary NN system), then NumIn is 4 (NumIn =  $4 + 4 * 0 + 5 * 0$ ). Similarly, if  $n$  is 2 and  $h$  is 6, then NumIn is 42 ( $4 + 4 * 2 + 5 * 6$ ).

About the other information of NN structure, the number of middle layer is dealt with as system parameter  $m$  and the number of output layer is five (constant) because the proposal system uses one unit of output layer for estimating one joint.

**2.4. Training Phase and Estimation Phase.** The execution of this estimation system is divided into 2 phases: the training phase and the estimation phase (Figure 3). The system parameters show the parameters discussed in the previous section ( $n, h, G$ , etc.).

**2.4.1. Training Phase.** The primary purpose of the training phase is to obtain optimized NN coefficient data  ${}_oW$  that can represent the input-output mapping from sEMG data to the finger joint angle dynamics system with high precision (Figure 3(a)). During this phase, the estimation system uses the feature and teacher vector sets for training, but does not use feedback.

The training is complete when it reaches the number of the maximum training epoch. After training, the coefficient data should be optimal. The initial values of the elements contributed by the time delay factor are 0, and the initial values of the NN coefficients are initialized by random number.

**2.4.2. Estimation Phase.** During the estimation phase, the estimated angle vector  $\hat{O}_k$  is calculated from feature vectors that are not used for training or the optimized NN coefficient  ${}_oW$  (Figure 3(b)). The system must use the same system parameter set as that used in the training phase.

The NN is not trained and calculates the estimated vector from the input vector. While system training requires a considerable amount of time, estimation is rapid and can be realized in real-time processing.

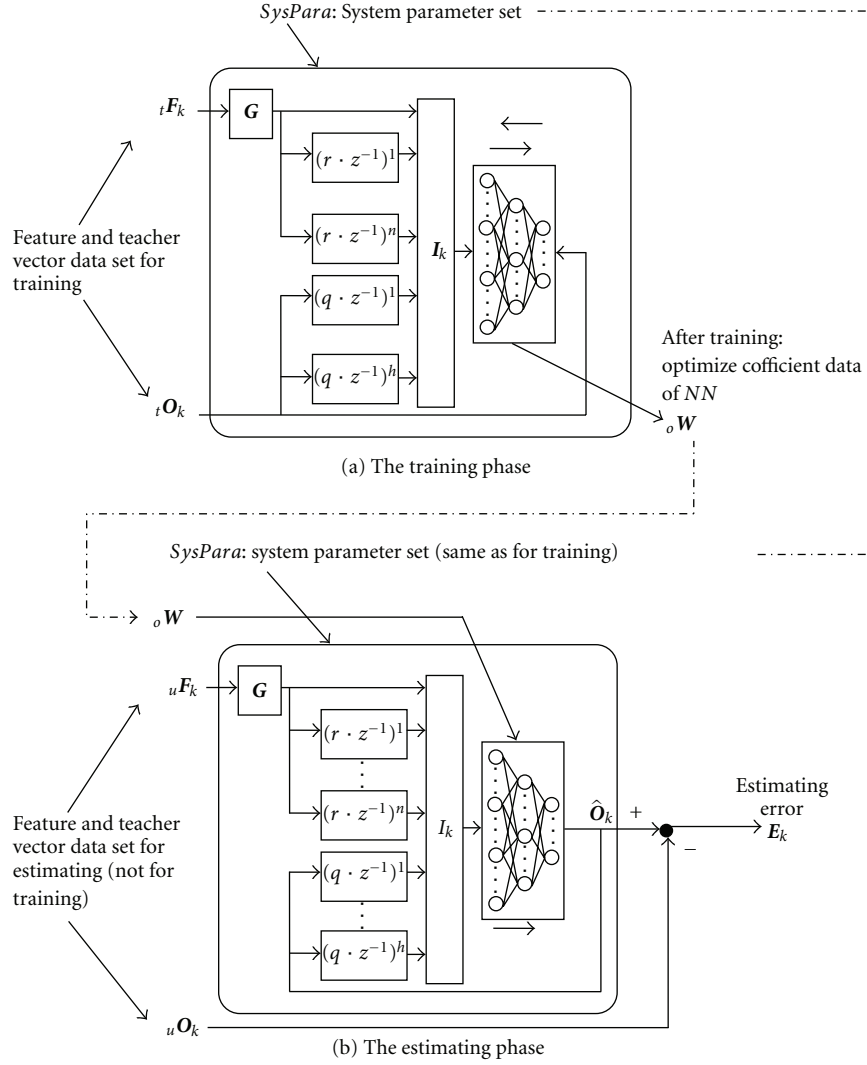


FIGURE 3: The estimating system at the training phase and estimating phase. System parameter *Syspara* represents the parameters set in Figure 2. The main purpose of the training phase is to optimize neural network coefficient data  ${}_oW$  represents high-precision input-output mapping (from sEMG to the finger joint angle dynamics system). During the estimating phase, the estimated angle vector  $\hat{O}_k$  is calculated from feature vectors that are not used for training or optimizing  ${}_oW$ .

### 3. Experiment

**3.1. Outline.** Through the experiment, the effectiveness of the estimating system and the contribution of the system parameters to the estimation result will be shown. Although all training and estimation were done offline in this experiment, the estimation can be performed in real time. This experiment was carried out on 5 able-bodied subjects (all are male, age is in 21–26, height is in 165–172 cm, weight is in 50–65 kg). This experiment was based on their approval after informed consent with Gifu University ethical committee.

**3.1.1. Target Finger Exercise.** The sEMG data reflect voluntary muscle contractions and strongly relate to human body motion. At the forearm in particular, the relationship between muscles and their performance is very complex. For

example, the flexor digitorum profundus contributes to the DIP flexion of the 2nd–5th fingers. Note that muscles and joints may not necessarily have a one-to-one relationship. In the present study, the following types of motion set are shown.

Motion Set 1: finger individual flexion and extension;

Motion Set 2: finger flexion and extension individually and together.

Motion Set 1 reveals the independence of finger motion with this estimation system. Single-finger flexion and extension are basic finger motions. Motion Set 2 includes the complexities of both single and multifinger motion. In daily life both single-finger and multifinger motions are important. Sample graphs are shown in Figure 4.



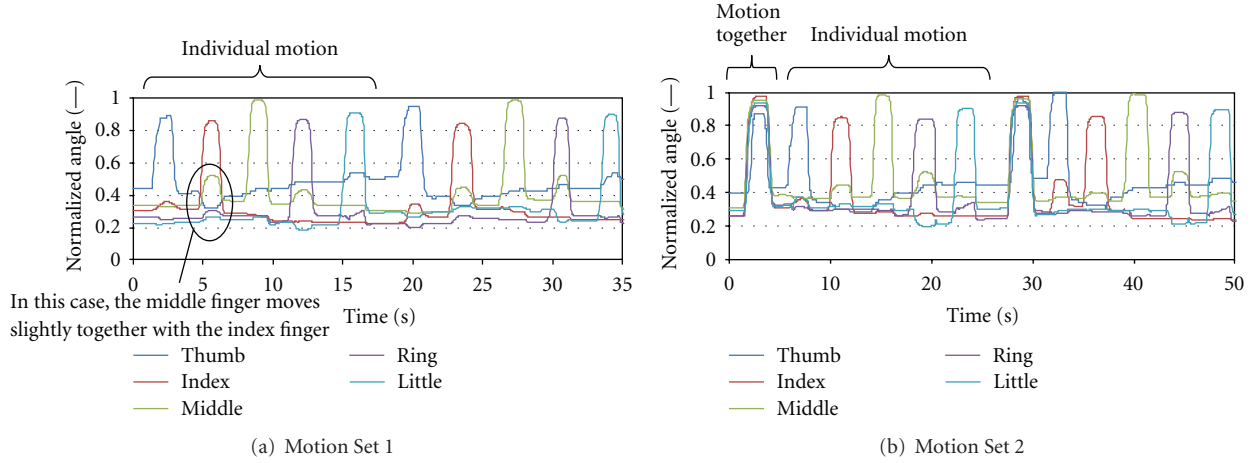


FIGURE 4: Examples of finger joint angles during each motion set. The subjects were instructed to move their fingers naturally; therefore more than one finger may move during finger motion. Joint velocities are not defined.

TABLE 2: Available values for system parameters. There are 8 values available for each parameter. Parameters  $n$  and  $h$  are integers with a value of 0 or more.  $G$  is a decimal number with a value of 1.0 or more, and  $r$  and  $q$  are also decimal numbers with a value of 1.0 or less.

System parameter	Available value
$n$	0, 1, 2, 3, 4, 5, 6, 7
$h$	0, 1, 2, 3, 4, 5, 6, 7
$m$	10, 20, 30, 40, 50, 60, 70, 80
$G$	1.0, 2.5, 5.0, 7.5, 10.0, 12.5, 15.0, 20.0
$r$	0.001, 0.05, 0.075, 0.1, 0.25, 0.5, 0.75, 1.0
$q$	0.001, 0.05, 0.075, 0.1, 0.25, 0.5, 0.75, 1.0

**3.1.2. Combination of System Parameters.** The proposed estimation system requires 6 system parameters. These parameters are independent of each other, and the total number of combinations is enormous because each parameter may have any value.

In order to limit the number of combinations of system parameters, the available values for each parameter were predetermined (Table 2). There are 8 values available for each parameter. Parameters  $n$ ,  $h$ , and  $m$  are integer values, and  $r$ ,  $q$ , and  $G$  are decimal values. These available values were determined by trial and error.

With this limitation on allowable values, the number of possible combinations is 262,144 ( $=8^6$ ). Because this number is too large to analyze, we adopted the regression method with random numbers.

**3.2. Experimental Procedure.** The experimental procedure was divided into the following phases.

- (i) Data acquisition: sEMG data set and finger joint angles for extracting normalizing parameters, training and estimating are needed in this system.

- (ii) Determination of system parameters: the 6 system parameter values are determined by random number.
- (iii) Training: the estimating system with the determined system parameters is trained with a data set.
- (iv) Estimating: after training, the system estimates finger joint angles with data not used for training (i.e., unknown data). The results and actual data from the data glove are compared and system performance is estimated.

In this experiment, phases (ii)–(iv) were repeated 1,000 times for each subject and motion set. Note that because new random number is generated at every parameter decision, the selected system parameters are not always same between the tasks and among the subjects. Since we had 5 subjects and 2 motion sets, the total number of trials (phases (ii)–(iv)) was 10,000.

**3.2.1. Data Acquisition.** In order to normalize the feature vectors extracted from sEMG and the teacher vector from the data glove, a data set for normalizing is needed. This data set is obtained from each subject. To normalize the sEMG information, it is necessary to perform sEMG during both relaxation and contraction of the muscles. Similarly, to normalize finger joint angles, the data obtained during full flexion and extension of each joint is necessary.

With respect to data sets for training and estimating, the measured terms are shown in Table 3.

**3.2.2. System Training for Each Subjects and Motion Set Individually.** First, the set of system parameters was determined by random number. Second, one estimation system trained with the feature-teacher data set and the set of system parameters until training epoch 200 was reached. This value of epoch limitation was determined by preexperiments, of which an example is shown in Figure 5. Almost all training trials had finished by the 50th epoch. Therefore, 200 epochs can be considered a sufficient value for this system.

TABLE 3: Measured data sets. This experiment used data sets from 5 subjects.

Subject	Motion set	Time [s]	
		For training	For estimating (not for training)
A	1	56	39
	2	91	53
B	1	63	25
	2	79	31
C	1	77	32
	2	86	33
D	1	65	26
	2	79	33
E	1	34	13
	2	40	17

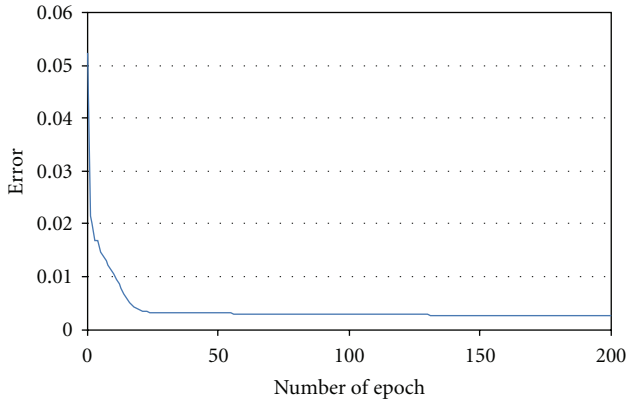


FIGURE 5: An example of the training process. While this graph shows only a single trial ( $n = 6$ ,  $h = 0$ ,  $m = 20$ ,  $G = 15.0$ ,  $r = 1.0$ ,  $q = 1.0$ ), almost all trials finished training by the 50th epoch.

This procedure was repeated for each subject and motion set, and 1000 combination parameter sets were tested. Global suboptimal parameter sets were then obtained using the above random search.

**3.2.3. Estimation with the Trained System and Evaluation of Results.** The 1000 parameter sets were tested with the estimation system for each subject and motion set. For each estimation system, estimation errors were calculated with data sets for estimation that were not used in training. The results were evaluated using the root means square (RMS) of the estimation error.

## 4. Results

**4.1. Best Performance Parameters Sets.** Table 4 shows the results of the above experiments. For each subject and motion set, 1000 trials were performed. In 1000 trials, the system parameters set with the smallest RMS of estimate

error can be considered the optimal solutions. Examples of the estimated graphs for Subject A are shown in Figures 6 and 7.

**4.2. Local Search Near the Optimal Solution.** On the assumption that the above random search can find a global optimal solution, system result behaviors near the solution were examined in greater detail. Experiments were performed in which 1 parameter varied while the remaining 5 remained fixed (the set of system parameters consists of a total of 6 parameters).

Figures 8 and 9 show the system behavior near the optimal solution for Subject A's motion sets. Note that the units of the vertical axis of each figure show the RMS error and are in different ranges. For example, in Figure 8, the range of (b) is from 7.0 to 8.2, while the range of (c) is from 0.0 to 50.0.

## 5. Discussion

For each trial results, finger joint angles can be estimated by a trained system with appropriate value system parameters. These parameters sets are determined by the above random search method, which functions as a global search in multidimensional space. Note again that the range of the vertical axis of each figure (Figures 8 and 9) is different, which means that the contribution of the parameter to the accuracy of the estimation is also different.

Parameters  $n$  and  $h$  relate to the number of input vector elements (or dimensions). Because the number of output vector  $\hat{\mathbf{O}}_k$  elements remains constant at 5,  $n$  and  $h$  can change the ratio of input items to output items. As a result, these parameters should have values that are neither very small nor very large. Parameter  $h$  in particular should be determined more carefully than others, as should Parameter  $n$  (Table 4, Figures 8(a), 8(b), 9(a) and 9(b)). Figures 8(b) and 9(b) show the estimation accuracy for each value of  $h$ . It appears that feedback components may make more of an impact on estimation than time-delayed sEMG components. In either case, the combination of  $n$  and  $h$  must be determined carefully (Table 4).

Parameter  $m$  represents the level of complexity at which NN can train input-output mapping. Generally, an NN with a large  $m$  that can train a complex input-output relationship is also able to train an unexpected relationship. An NN system with a small  $m$  cannot carry out high-precision estimation. Based on the data shown in Table 4 and Figures 8(c) and 9(c), it is clear that  $m$  should be 20 or more to achieve a highly accurate estimation, and that there is no significant difference among large  $m$  values. In the local search cases shown in Figures 8 and 9, the dimensions of the input vector into the NN and of the output vector remain constant. In other words, in the local search for Parameter  $m$ , the complexity of the output-input relationship is constant as long as the value of  $m$  is adequate. As described above, too large a value of  $m$  is not desirable from the view point of avoiding training unexpected relationships and the cost of calculation of training. The present results suggest the use of

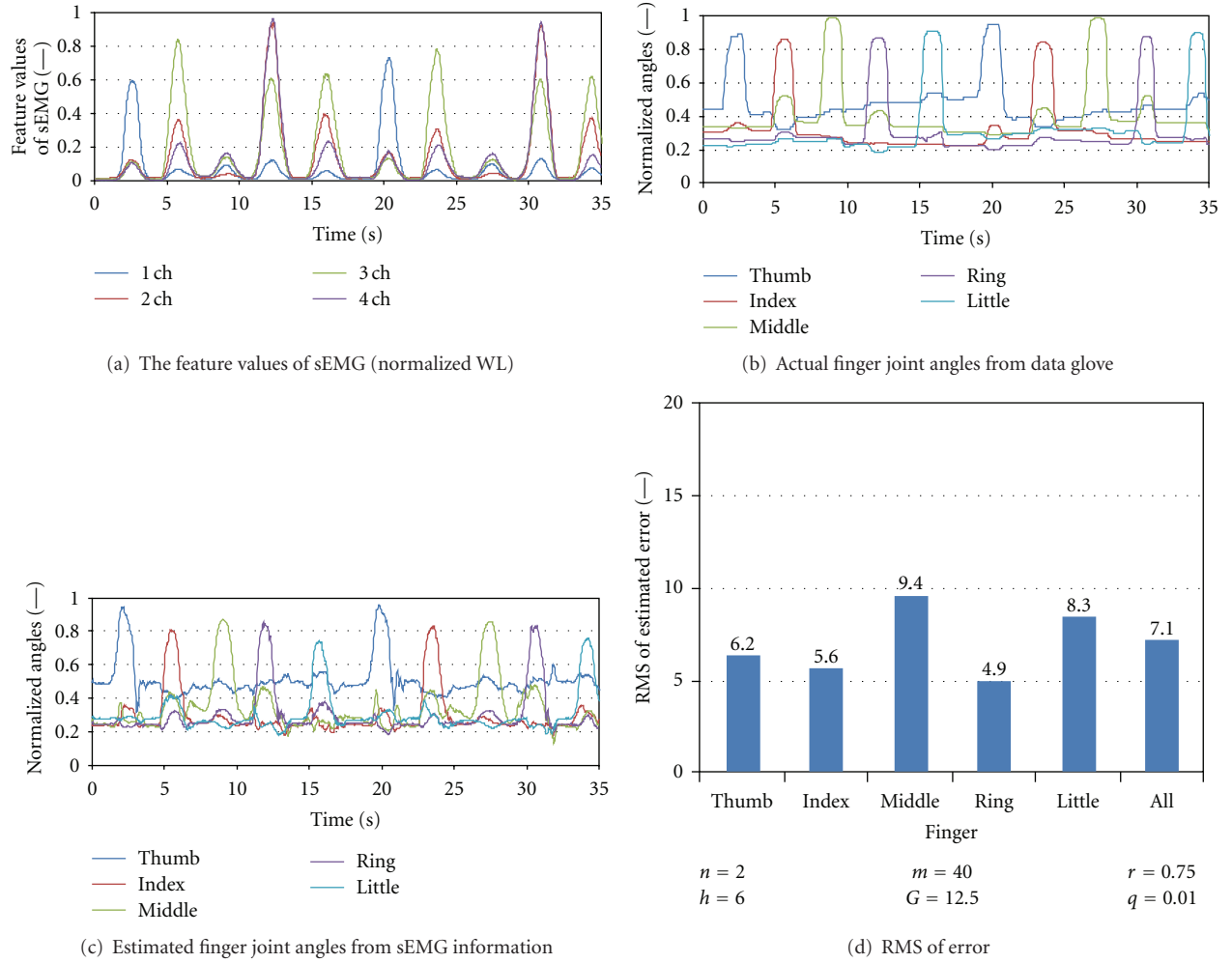


FIGURE 6: Results for Subject A and Motion Set 1: (a) feature vectors extracted from sEMG, (b) actual joint data from the data glove, (c) estimated results, and (d) RMS of estimated errors.

TABLE 4: Best performance parameter sets. These parameter sets are consistent with the best performance out of 1000 sets. These sets assume the quasiglobal optimal solutions obtained by random search. As examples, the result estimate graphs of Subject A are shown in Figures 6 and 7.

Subjects	Data		System parameters					
	Motion set	RMS of estimating error %	$n$	$h$	$m$	$G$	$r$	$q$
A	1	7.1	2	6	40	12.5	0.750	0.010
	2	10.3	4	3	30	12.5	0.750	0.050
B	1	10.9	6	4	50	15.0	1.000	0.100
	2	10.5	7	4	60	12.5	0.750	0.010
C	1	11.4	3	5	60	7.5	0.500	0.250
	2	13.2	3	0	70	10.0	0.500	0.100
D	1	7.4	6	0	20	15.0	1.000	0.100
	2	14.2	3	4	80	12.5	1.000	0.075
E	1	11.8	6	0	20	20.0	0.750	0.075
	2	14.4	6	5	60	7.5	1.000	0.010



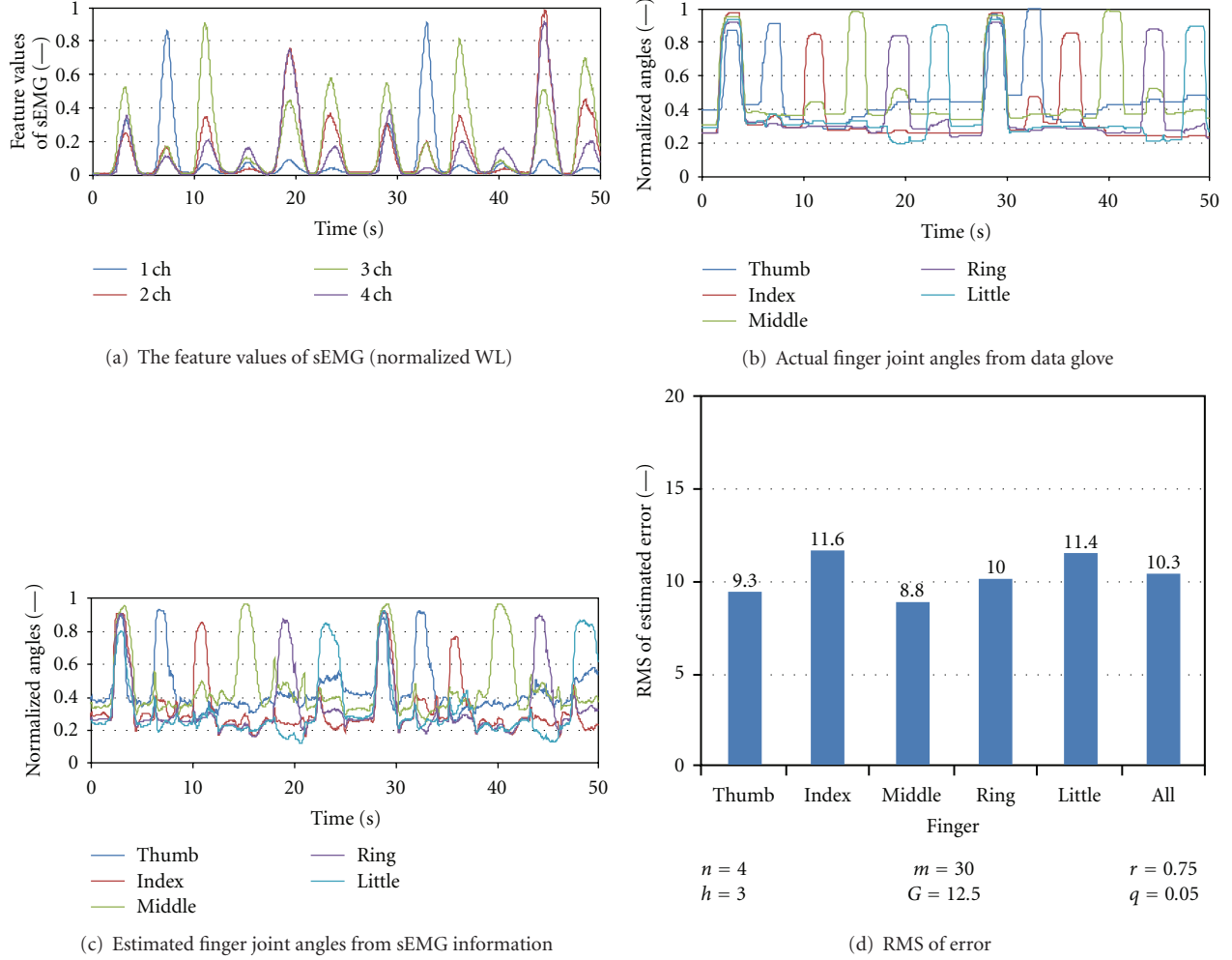


FIGURE 7: Results for Subject A and Motion Set 2: (a) feature vectors extracted from sEMG, (b) actual joint data from the data glove, (c) estimated results, and (d) RMS of estimated errors.

a sufficiently small value of  $m$  (in this experiments, the value is 20 from Figures 8 and 9).

Parameters  $G$ ,  $r$ , and  $q$  are the gain constant for the feature vector, the forgetting rate for older feature vectors, and the forgetting rate for feedback vectors, respectively. These parameters determine the relative amplitude of the input vector elements and therefore relate to training the NN.

According to the data shown in Table 4, and Figures 8 and 9, the larger the value of  $G$ , the more precise the estimation. Thus, based on the function of this parameter presented in previous section, the components that relate to the sEMG are more important than those relating to feedback. This is a natural result because the sEMG relates directly to muscle contraction and finger joints are moved by muscles.

With respect to Parameter  $r$ , an estimation system in which  $r$  is near 1.0 can estimate with great precision. That is, among the sEMG components in the input vector into NN, both newer and older data are helpful for highly accurate estimation. With respect to Parameter  $q$ , a system in which  $q$  has a small value can estimate with great precision. Older

feedback components are not more important than newer elements in this case. Comparing these 2 parameters, the contribution of  $q$  to the estimation result is larger than that of  $r$ , and parameter  $q$  should be determined carefully.

As discussed in the previous section and (1), each unit training depend on the input value of the unit with training method of the NN that is used in the present system. Additionally, the coefficients of the NN change in small steps at every training step. Thus, we believe that Parameters  $G$ ,  $r$ , and  $q$  may be used to modify NN training appropriately.

The system described by Smith et al. [37] targets finger flexion/extension individually with an estimating precision of  $\sim 11\%$  using 15 sEMG channels. In contrast, the present system can train a dynamical system at a similar level of precision (7.1 $\sim$ 11.8%) in the case of Motion Set 1 (which is equivalent to Smith's motion set), but using only 4 sEMG channels, due to the proposed NN system structure with time delay, feedback, and some coefficients. This is a large advantage with respect to the sEMG system and provides a convenient interface system. With this structure, the system

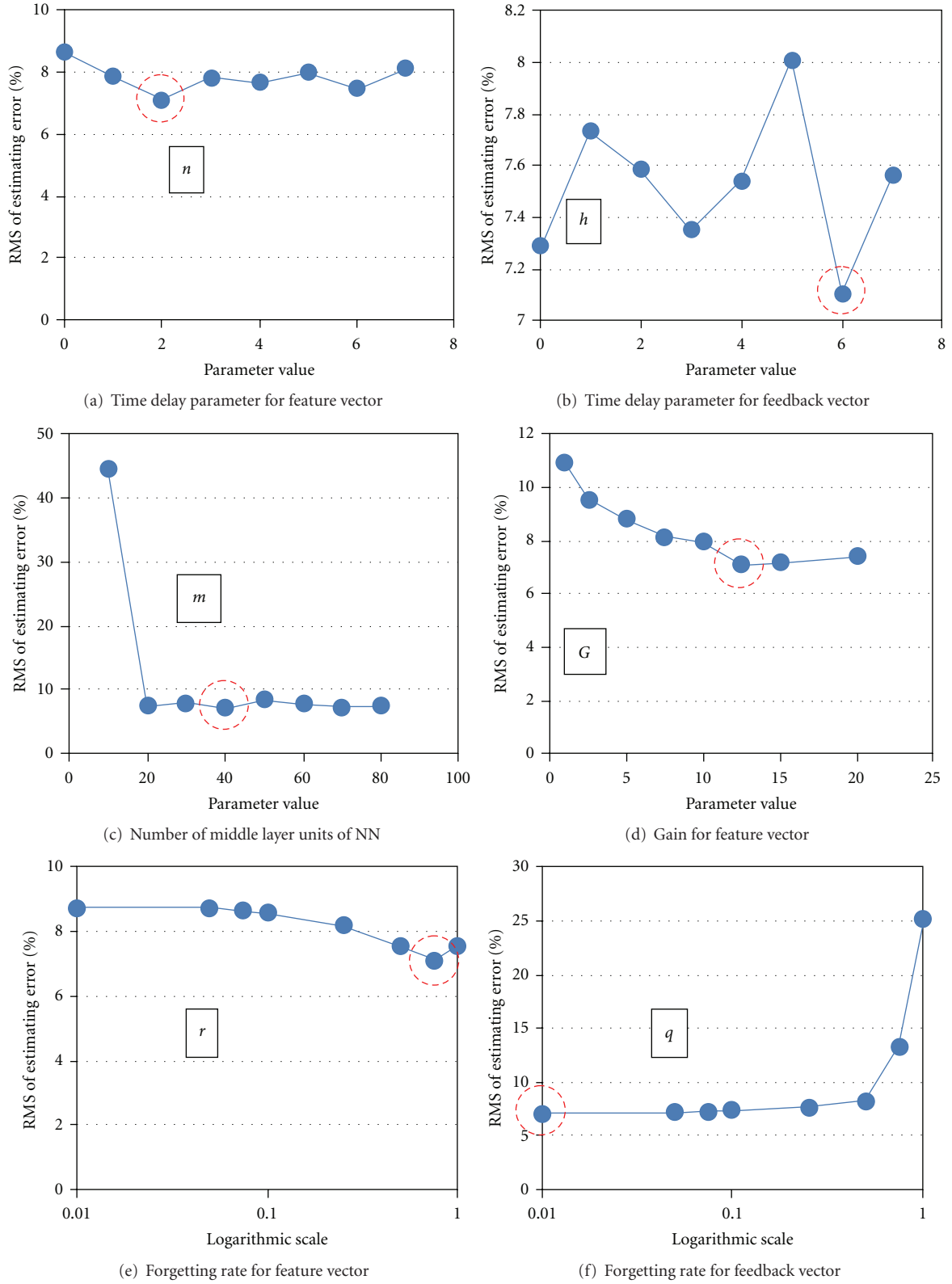


FIGURE 8: The result error behavior around the optimal solution for Subject A and Motion Set 1. The set of suboptimal system parameters is as follows:  $n = 2$ ,  $h = 6$ ,  $m = 40$ ,  $G = 12.5$ ,  $r = 0.75$ ,  $q = 0.01$ ; and the RMS is 7.1%. Red dashed circles indicate suboptimal parameters. \*The horizontal axis of all figures shows parameter value. \*\* The horizontal axes of (e) and (f) are in logarithmic scale.

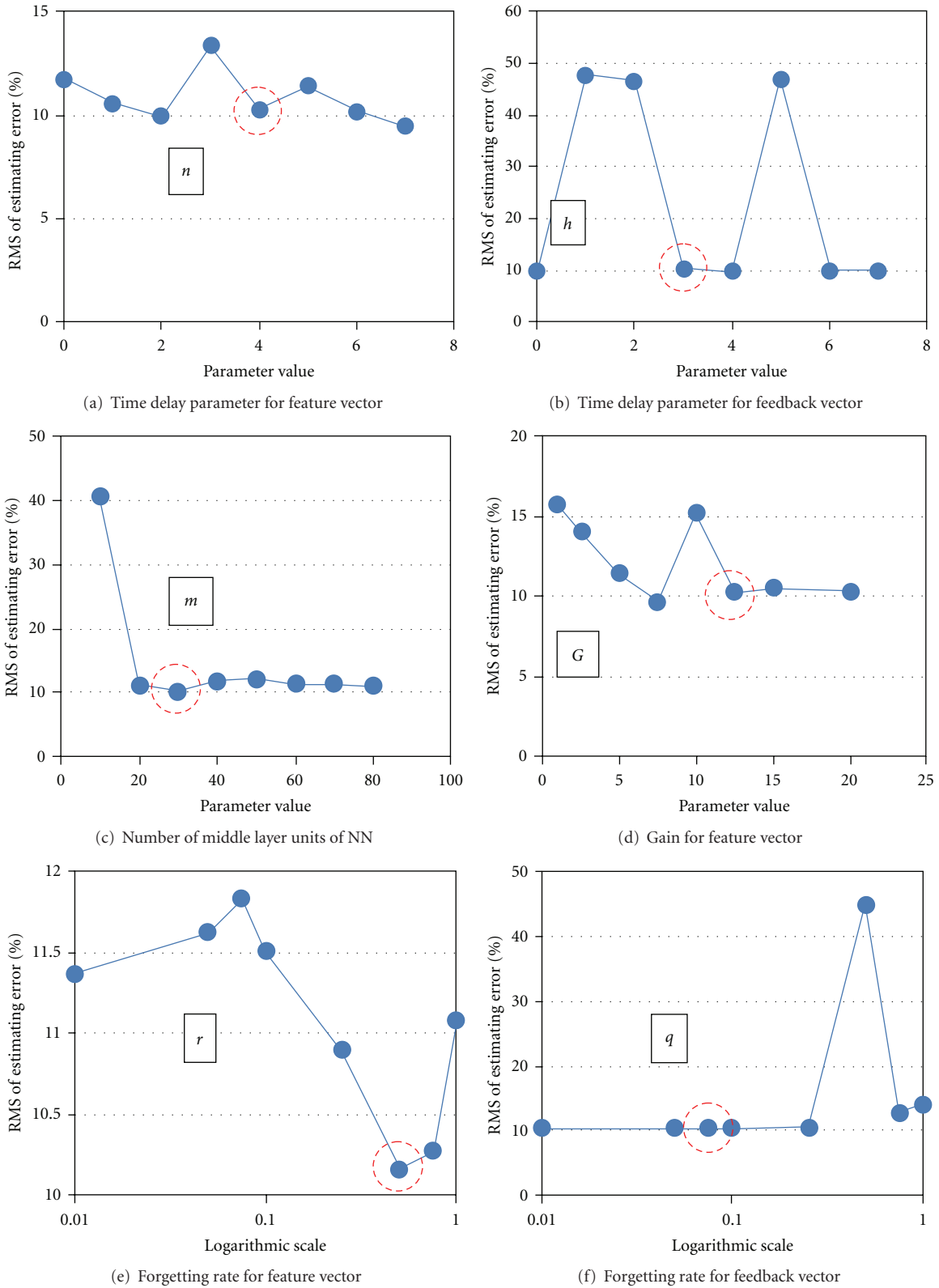


FIGURE 9: The result error behavior around the optimal solution for Subject A and Motion Set 2. The set of suboptimal system parameters is as follows:  $n = 4$ ,  $h = 3$ ,  $m = 30$ ,  $G = 12.5$ ,  $r = 0.75$ ,  $q = 0.05$ ; and the RMS is 10.3%. Red dashed circles indicate suboptimal parameters. Note that range of vertical axes is difference with Figure 8. \*The horizontal axis of all figures shows parameter value. \*\*The horizontal axes of (e) and (f) are in logarithmic scale.

can train the essential sEMG-finger joint angle system and the system can thus accurately estimate angles with little sEMG information.

## 6. Conclusions

This paper presents a finger joint angle estimation system based on sEMG. The type of information provided by sEMG is very useful not only for creating and using robot hands but also as a rehabilitation interface; however, the relationship between sEMG and finger joint angles is complex. In order to train the sEMG-finger joint angle dynamic system, the estimation system adopts time delay and a feedback stream with specific system parameters.

We tested the proposed system with 5 subjects using 2 types of finger motion sets for each subject: (1) individual flexion and extension of all fingers and (2) flexion and extension of all fingers individually and together. For each subject and motion set, 1000 system parameter sets were used to establish the estimation system by random search and to set the parameters such that they produce results with the smallest error (i.e., the optimal solution). Furthermore, in order to analyze solution behavior around an optimal point, local analysis was carried out for single subjects.

From the results of the above, certain characteristics of the 6 system parameters were defined. While time delay parameters  $n$  and  $h$  must be selected for the subject or motion set, the value of  $m$  is sufficient at a certain level (in the present experimental results, 20). Gain parameters  $G$ ,  $r$ , and  $q$  should be large, large, and small, respectively.

The proposal system achieved the continuous multifinger joint angle estimation from sEMG. The angle estimation system has high versatility rather than pattern recognition system. In addition, multijoints estimation is important for controlling robotic device comparing with single-joint estimation [35, 36]. Furthermore, in case of Motion Set 1, the proposed system achieved a 7.1–11.8% RMS error in its estimation. This accuracy is approximately same level achieved by the system described by Smith et al. [37]; however, while Smith's system uses 15 sEMG channels, the proposed system is able to use only 4 due to the proposed NN system structure with time delay, feedback, and certain coefficients. This is a large advantage of our sEMG system, which provides a convenient interface system.

The results of the present experiments (Table 4) indicate that the proposed system can be applied to many subjects, that is, it is universal. The system introduced in this paper can be used in other applications for controlling humanoid robot hands, especially with regard to fingertip position, fingertip force, and wrist posture. It is useful and helpful for controlling devices used to assist rehabilitation or for estimating the degree of recovery in patients undergoing rehabilitation.

## Abbreviations

Abd: Abduction-adduction  
BMI: Brain machine interface  
DIP: Distal interphalangeal

EEG: Electroencephalogram  
MP: Metacarpophalangeal  
NN: Neural network  
PIP: Proximal interphalangeal  
RMS: Root means square  
sEMG: Surface electromyogram  
WL: Waveform length.

## Conflict of Interests

The authors declare that they have no competing interests.

## Authors' Contribution

M. Hioki designed the system and experiments, conducted data collection and analysis, and wrote the paper. H. Kawasaki supervised the study. Both authors have read and approved the final paper.

## References

- [1] D. Coyle, G. Prasad, and T. M. McGinnity, "A time-series prediction approach for feature extraction in a brain-computer interface," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 13, no. 4, pp. 461–467, 2005.
- [2] R. Ortner, B. Z. Allison, G. Korisek, H. Gaggel, and G. Pfurtscheller, "An SSVEP BCI to control a hand orthosis for persons with tetraplegia," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 1, pp. 1–5, 2011.
- [3] D. Tkach, H. Huang, and T. A. Kuiken, "Study of stability of time-domain features for electromyographic pattern recognition," *Journal of NeuroEngineering and Rehabilitation*, vol. 7, no. 1, article 21, 2010.
- [4] M. Asghari Oskoei and H. Hu, "Myoelectric control systems—a survey," *Biomedical Signal Processing and Control*, vol. 2, no. 4, pp. 275–294, 2007.
- [5] K. C. McGill and H. R. Marateb, "Rigorous a posteriori assessment of accuracy in EMG decomposition," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 1, pp. 54–63, 2011.
- [6] K. R. Wheeler, M. H. Change, and K. H. Knuth, "Gesture-based control and EMG decomposition," *IEEE Transactions on System, Man, and Cybernetics*, vol. 36, no. 4, pp. 503–514, 2006.
- [7] Otto Bock Co., Ltd., <http://www.ottobockus.com>.
- [8] SPS Co., Ltd., <http://www.spsco.com/>.
- [9] Harada Electronics Industry Ltd., <http://www.h-e-i.co.jp/index.html>.
- [10] S. A. Dalley, T. E. Wiste, T. J. Withrow, and M. Goldfarb, "Design of a multifunctional anthropomorphic prosthetic hand with extrinsic actuation," *IEEE/ASME Transactions on Mechatronics*, vol. 14, no. 6, pp. 699–706, 2009.
- [11] K. B. Fite, T. J. Withrow, X. Shen, K. W. Wait, J. E. Mitchell, and M. Goldfarb, "A gas-actuated anthropomorphic prosthesis for transhumeral amputees," *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 159–169, 2008.
- [12] L. Zollo, S. Roccella, E. Guglielmelli, M. C. Carrozza, and P. Dario, "Biomechatronic design and control of an anthropomorphic artificial hand for prosthetic and robotic applications," *IEEE/ASME Transactions on Mechatronics*, vol. 12, no. 4, pp. 418–429, 2007.

- [13] C. G. Burgar, P. S. Lum, P. C. Shor, and H. F. M. Van Der Loos, "Development of robots for rehabilitation therapy: the Palo Alto VA/Stanford experience," *Journal of Rehabilitation Research and Development*, vol. 37, no. 6, pp. 663–673, 2000.
- [14] D. J. Reinkensmeyer, C. T. Pang, J. A. Nessler, and C. C. Painter, "Web-based telerehabilitation for the upper extremity after stroke," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 10, no. 2, pp. 102–108, 2002.
- [15] J. Oblak, I. Cikajlo, and Z. Matjačić, "Universal haptic drive: a robot for arm and wrist rehabilitation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 3, pp. 293–302, 2010.
- [16] P. R. Culmer, A. E. Jackson, S. Makower et al., "A control strategy for upper limb robotic rehabilitation with a dual robot system," *IEEE/ASME Transactions on Mechatronics*, vol. 15, no. 4, Article ID 5263023, pp. 575–585, 2010.
- [17] J. C. Perry, J. Rosen, and S. Burns, "Upper-limb powered exoskeleton design," *IEEE/ASME Transactions on Mechatronics*, vol. 12, no. 4, pp. 408–417, 2007.
- [18] M. D. Ellis, T. M. Sukal-Moulton, and J. P. Dewald, "Impairment-based 3-D robotic intervention improves upper extremity work area in chronic stroke: targeting abnormal joint torque coupling with progressive shoulder abduction loading," *IEEE Transactions on Robotics*, vol. 25, no. 3, pp. 549–555, 2009.
- [19] K. Kiguchi, K. Iwami, M. Yasuda, K. Watanabe, and T. Fukuda, "An exoskeletal robot for human shoulder joint motion assist," *IEEE/ASME Transactions on Mechatronics*, vol. 8, no. 1, pp. 125–135, 2003.
- [20] L. Masia, H. I. Krebs, P. Cappa, and N. Hogan, "Design and characterization of hand module for whole-arm rehabilitation following stroke," *IEEE/ASME Transactions on Mechatronics*, vol. 12, no. 4, pp. 399–407, 2007.
- [21] Y. Choi, J. Gordon, D. Kim, and N. Schweighofer, "An adaptive automated robotic task-practice system for rehabilitation of arm functions after stroke," *IEEE Transactions on Robotics*, vol. 25, no. 3, pp. 556–568, 2009.
- [22] L. Connelly, Y. Jia, M. L. Toro, M. E. Stoykov, R. V. Kenyon, and D. G. Kamper, "A pneumatic glove and immersive virtual reality environment for hand rehabilitative training after stroke," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 5, pp. 551–559, 2010.
- [23] L. Dovat, O. Lamercy, R. Gassert et al., "HandCARE: a cable-actuated rehabilitation system to train hand function after stroke," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 16, no. 6, pp. 582–591, 2008.
- [24] A. Gupta and M. K. O'Malley, "Design of a haptic arm exoskeleton for training and rehabilitation," *IEEE/ASME Transactions on Mechatronics*, vol. 11, no. 3, pp. 280–289, 2006.
- [25] H. Kawasaki, S. Ito, Y. Ishigure et al., "Development of hand motion assist robot for rehabilitation therapy by patient self-motion control," in *Proceedings of IEEE International Conference on Rehabilitation Robotics (ICORR '07)*, pp. 234–240, 2007.
- [26] T. Mouri, H. Kawasaki, T. Aoki, Y. Nishimoto, S. Ito, and S. Ueki, "Telerehabilitation for fingers and wrist using a hand rehabilitation support system and robot hand," in *Proceedings of the 9th International IFAC Symposium on Robot Control (SYROCO '09)*, pp. 751–756, 2009.
- [27] K. Kong and D. Jeon, "Fuzzy control of a new tendon-driven exoskeletal power assistive device," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM '05)*, pp. 146–151, July 2005.
- [28] X. L. Hu, K. Y. Tong, R. Song, X. J. Zheng, and W. W. F. Leung, "A randomized controlled trial on the recovery process of wrist rehabilitation assisted by electromyography (EMG)-driven robot for chronic stroke," in *Proceedings of the IEEE International Conference on Rehabilitation Robotics (ICORR '09)*, pp. 28–33, June 2009.
- [29] J. U. Chu, I. Moon, Y. J. Lee, S. K. Kim, and M. S. Mun, "A supervised feature-projection-based real-time EMG pattern recognition for multifunction myoelectric hand control," *IEEE/ASME Transactions on Mechatronics*, vol. 12, no. 3, pp. 282–290, 2007.
- [30] J. U. Chu and Y. J. Lee, "Conjugate-prior-penalized learning of gaussian mixture models for multifunction myoelectric hand control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 17, no. 3, pp. 287–297, 2009.
- [31] Y. H. Liu, H. P. Huang, and C. H. Weng, "Recognition of electromyographic signals using cascaded kernel learning machine," *IEEE/ASME Transactions on Mechatronics*, vol. 12, no. 3, pp. 253–264, 2007.
- [32] J. W. Sensinger, B. A. Lock, and T. A. Kuiken, "Adaptive pattern recognition of myoelectric signals: exploration of conceptual framework and practical algorithms," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 17, no. 3, pp. 270–278, 2009.
- [33] L. J. Hargrove, E. J. Scheme, K. B. Englehart, and B. S. Hudgins, "Multiple binary classifications via linear discriminant analysis for improved controllability of a powered prosthesis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 1, pp. 49–57, 2010.
- [34] N. Bu, M. Okamoto, and T. Tsuji, "A hybrid motion classification approach for EMG-based human—robot interfaces using bayesian and neural networks," *IEEE Transactions on Robotics*, vol. 25, no. 3, pp. 502–511, 2009.
- [35] N. A. Shrirao, N. P. Reddy, and D. R. Kosuri, "Neural network committees for finger joint angle estimation from surface EMG signals," *BioMedical Engineering Online*, vol. 8, article 2, 2009.
- [36] T. Kitamura, N. Tsujiuchi, and T. Koizumi, "Manipulation of robot hand based on motion estimation using EMG signals," *Transactions of the Japan Society of Mechanical Engineers*, vol. 73, no. 11, pp. 3024–3030, 2007.
- [37] R. J. Smith, F. Tenore, D. Huberdeau, R. Etienne-Cummings, and N. V. Thakor, "Continuous decoding of finger position from surface EMG signals for the control of powered prostheses," in *Proceedings of the 30th Annual International IEEE EMBS Conference*, pp. 197–200, August 2008.



