Direct Estimation of Wrist Joint Angular Velocities from Surface EMGs by Using an SDNN Function Approximator

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

[1] University of Tsukuba, 1-1-1 Tennodai, Tsukuba, Ibaraki, Japan,
{horie|tanno|mor}@bcl.esys.tsukuba.ac.jp,
[2] Sapporo University of Health Sciences, 2-1-15 Nakanuma Nishi 4, Higashi-ku, Sapporo, Hokkaido, Japan,
sue@sapporo-hokeniryou-u.ac.jp

Abstract. The present paper proposes a method for estimating joint angular velocities from multi-channel surface electromyogram (sEMG) signals. This method uses a selective desensitization neural network (SDNN) as a function approximator that learns the relation between integrated sEMG signals and instantaneous joint angular velocities. A comparison experiment with a Kalman filter model shows that this method can estimate wrist angular velocities in real time with high accuracy, especially during rapid motion.

Keywords: Surface Electromyogram, Angular Velocity Estimation, Selective Desensitization Neural Network

1 Introduction

Surface electromyograms (surface EMGs) are electrical activities recorded using skin surface electrodes. Surface EMGs are produced by skeletal muscles and contain information about a motion and its purpose. Recently, methods have been proposed for recognizing human motions from surface EMGs in order to develop an EMG-based human-machine interface. We can classify these methods into two types: classification of motion types and estimation of the joint angles.

Methods that can recognize complex hand motions in real time have been proposed (e.g., [FO1]). However, these methods can recognize only the summary of the motion purpose and classify only six to eight types of motion with high accuracy. We cannot develop a user-friendly interface that reflects the user's intentions adequately with only classification methods.

Joint-angle estimation methods (e.g., [KK1]) can recognize more detailed purposes of motion than classification methods. However, these methods cannot estimate joint angles during rapid motion. The non-linear relationship between the isometric muscle strength and the joint angle during rapid motion makes it difficult for these methods to model the motion. Thus, these methods cannot be applied to a practical interface.

In addition, the estimation method of joint angles and velocities using a Kalman filter (KF) [AK1] has been proposed. This method, however, was applied only to slow motion (about 30 deg/s). No method has been proposed for estimating the angular velocities of fast-moving joints such as the wrist (wrist angular velocities may reach 1,900 deg/s at a maximum). Furthermore, the KF may not have ability to express the dynamics of wrist rotation around the roll axis because of the non-linearity of the relation between the surface EMGs and the angular velocity.

In this study, we model the relationship between multichannel surface EMGs and wrist joint angular velocities using a function approximator and estimate the latter from the former. This method will enable an interface to use motion-purpose information which could not previously be obtained.

We choose a selective desensitization neural network (SDNN) as the function approximator. Recent studies have shown that the SDNN has excellent learning and generalization abilities. For example, Nonaka et al. applied an SDNN to the approximation of a two-variable function and showed that the SDNN can learn a non-linear and discontinuous function with a small training sample [NM1]. These features will enable the proposed method to learn the complicated relation between surface EMGs and the instantaneous joint angular velocities.

The following sections describe the SDNN and our proposed method. We also verified the efficacy of the proposed method through the estimation of wrist angular velocities.

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 [NM1]. In this section, we explain how a function $y = f(\mathbf{x})$ is approximated based on an SDNN given a continuous-valued input vector $\mathbf{x} = (x_1, \ldots, x_m)(m \geqq 2)$.

The input layer of the SDNN consists of m neural groups. 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 interval corresponds to different activity patterns which are configured using equal numbers, $+1$ and -1. The pattern changes gradually as the variable value changes progressively, resulting in a high correlation between the patterns when the values are close and a low correlation when the values are far apart. The correlation between patterns of the values that are far apart is 0.

The middle layer comprises $m(m-1)$ $(= {}_m\mathrm{P}_2)$ neural groups $(G^{1,1}, G^{1,2}, \ldots, G^{m,m-1})$. 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 output of the units, regardless of their input and inner potential. For example, if the ith units of G^{μ} are desensitized by the jth unit of G^ν, the output of the ith unit of $G^{\mu,\nu}$ is given by

$$g_i^{\mu,\nu} = \frac{g_j^\nu + 1}{2} \cdot g_i^\mu. \tag{1}$$

where g_i^μ is the output of the ith unit of G^μ and g_j^ν is the output of the jth unit of G^ν. This creates a pattern in which half of the units are 0 and the rest are the same as $G^m u$ (either +1 or −1).

The output layer of the SDNN comprises l units. Each of the units is connected to all of the units in the middle layer. The output of the ith unit in the output layer is calculated by

$$\begin{aligned} y^i &= H\left(\textstyle\sum_{\mu,\nu(\mu\neq\nu)} \sum_{j=1}^{n} w_{i,j}^{\mu,\nu} \cdot g_j^{\mu,\nu} + h_i\right) \\ H(u) &= \begin{cases} 1 \ (u > 0) \\ 0 \ (otherwise) \end{cases} \end{aligned} \tag{2}$$

where $w_{i,j}^{\mu,\nu}$ is the synaptic weight from the jth unit of the $G^{\mu,\nu}$ 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 an output of 1 in the output layer. Learning is achieved by error-correction training (the p-delta learning rule [AW1]) in this network.

3 Proposed method

Our proposed method has three components: surface EMG acquisition, signal preprocessing, and function approximation (Figure 3.1).

3.1 Surface EMG Acquisition

The surface EMGs are measured at 10 points on the forearms (Figure 3.1). A relatively large number of sensors is required, but it is not necessary to position the sensors accurately on specific muscles. The surface EMGs are 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. In the proposed method, the integral of the surface EMG (IEMG) and its mean over the previous 300 ms (average IEMG: AIEMG) are obtained as features.

IEMGs can be obtained easily with little time lag using a low-pass filter. Thus, the IEMG is well-suited for a human-machine interface that requires a rapid response. Unfortunately, this signal is not directly related to joint angular velocities; it is related to the force generated by muscles. It is difficult to achieve accurate estimation with only IEMGs. AIEMG is the mean of the IEMG over the

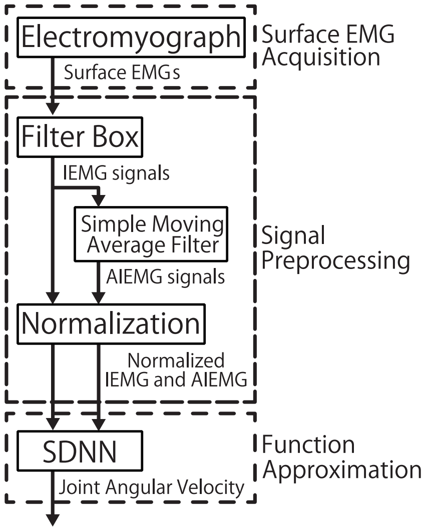
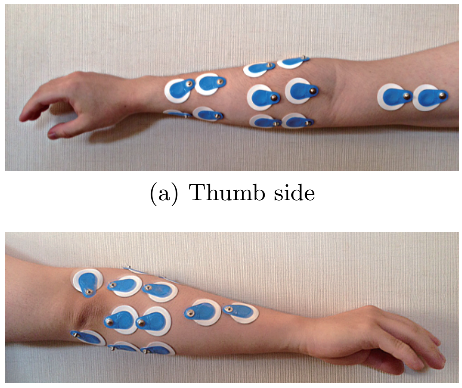


Fig. 1. Structure of the proposed method.



(a) Thumb side

(b) Small finger side

Fig. 2. Arrangement of EMG sensors.

previous 300 ms. We consider that the AIEMG corresponds to muscle contraction speed. AIEMG is well-suited for estimation of joint angular velocities. However, there is a large time lag between the AIEMGs and the actual joint angular velocity itself. Therefore, the combined use of both of these features can be expected to enhance the accuracy and reduce the adverse effects of the time lag between the AIEMGs and the actual angular velocities.

In the proposed method, a filter box (Oisaka Electronic Device Ltd.) and a simple-moving-average (SMA) filter are used to convert the surface EMGs into features. Both features are normalized against their maximum values during each time step before being used as inputs for the SDNN.

3.3 Function Approximation

As a component of function approximation, the SDNN models the relation between the features and the joint angular velocities. The SDNN is constructed as shown in Figure 4.1 and used as a function approximator. Note that the SDNN is prepared for each axis of rotation, but all SDNNs have the same structure and parameters.

The input layer comprises 10 neural groups ($G_1, \cdots, G_{10}$) and 10 other neural groups ($G_{11}, \cdots, G_{20}$), which represent the IEMGs and the AIEMGs, respectively. The activity patterns of the units are determined from the input signals using the parameters $n = 96, q = 96$.

The middle layer has two parts P_1 and P_2, which comprise 90 and 20 neural groups, respectively. The units of the neural groups in P_1 are connected with the units of two neural groups from $G_1, \cdots, G_{10}$. In P_2 the units of the neural groups are connected with the units in the corresponding pair of the neural groups $((G_1, G_{11}), \cdots, (G_{10}, G_{20}))$. In the middle layer, half of the units are desensitized by the corresponding units in the input layer.

The output layer comprises 140 units and calculates the final output of the SDNN by $0.01k - 0.2$ ([-0.2, 1.2]), where k is the number of units with outputs of 1 in the output layer. The output represents the normalized joint angular velocity, which is calculated by

$$V_n(t) = \frac{V_m(t)}{2 \cdot \max_{t'} |V_m(t')|} + 0.5 \tag{3}$$

where $V_n(t)$ and $V_m(t)$ are the normalized and measured joint angular velocities at time t, respectively. The output range of the SDNN is wider than that of the normalized joint angular velocity, which enhances the learning ability of the SDNN. According to the error-correction training algorithm (the p-delta learning rule [AW1]), the SDNN repeats the training process until the root mean squared error (RMSE) is sufficiently low or a specific number of iterations have been completed.

4 Experiment

To evaluate the proposed method, we performed a comparison experiment with a Kalman filter (KF) model [AK1]. The KF model did not use the AIEMGs as input because a preliminary experiment showed that they decrease the estimation accuracy. To implement the KF model, Matlab and the System Identification Toolbox were used.

4.1 Method

A three-axis gyroscope (MP-G3-2000B, MicroStone Co. Ltd.) was used to measure the wrist angular velocities. The gyroscope was mounted on the back of the right hand (Figure 4) and measured the angular velocities around the pitch and roll axes. The velocities were normalized against the maximum values during each time step.

To obtain measurements of wrist angular velocities and surface EMGs, eight male subjects (24 ± 2 years old) were asked to execute flexion-extension and supination-pronation repeatedly at different speeds for 10 s. The subjects repeated this task nine times for each motion, and finally, we obtained 18 (2 motions $\times$ 9 repetitions) samples in total. Next, we selected six samples (three samples for each motion) to train the corresponding SDNN and performed three-fold cross-validation.

4.2 Results

The experimental results are summarized in Table 1. The RMSE values around the pitch axis (corresponding to flexion and extension) with the proposed method

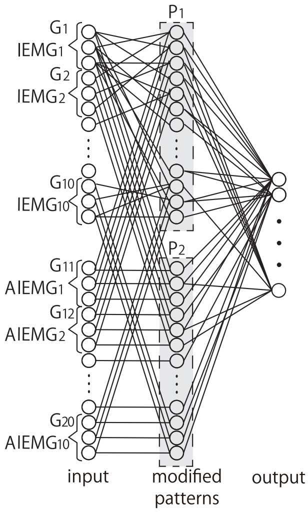


Fig. 3. Structure of the SDNN used in the proposed method.

Fig. 4. Installed gyroscope.

and the KF model were 79.7 and 97.7 deg/s, respectively. Each system could estimate the approximate angular velocities around the pitch axis. There were no significant differences between the methods. In contrast, the RMSE around the roll axis (corresponding to supination and pronation) with the KF model (232.8 deg/s) were two times higher than those obtained with the proposed method ($p < 0.01$: Wilcoxon signed-rank test).

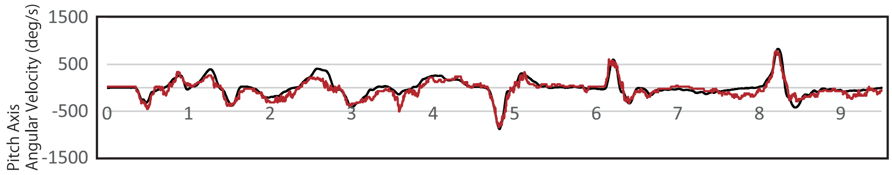
Figures 5 and 6 show the example of the estimated angular velocities with the proposed method and the KF model, respectively. Figure 6(b) demonstrates that the KF model could not estimate the angular velocity around the roll axis, especially during rapid motion.

The estimation accuracy of the KF model depends on the linearity of the relation between the IEMG signals and the angular velocities. The relation between the signals and the joint angular velocities around the roll axis may be very complicated because pronation and supination require the coordination of multiple muscles (flexor carpi radialis, pronator teres, etc.), making it difficult for the KF model to solve this problem.

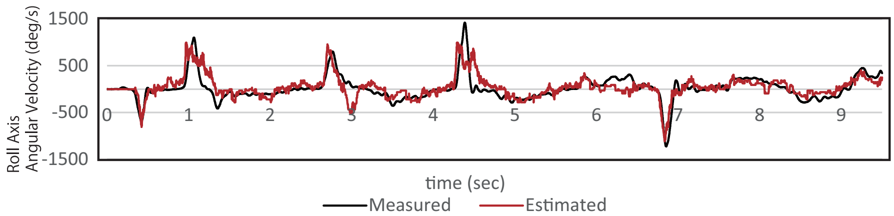
In contrast, the SDNN has adequate ability to learn the non-linearity relationship. Thus, the proposed methods can handle rapid motion or rotation around the roll axis. The difference in learning ability between the KF and the SDNN caused the variation in estimation accuracy.

5 Conclusion

In this study, we proposed a method for directly estimating the wrist angular velocities around the pitch and roll axes from multichannel surface EMGs using

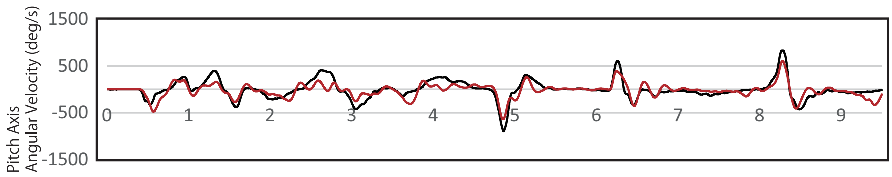


(a) Flexion-Extension

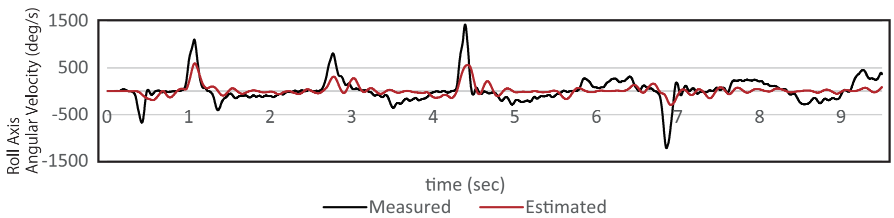


(b) Supination-Pronation

Fig. 5. Example of joint angular velocity estimation with the proposed method.



(a) Flexion-Extension



(b) Supination-Pronation

Fig. 6. Example of joint angular velocity estimation with the KF.

Table 1. Root mean squared error among the estimated angular speeds (deg/s).

(a) Around the pitch axis

	Subject A	B	C	D	E	F	G	H	Ave.
SDNN	63.9	60.0	84.8	89.5	76.3	60.4	113.3	89.6	79.7
KF	91.7	43.1	137.1	142.5	108.7	49.9	127.3	81.4	97.7

(b) Around the roll axis

	Subject A	B	C	D	E	F	G	H	Ave.
SDNN	115.3	105.7	122.4	140.1	124.3	133.6	134.8	100.5	121.9
KF	243.3	150.9	265.8	247.9	220.5	272.0	285.0	176.7	232.8

SDNNs. The experimental results show that the proposed method can estimate the approximate angular velocities of the wrist. The proposed method will be useful for recognizing the purpose of the rapid motion.

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 and by Tateishi Science and Technology Foundation grant number 2157011.

References

[FO1] Fukuda, O., Tsuji, T., Kaneko, M., Otsuka, A.: A Human-Assisting Manipulator Teleoperated by EMG Signals and Arm Motions. IEEE Transaction on Robotics and Automation, 19, 2, 210–222 (2003).

[KK1] Koike, Y., Kawato, M.: Estimation of Arm Posture in 3D–Space from Surface EMG Signals Using a Neural Network Model. IEICE Transactions on Information and Systems, E77-D, 4, 368–375 (1994)

[AK1] Artemiadis, P. K., Kyriakopoulous, K. J.: EMG–Based Control of a Robot Arm Using Low–Dimensional Embeddings. IEEE Transaction on Robotics, 26, 2, 393–398 (2010)

[NM1] Nonaka, K., Tanaka, F., Morita, M.: The Capability of Selective Desensitization Neural Networks at Two-Variable Function Approximation. IEICE Technical Report, Neurocomputing, 111, 96, 113–118 (2011)

[AW1] Auer, P., Burgsteiner, H., Maass, W.: A learning rule for very simple universal approximators consisting of a single layer of perceptrons. Neural Networks, 21, 5, 786–795 (2008).