2021 43rd Annual International Conference of the
IEEE Engineering in Medicine & Biology Society (EMBC)
Oct 31 - Nov 4, 2021. Virtual Conference

Time-domain Mixup Source Data Augmentation of sEMGs for Motion Recognition towards Efficient Style Transfer Mapping

Suguru Kanoga, Tomoumi Takase, Takayuki Hoshino, and Hideki Asoh, *Member*, *IEEE*

***Abstract*— Motion recognition based on surface electromyogram (sEMG) recorded from the forearm is attracting attention for its applicability because it easily integrates with wearable devices and has a high signal-to-noise ratio. Inter-subject variability and inadequate data availability are common problems encountered in classifiers. Transfer learning (TL) techniques can reduce the inter-subject variability; however, when the amount of data recorded from each source subject is small, the TL-combined classifier is prone to overfitting problems. In this study, we tested the accuracy of motion recognition with and without TL when the source dataset was increased up to 10 times with a time-domain data augmentation method called mixup. The performance was evaluated using an 8-class sEMG dataset containing wearable sensing data from 25 subjects. We found that mixup improved the performance of TL-combined classifiers (support vector machine and 4-layered fully connected feedforward neural network). In future work, we plan to investigate the relationship between the amount of data and sEMG-based motion recognition by comparing multiple sEMG datasets and multiple data augmentation methods.**

I. Introduction

Myoelectric control systems (MCSs) provide a communication channel between humans and external machines by measuring and analyzing surface electromyograms (sEMGs). The inclusion of a machine learning-based motion recognition module in the system allows for robust control; thus, many MCSs have been developed to combine with many applications, such as prostheses [1] and robotic arms [2]. Owing to a large difference between individuals (i.e., inter-subject variability), it is necessary to learn classifiers for each individual (i.e., within-subject classifiers); however, the construction of a within-subject classifier requires a considerable time for data measurement from the user. Recently, it has been reported that transfer learning (TL) techniques can provide an effective cross-subject classifier for a user by using a premeasured dataset from other users (source subjects), which requires only a small amount of calibration data from the target user [3], [4]. Thus, the realization of an easy-to-use system using sEMG has been attracting attention.

The amount of data from a target user as well as each source subject is limited because measuring a large amount of labeling data from each subject is difficult owing to the time required for measurement and the burden of labeling. Accurate TL-combined classifiers based on deep learning (DL) approaches, such as the fully connected feedforward neural network (FCFNN), convolutional neural network (CNN), and recurrent neural network (RNN) are useful [3], [4] for large datasets, such as NINAPRO [5] and CapgMyo [6]; however, it may be difficult to construct them in environments where a small source dataset is prepared.

To develop a pipeline that works well when the amount of data for both target and source subjects is limited, we proposed a shallow TL technique called style transfer mapping (STM) that transforms target data into representative points of the source dataset using the affine transform [7]. In our previous study, seven source subject data that were similar to the target user were integrated into one large source dataset to find representative points with diversity. This technique greatly improves the accuracy of a cross-subject support vector machine (SVM) classifier. However, the amount of data in the integrated dataset was still limited because each source subject had a small amount of data. In addition, it did not address individual differences among source subjects. Owing to inadequate data availability and the large inter-subject variability of source subjects, the TL-combined classifier easily causes an overfitting problem. Because the cross-subject classifier itself should refer to the entire source dataset and learn good classification boundaries, overfitting is undesirable and can be improved. If the amount of source data can be increased while mitigating the differences in the data of each source subject, the ability of the TL-combined classifier may be improved.

Data augmentation (DA) methods increase the amount and variety of original sEMG data on the subjects by transforming an existing labeled sample, thereby allowing the model to learn the range of intra-class variation that may be observed [8]. We assumed that augmenting the data by mixing the same class of data among the source subjects can mitigate the effects of individual differences in the source dataset, which will also contribute to improving the accuracy of the TL-combined classifiers. Thus, in this study, we applied a DA method called mixup [9] to mix sEMG data from different source subjects in the time domain and assess the impact on two classifiers (SVM and 4-layered FCFNN) and STM-combined ones.

II. Materials and Methods

The overall analysis pipeline is shown in Fig. 1, and the respective modules are described in the following subsections.

A. Dataset

We used the same dataset as in the previous study [7], which consists of data from 25 individuals performing eight

*This work was supported in part by the New Energy and Industrial Technology Development Organization (NEDO) [project number JPNP20006], Japan, and JSPS KAKENHI Grant Number JP20K19854 and JP20K19888.

S. Kanoga, T. Takase, T. Hoshino, and H. Asoh are with the National Institute of Advanced Industrial Science and Technology (AIST), 2-4-7 Aomi, Koto, Tokyo 135-0064, Japan (Email: s.kanouga@aist.go.jp).

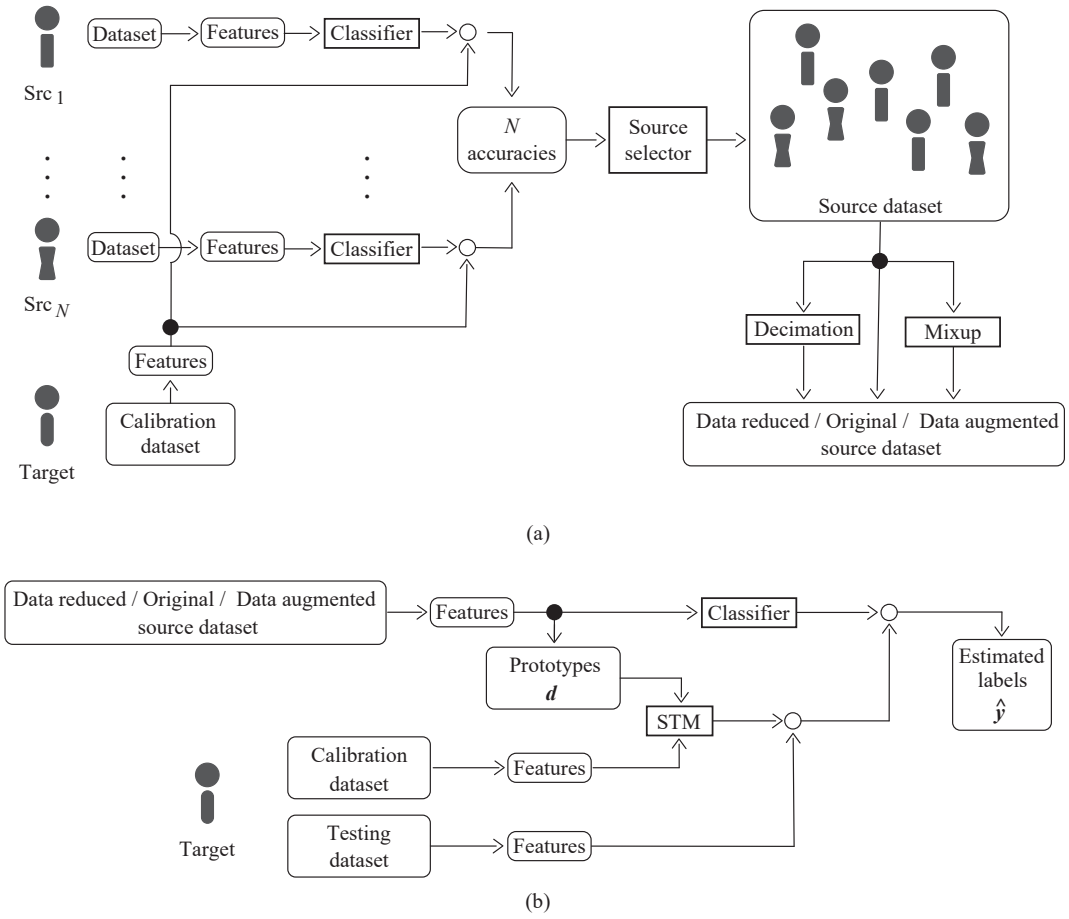

Fig. 1. (a) Making data reduced, original, and data augmented source datasets. (b) TL-combined classification using the prepared source datasets.

types of 1-degree-of-freedom forearm motions. Each trial has a 6-second length and is repeated five times, resulting in 40 trials. The data were recorded at a sampling rate of 200 Hz with eight electrode pairs attached to the right forearm using a Myo Gesture Control Armband (Thalmic Labs).

B. Preprocessing

The signals were high-pass filtered at 15 Hz through a fifth-order Butterworth filter. A data interval of 1.5 s was extracted from 6 s of data using a sample entropy-based onset detection method [7]. The first time point that exceeded the threshold was identified as the onset for the active segment.

C. Data Augmentation

The amount of data in the source dataset can be easily increased using a simple mixup operation as follows:

$$\tilde{\boldsymbol{x}} = \lambda \boldsymbol{x}_i + (1-\lambda)\boldsymbol{x}_j, \tag{1}$$

where $\boldsymbol{x}_i$, $\boldsymbol{x}_j$, and λ are the i-th and j-th source data, and the mix rate ($0 < \lambda < 1$) sampled from the beta distribution $\mathcal{B}(\alpha, \alpha)$ ($\alpha \in (0, \infty)$). We set α to 1 to generate samples uniformly and determine λ for each mixup operation. Considering the fact that labels can also be mixed in the same way, we applied the mixup method to the same-class data.

It is clear from Eq. (1) that the mixup method is a linear interpolation of two types of data, and the augmented data has an intermediate characteristics between them. Simply, this method can mitigate the differences in the datasets of each source subject. In our experiments, a classifier was trained for each target subject with a dataset that was prepared individually by grouping samples from seven source subjects that were similar to the target subject, as described in Section II-E. Two data were randomly selected from the source dataset. In addition, this method was performed between different trials as well as between the same trials; however, two data were constantly selected without duplication (i.e., from different subjects or different trials) because mixing between the same trials in a subject does not generate new data.

To include verification when the amount of data decreased, we prepared seven different types of data (1/10, 1/5, 1/2, 1, 2, 5, and 10 times). The number of data per class was adjusted to be the same.

D. Feature Extraction

A 1.5-second, 8-channel segment was further subdivided into 0.25-second analysis windows with class labels. An

analysis window has an overlap of 80% with the previous and next windows (i.e., 0.05-second shifts); thus, a 1.5-second segment yielded 26 0.25-second analysis windows.

For an analysis window of one channel, a gold standard 11-dimensional feature set (mean absolute value, zero crossing, slope sign changes, waveform length, root mean square, and sixth-order autoregressive coefficients) [10] was extracted. We then have an 8-channel data; thus, a 0.25-second analysis window was translated into an 88-dimensional feature vector (11 features, 8 channels). All these features were normalized using the z-score to eliminate the scaling effects among different features.

E. Classifier

We applied SVM with a radial basis function kernel and 4-layered FCFNN (containing three hidden layers) classifiers for 8-class motion recognition. SVM was implemented in LIBSVM for MATLAB [11], and FCFNN was implemented in MATLAB's DL toolbox based on full-batch training. For the target user, we trained a classifier by grouping the data of seven subjects other than the user. These seven subjects were those whose classifiers showed high classification accuracy against the calibration data of the user (see Fig. 1(a)). Each individual classifier has been optimized with 4-fold cross validation (CV)).

The hyperparameters in SVM and FCFNN were optimized by grid-search with 4-fold CV of data reduced, original, and data augmented source datasets. The hyperparameters in SVM, cost parameter C, and kernel parameter σ were tuned with $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$. The hyperparameters in FCFNN, the number of units of three hidden layers, and weight decay were tuned with $\{22, 44, \ldots, 352\}$ and $\{e^{-2}, e^{-3}, \ldots, e^{-7}\}$. We omitted the case in which the number of units increased more than the number of units in the previous hidden layer. In addition, the quartet (initial learning rate, activation function, optimizer, maximum number of epochs) was set to 0.01, ReLU, Adam, 100.

F. Transfer Learning

To transform the target data $\boldsymbol{x}_n$ into the source dataset, we applied a supervised STM algorithm that learns the affine transform by minimizing the weighted squared error [12]:

$$\min_{\boldsymbol{A},\boldsymbol{b}} \sum_{n=1}^{N} ||\boldsymbol{A}\boldsymbol{x}_n + \boldsymbol{b} - \boldsymbol{d}_n||_2^2 + \beta||\boldsymbol{A} - \boldsymbol{I}||_F^2 + \gamma||\boldsymbol{b}||_2^2, \quad (2)$$

where $||\cdot||_F^2$, $||\cdot||_2^2$, and $\boldsymbol{I}$ are the Frobenius norm of the matrix, L_2 norm of the vector, and identity matrix, respectively. The second and third terms prevent it from distancing from the original position. STM cannot map calibration data directly to the source dataset; however, by finding representative points (prototypes) that represent the source dataset well, the STM algorithm can learn the affine transform matrix $\boldsymbol{A}$ and $\boldsymbol{b}$ based on the points as prototypes (see Fig. 1(b)). We clustered the source dataset via K-means ++ clustering [13] in each class and repeated the operation 10 times to obtain reliable cluster centers. In addition, the number of cluster K was optimized from 1 to 7 via `evalclusters` function with source datasets. The nearest class center to a calibration sample $\boldsymbol{x}_n$ is defined as a prototype $\boldsymbol{d}_n$.

The parameters of the affine transform $\boldsymbol{A}$ and $\boldsymbol{b}$ were trained using prototypes and a calibration dataset (all-class first-trial data of the target). The hyperparameters in the STM and tuning parameters β and γ were tuned with $\{0, 0.2, \ldots, 3\}$ [14] using all class second-trial target data.

G. Evaluation

In this study, we examined 8-class motion recognition performances of two types of classifiers (SVM and FCFNN) and two types of TL-combined classifiers (STM-SVM and STM-FCFNN) using a testing dataset that contains all class third- to fifth-trial data of the target. The source datasets have seven different amounts of data.

III. Results and Discussion

A. Motion Recognition Performances

The classification accuracies with two types of classifiers (SVM and FCFNN) and two types of TL-combined classifiers (STM-SVM and STM-FCFNN) over seven different amounts of source dataset are shown in Fig. 2). Both SVM and FCFNN showed a drastic improvement in accuracy by applying STM, and this trend was observed regardless of the number of data (75.00–80.38% in SVM and FCFNN and 82.21–88.54% in STM-SVM and STM-FCFNN). The accuracy of STM-SVM was higher than that of STM-FCFNN, with the highest accuracy of 88.54% for STM-SVM using two and five source data (see the top figure in Fig. 2). However, in both cases, the TL-combined classifiers showed an improvement in accuracy with more data than with the original amount of data. In particular, the largest improvement of 1.03% was observed for FCFNN when the amount of source data was five times larger (see the bottom figure in Fig. 2).

A decrease in accuracy was observed when TL was not used. Even if the data of the seven source subjects are similar, they are different from the target subject, and increasing the data of the source subjects may overfit the data and cause a decrease in the classification accuracy of the target. However, it is important that cross-subject classifiers are fitted by the source dataset if they are to be applied to TL.

Because SVM finds support vectors, which assist in finding the classification boundary from a small amount of data, the location of support vectors may not change even if a DA method is applied. In the case of sEMG, where the amount of data is limited, it is natural that SVM is a famous and gold standard classifier for sEMG-based motion recognition. It has been reported in various studies that if sufficient number and variety of data are available, motion recognition performance is higher in DL-based approaches than in traditional shallow machine learning methods, such as linear discriminant analysis, hidden Markov model, and SVM [15], [16]. The results of this experiment showed that the change in accuracy with a change in the number of data

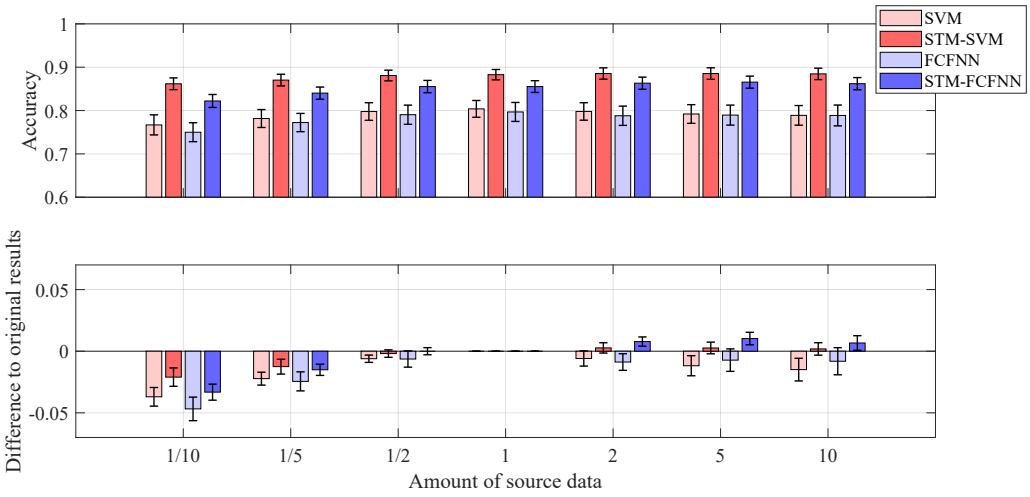

Fig. 2. Classification accuracies with two types of classifiers and two types of transfer learning-combined classifiers over seven different amounts of source datasets (top). Difference in accuracy to original data amount (bottom).

was smaller for STM-SVM than for STM-FCFNN, thus, supporting the above reason.

DL architectures will be used for classifiers in sEMG-based motion recognition, once DA methods have shown in various situations that they can successfully increase the amount and variety of original data and become a common technology in the community. The same suggestion has been considered in TL techniques [3], [4]. When determining the relationship between the target and source datasets and bridging them, it is expected to be more accurate if the relationship is represented in more (deeper) layers rather than only in a single (shallow) layer. We used a shallow TL technique, supervised STM, to transform the target data into the source dataset to avoid retraining the classifier. However, by applying the mixup method, the performance of TL-combined classifiers (STM-SVM and STM-FCFNN) was improved. Thus, DA methods have the potential to contribute to the improvement of TL-combined classifiers.

B. Future Work

In future work, we will include a comprehensive comparison of DA methods based on generative models, such as generative adversarial networks [17], [18] to create synthetic data and methods that synthesize real data. In addition, we will construct a DL classifier that combines DA and TL methods to build a highly accurate MCS that works well even with a small number of target and source data.

REFERENCES

[1] F. Peng, C. Zhang, B. Xu, J. Li, Z. Wang, and H. Su, "Locomotion prediction for lower limb prostheses in complex environments via sEMG and inertial sensors," *Complexity*, vol. 2020, 2020.

[2] B. Guo, Y. Ma, J. Yang, Z. Wang, and X. Zhang, "Lw-CNN-based myoelectric signal recognition and real-time control of robotic arm for upper-limb rehabilitation," *Comput. Intel. Neurosc.*, vol. 2020, 2020.

[3] A. Gautam, M. Panwar, D. Biswas, and A. Acharyya, "MyoNet: A transfer-learning-based LRCN for lower limb movement recognition and knee joint angle prediction for remote monitoring of rehabilitation progress from sEMG," *IEEE J. Transl. Eng. Health Med.*, vol. 8, pp. 1–10, 2020.

[4] X. Chen, Y. Li, R. Hu, X. Zhang, and X. Chen, "Hand gesture recognition based on surface electromyography using convolutional neural network with transfer learning method," *IEEE J. Biomed. Health Inform.*, 2020.

[5] M. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A.-G. M. Hager, S. Elsig, G. Giatsidis, F. Bassetto, and H. Müller, "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Sci. Data*, vol. 1, no. 1, pp. 1–13, 2014.

[6] W. Geng, Y. Du, W. Jin, W. Wei, Y. Hu, and J. Li, "Gesture recognition by instantaneous surface EMG images," *Sci. Rep.*, vol. 6, no. 36571, 2016.

[7] S. Kanoga, T. Hoshino, and H. Asoh, "Subject transfer framework based on source selection and semi-supervised style transfer mapping for sEMG pattern recognition," in *ICASSP 2020*. IEEE, 2020, pp. 1349–1353.

[8] P. Tsinganos, B. Cornelis, J. Cornelis, B. Jansen, and A. Skodras, "Data augmentation of surface electromyography for hand gesture recognition," *Sensors*, vol. 20, no. 17, 2020.

[9] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *ICLR 2018*, 2018.

[10] Y. Huang, K. B. Englehart, B. Hudgins, and A. D. C. Chan, "A Gaussian mixture model based classification scheme for myoelectric control of powered upper limb prostheses," *IEEE Trans. Biomed. Eng.*, vol. 52, no. 11, pp. 1801–1811, 2005.

[11] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, 2011.

[12] X.-Y. Zhang and C.-L. Liu, "Style transfer matrix learning for writer adaptation," in *CVPR 2011*. IEEE, 2011, pp. 393–400.

[13] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," Tech. Rep., Stanford, 2006.

[14] J. Li, S. Qiu, Y.-Y. Shen, C.-L. Liu, and H. He, "Multisource transfer learning for cross-subject EEG emotion recognition," *IEEE Trans. on Cybern.*, 2019.

[15] Y. Hu, Y. Wong, W. Wei, Y. Du, M. Kankanhalli, and W. Geng, "A novel attention-based hybrid CNN-RNN architecture for sEMG-based gesture recognition," *PLOS ONE*, vol. 13, no. 10, pp. e0206049, 2018.

[16] S. Tam, M. Boukadoum, A. Campeau-Lecours, and B. Gosselin, "A fully embedded adaptive real-time hand gesture classifier leveraging HD-sEMG and deep learning," *IEEE Trans. Biomed. Circuits Syst.*, vol. 14, no. 2, pp. 232–243, 2019.

[17] R. Anicet Z. and E. Luna C., "Parkinson's disease EMG data augmentation and simulation with DCGANs and style transfer," *Sensors*, vol. 20, no. 9, 2020.

[18] E. Campbell, J. A. D. Cameron, and E. Scheme, "Feasibility of data-driven EMG signal generation using a deep generative model," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2020, pp. 3755–3758.