Semi-Supervised Learning for Surface EMG-based Gesture Recognition

Yu Du[1], **Yongkang Wong**[3], **Wenguang Jin**[2], **Wentao Wei**[1], **Yu Hu**[1]
Mohan Kankanhalli[4], **Weidong Geng**[1⋆]
[1]College of Computer Science, Zhejiang University
[2]College of Information Science and Electronic Engineering, Zhejiang University
[3]Smart Systems Institute, National University of Singapore
[4]School of Computing, National University of Singapore

Abstract

Conventionally, gesture recognition based on non-intrusive muscle-computer interfaces required a strongly-supervised learning algorithm and a large amount of labeled training signals of surface electromyography (sEMG). In this work, we show that temporal relationship of sEMG signals and data glove provides implicit supervisory signal for learning the gesture recognition model. To demonstrate this, we present a semi-supervised learning framework with a novel Siamese architecture for sEMG-based gesture recognition. Specifically, we employ auxiliary tasks to learn visual representation; predicting the temporal order of two consecutive sEMG frames; and, optionally, predicting the statistics of 3D hand pose with a sEMG frame. Experiments on the NinaPro, CapgMyo and csl-hdemg datasets validate the efficacy of our proposed approach, especially when the labeled samples are very scarce.

1 Introduction

A Muscle-Computer Interface (MCI) [Saponas *et al.*, 2008] is an interaction methodology that directly transforms myoelectrical signals from mere reflections of muscle activities into interaction commands that convey the user's intention. Gesture recognition based on surface electromyography (sEMG) is the technical core of non-intrusive MCIs, where sEMG measures the muscle's electrical activity from the skin surface using one or more electrodes.

sEMG based gesture recognition can be naturally defined as a pattern classification problem, where a classifier is usually trained with supervised learning approach using large amount of labeled sEMG signals. Compared with more established visual recognition dataset, the quantity and quality of labeled samples in existing sEMG based gesture recognition datasets are relatively poor. The quantity issue can be overcome via data augmentation [Atzori *et al.*, 2016] or using data of a group of subjects [Geng *et al.*, 2016]. The latter is important as the sEMG signals are highly subject specific and vary considerably between recording sessions of the same user under the same experimental paradigm. Furthermore, the statistics of sEMG signals may also change within the same recording session due to fatigue and changes in arm posture [Farina *et al.*, 2014]. Another important issue is that the recorded samples may not perfectly match the gesture label, which is caused by the delay from human reaction time and transition movements. To ensure the label quality, existing studies only use a relatively small segment of sEMG signals for model training and evaluation [Atzori *et al.*, 2012; Amma *et al.*, 2015; Geng *et al.*, 2016]. For example, Atzori *et al.* [2012] removed sEMG signals with ambiguous labels by dividing each trial into three equally sized segments and retaining data only from the center segment.

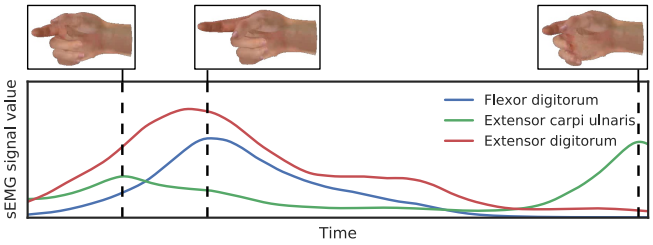

Figure 1: sEMG signals recorded on different muscles and the corresponding 3D hand poses for a pointing gesture.

sEMG signal records muscles' electrical activities from the surface of the skin, which reflect the motor unit action potential generated by the *firing* of muscles. The firing order of the muscles measured by sEMG provides an overview of the neuromuscular activity pattern in the movement [Konrad, 2005], and the relative timing of the firing order over changes in rate and amplitude of movement is preserved across a variety of activities [Fowler, 1983] (Fig. 1). Therefore, temporal relationship between neighbouring frames of sEMG signals reflect the relative timing of muscle activity, and provide a useful supervisory signal for the resulting gestures.

From the machine learning perspective, most of the sEMG-based gesture recognition methods train models under the supervised paradigm, where each training sample is associated with a gesture label. Considering the limitation in the availability of large scale high quality labeled samples, a possible solution is to deploy Semi-Supervised Learning (SSL) algorithms [Shahshahani and Landgrebe, 1994], which make use of both labeled and unlabeled data for training. In the con-

⋆Corresponding author: Weidong Geng (gengwd@zju.edu.cn)

text of deep learning, SSL is often based on multi-task learning [Caruana, 1997], in which one or more auxiliary tasks (supervised or unsupervised) are simultaneously solved together with the main task to learn a shared feature representation, including predicting relative location of image patches [Doersch *et al.*, 2015], real-world physical interaction with observed objects [Pinto *et al.*, 2016] and predicting ambient sound associated with a video frame [Owens *et al.*, 2016].

Our Contribution Inspired by the recent success of deep learning-based hand gesture recognition with a single frame of sEMG signals [Geng *et al.*, 2016], we present a SSL framework to train a classifier by exploring the temporal coherence between signals as an auxiliary task. We formulate the learning task as a multi-task learning problem. Specifically, we train a ConvNet to simultaneously predict three targets with a single frame of sEMG signals: (1) the hand gesture, (2) the temporal order of two consecutive sEMG frames, and, optionally, (3) the statistics of 3D hand pose. The two auxiliary tasks (i.e., task 2 and 3) implicitly require knowledge of dynamics (i.e., derived from sEMG signals) and shapes of the hand movements (i.e., data recorded by data glove), thus help learning useful feature representation of sEMG signals when available gesture labels are limited. Experiments on three benchmark datasets (i.e., NinaPro [Atzori *et al.*, 2014], CapgMyo [Geng *et al.*, 2016], and csl-hdemg [Amma *et al.*, 2015]) indicate that our approach outperforms state-of-the-art methods. To the best of our knowledge, this is the first work to address sEMG-based gesture recognition problem in an end-to-end framework in a semi-supervised manner.

2 Related Work

Deep learning and convolutional neural networks have recently revolutionized the development of machine learning and computer vision applications [Cheng *et al.*, 2016; Pinto *et al.*, 2016; Doersch *et al.*, 2015]. In recent years, deep convolutional networks (ConvNets) have also been applied to recognize hand gestures from sEMG signals [Atzori *et al.*, 2016; Geng *et al.*, 2016]. However, ConvNets have numerous learning parameters that need to be trained over a large amount of quality labeled data.

To handle the problem of the availability of labeled data, SSL approach has been broadly explored in deep learning where a classifier is trained using both labeled and unlabeled samples. A classic approach in SSL is the bootstrapping method, which starts with training an initial model using a small number of labeled examples followed by using the trained model to label the unlabeled data. The model is iteratively retrained using the high-confidence self-labeled examples in addition to the original examples. Cheng *et al.* [2016] proposed a diversity preserving co-training algorithm to guide a ConvNet to learn from the unlabeled RGB-D data by utilizing the complementary cues of the RGB and depth data with bootstrapping model. It achieved competitive performance for object recognition on benchmark RGB-D dataset with only 5% labeled training data. However, the performance is subject to the quality of the inital labeled data. To address this issue, active learning was utilized to select the most informative samples, and combined with SSL for improved performance [Leng *et al.*, 2013; Wang *et al.*, 2016]. Guo *et al.* [2016] extended semi-supervised active learning to handle a scenario where target domain has different but related classes as source domain. SSL in deep learning are often based on multi-task learning [Caruana, 1997], in which one or more auxiliary tasks (supervised or unsupervised) are simulaneously solved together with the main task to learn a shared feature representation. The auxiliary tasks are prediction tasks in which the prediction target is automatically derived from a natural signal, instead of human annotations.

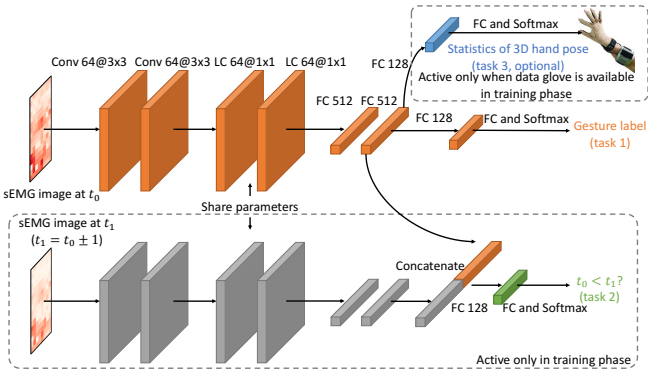

Figure 2: Illustration of the proposed semi-supervised ConvNet for sEMG-based gesture recognition. The networks for three tasks are shown in different colors in which the bottom network (first 6 layers) are shared. The boxes represent the inputs and outputs of different layers of the network. The text between the boxes describe the layers, where Conv, LC and FC denote convolution layer, locally-connected layer and fully-connected layer, respectively. The number after the layer name denotes the number of filters, and the numbers after @ denote convolution kernel size. The networks in the dashed box are only used in training phase.

In the unsupervised learning literature, one way to learn the visual representation (or embedding) is to create a supervised *pretext* task, where a supervisory signal, or regularizer, can be extracted from the unlabeled data. Example of the pretext task include temporal coherence from two consecutive frames [Mobahi *et al.*, 2009; Wang and Gupta, 2015], predicting the relative position of two image patches [Doersch *et al.*, 2015], physical interaction [Pinto *et al.*, 2016], reconstruction from noisy data [Bengio *et al.*, 2014], and utilizing ambient sound to learn visual representation [Owens *et al.*, 2016]. Recently, Stewart and Ermon [2017] utilized physics and domain knowledge as constraint that should hold over the output space to train neural networks. Our pretext task 2 (shown in Fig. 2) can be viewed as an unsupervised special case of rank learning [Joachims, 2002], in which the objective is to predict the relative timing order between inputs.

3 Semi-Supervised Gesture Recognition

Inspired by the recent work on deep learning-based gesture recognition [Geng *et al.*, 2016], we employ a ConvNet to model the gesture classifier in task 1 and the auxiliary predictors in task 2 and 3 (see Fig. 2).

We choose temporal order of neighbouring sEMG frames as the prediction target for task 2 because it reflects the built-in firing order of muscles, which is insensitive to the rate

and amplitude of the performed gesture. We hypothesize that the task of predicting the temporal order implicitly requires knowledge of dynamic features of muscle activity.

We choose the statistics of 3D hand pose as the prediction target for task 3. This is because the sequence of 3D hand poses directly determines the perceived hand gestures and has richer supervisory signals than the gesture labels. 3D hand pose describes the spatial status of the hand at a specific time, and thus provides more fine-grained supervision for the gesture classifier (shown in Fig. 1). Moreover, 3D hand pose is less ambiguous than gesture label in transition movements, and thus provides more reliable supervision. In our framework, we make task 3 as an optional task because the ground truth of 3D hand pose, which is usually recorded by data glove, may not always be available in the training set.

In the following sections, we first formulate the learning problem and then describe the design of the three tasks.

3.1 Problem Statement

Let $\mathcal{L} = \{(\mathbf{x}_i^l, y_i)\}_{i=1}^{N_l}$, where $\mathbf{x}^l \in \mathbb{R}^C$ denotes the instantaneous sEMG signals of the C channels in the training sessions and y_i is the corresponding gesture label. Unlabeled data in the test sessions are denoted as $\mathcal{T} = \{\mathbf{x}_i^t\}_{i=1}^{N_t}$. Gesture recognition with instantaneous sEMG signals is a classification problem in which a classifier f_θ is built to predict the hand gesture to which $\mathbf{x}^t$ belongs, where θ is the unknown parameter estimated from $\mathcal{L}$.

Let $\mathcal{U} = \{\mathbf{x}_i^u\}_{i=1}^{N_u}$ denote the unlabeled sEMG frames, $\mathcal{V} = \left\{(\mathbf{x}_i^v, \mathbf{z}_i)\right\}_{i=1}^{N_v}$ denote all sEMG frames that have corresponding 3D hand poses recorded, where $\mathbf{z}$ is the statistics of 3D hand pose. SSL for sEMG-based gesture recognition is to train a classifier f_θ with $\mathcal{S} = \mathcal{L} \cup \mathcal{U} \cup \mathcal{V}$ to predict the hand gesture to which $\mathbf{x}^t$ belongs. Let $\mathcal{X} = \left\{\mathbf{x} \middle| \mathbf{x} \in \mathcal{S} \text{ or } (\mathbf{x}, \cdot) \in \mathcal{S}\right\}$, $t_\mathbf{x}$ denotes the time of $\mathbf{x}$, $\bar{\mathbf{x}}$ denotes a frame at time $t_\mathbf{x} + \delta$, where $\delta \in \mathcal{D} = \{-1, 1\}$. f_θ is trained by optimizing the following objective function:

$$\underset{\Theta}{\operatorname{argmin}} \frac{1}{N_l} \sum_{(\mathbf{x},y)\in\mathcal{L}} L_l(\mathbf{x}, y|\Theta) + \frac{\alpha}{2N_s} \sum_{\mathbf{x}\in\mathcal{X},\delta\in\mathcal{D}} L_u(\mathbf{x}, \bar{\mathbf{x}}|\Theta) + \frac{\beta}{N_v} \sum_{(\mathbf{x},\mathbf{z})\in\mathcal{V}} L_v(\mathbf{x}, \mathbf{z}|\Theta) \quad (1)$$

where gesture classifier parameters $\theta \subset \Theta$, $N_s = N_l + N_u$ is the number of elements in $\mathcal{S}$, α and β are weight hyper-parameters. L_l, L_u and L_v are the loss functions for (1) gesture classification, (2) temporal order prediction of sEMG frames, and (3) 3D hand pose statistics prediction (optional), respectively.

We use a ConvNet to model the predictors in Eqn. (1). In the training phase, given the sEMG images, we make predictions for the three tasks using three sub-networks, respectively (shown in Fig. 2), calculate the loss in Eqn. (1), back-propagate the gradients and update the parameters. The sub-networks for task 1 and 3 are activated only if the training sample satisfies $\mathbf{x} \in \left\{\mathbf{x} \middle| (\mathbf{x}, y) \in \mathcal{L}\right\}$ and $\mathbf{x} \in \left\{\mathbf{x} \middle| (\mathbf{x}, \mathbf{z}) \in \mathcal{V}\right\}$, respectively. In the recognition phase, the auxiliary sub-networks are dropped, the configuration of the gesture classification network is same as GengNet [Geng *et al.*, 2016], and thus without additional runtime cost.

3.2 Gesture Classification

Our gesture classification ConvNet has eight layers (denoted as MyoNet, colored in orange in Fig. 2). The input to the ConvNet consists of a 1×10 image for NinaPro [Atzori *et al.*, 2014], an 8×16 image for CapgMyo [Geng *et al.*, 2016] and a 7×24 image for csl-hdemg [Amma *et al.*, 2015]. The first two hidden layers are convolutional layers, each of which consists of 64 3×3 filters with a stride of 1 and a zero padding of 1. The next two hidden layers are locally-connected [Taigman *et al.*, 2014], each of which consists of 64 non-overlapping 1×1 filters. The next three hidden layers are fully-connected and consists of 512, 512 and 128 units, respectively. The network ends with a G-way fully-connected layer and a softmax function, where G is the number of gestures. We adopted ReLU non-linearity [Krizhevsky *et al.*, 2012] after each hidden layer, batch normalization [Ioffe and Szegedy, 2015] after the input and before each ReLU non-linearity, and dropout [Srivastava *et al.*, 2014] with a probability of 0.5 after the fourth, fifth and sixth layers.

The first 6 layers are shared in the three tasks, denoted as $h_c(\cdot|\theta_c)$. The gesture classification sub-network consists of the 7th and 8th layers, denoted as $\hat{f}(\cdot|\theta_l)$. The gesture classifier is defined as $f_\theta(\mathbf{x}) = \hat{f}(h_c(\mathbf{x}|\theta_c)|\theta_l)$, where $\theta_c \subset \Theta$ and $\theta_l \subset \Theta$ are learning parameters, and $\theta = \theta_c \cup \theta_l$. The loss function L_l is a cross-entropy loss:

$$L_l(\mathbf{x}, y|\theta) = -\sum_{i=1}^{G} \mathbf{1}_i(y) \log f_\theta^i(\mathbf{x}) \quad (2)$$

where $f_\theta^i(\mathbf{x})$ is the i-th dimension of $f_\theta(\mathbf{x})$, $\mathbf{1}_i(y)$ is the indicator function.

Although multiple frames of sEMG signals are available, we make predictions from a single frame, so that the learned feature representation will be more likely to be transferred to existing single frame recognition framework [Geng *et al.*, 2016].

In the recognition phase, the trained ConvNet is utilized to recognize hand gestures from sEMG images frame by frame, minimizing the observational latency into one frame. Additionally, a majority voting scheme is used when two or more frames are available. Using this scheme, a window of sEMG signals is labeled with the class that receives the most votes.

3.3 Temporal Order Prediction of sEMG frames

We define temporal order prediction of sEMG frames as a binary classification problem, in which a classifier

$$h_u(\mathbf{x}_i, \mathbf{x}_j|\Theta) = \hat{h}_u\Big(\left[h_c(\mathbf{x}_i|\theta_c)^\intercal \; h_c(\mathbf{x}_j|\theta_c)^\intercal\right]^\intercal \Big| \theta_u\Big) \quad (3)$$

is trained to predict the temporal order between two neighbouring sEMG frames $\mathbf{x}_i$ and $\mathbf{x}_j$ in the same recording session (i.e., whether $i < j$ or $i > j$), where $\hat{h}_u(\cdot, \cdot|\theta_u)$ is the prediction sub-network and $\theta_u \subset \Theta$ are learning parameters.

The entire classifier $h_u(\mathbf{x}_i, \mathbf{x}_j|\Theta)$ is modeled by a Siamese network, in which the bottom two streams share the same parameters (the gray and the corresponding orange networks in Fig. 2). The resulting feature vectors $h_c(\mathbf{x}_i|\theta_c)$ and $h_c(\mathbf{x}_j|\theta_c)$ are concatenated and transformed by $\hat{h}_u$, which consists of two fully-connected layers. The fully-connected layers have

the same configuration as that of the 7th and 8th layers of MyoNet, except that the output dimension is 2. The loss function L_u is a cross-entropy loss.

While a direct regression between two consecutive sEMG images is possible, the learned representation may be problematic for gesture classification. This is because the shared network will work like an autoencoder and try to preserve all the details needed for reconstructing the next frame of sEMG signals at pixel level. The resulting feature representation is often redundant for high level classification tasks [Rasmus *et al.*, 2015]. And it may also make the weight hyper-parameter α in Eqn. (1) sensitive to the scale and the channel number of sEMG signals.

For comparison, we trained a model to predict the next frame of sEMG signals, which is equivalent to predicting the difference sEMG image (i.e., changes between two consecutive sEMG images in the temporal sEMG sequence), in lieu of the classification-based model.

3.4 3D Hand Pose Statistics Prediction

In this framework, instead of regressing the joint angles or positions directly from the corresponding sEMG image, we define explicit hand pose categories and formulate the prediction task as a classification problem. This is because the mappings from sEMG to joint angles are different for each joint, thus making it difficult to balance the weights of different joints in the regression loss. For example, extension of the ring finger is usually harder than extension of the index finger. Another reason is that the weight hyper-parameter β in Eqn. (1) may be sensitive to the scale and the dimension of hand pose $\mathbf{z}$ with a regression loss L_v.

We define 3D hand pose statistics prediction as a classification problem, in which a classifier $h_v(\mathbf{x}|\Theta) = \hat{h}_v\big(h_c(\mathbf{x}|\theta_c)\big|\theta_v\big)$ is trained to predict the hand pose label of $\mathbf{x}$. $\hat{h}_v(\cdot|\theta_v)$ is the hand pose statistics prediction sub-network and $\theta_v \subset \Theta$ are learning parameters. The sub-network consists of two fully-connected layers with the same configuration as that of the 7th and 8th layers of MyoNet. The dimensionality of the output layers is the same as the number of hand pose categories. The loss function L_v is a cross-entropy loss.

The 3D hand poses in the training set were first clustered using k-means algorithm. Each hand pose was labeled with the index of the closest centroid. The input of k-means was the 22 channels of raw data collected by data glove (Cyberglove II) and its difference in time. The number of clusters k should be larger than the number of gestures G because we need hand pose labels to provide more fine-grained supervisory signals than gesture labels.

In this work, we also trained models to predict the raw signals collected by data glove and other vector representations of 3D hand pose, in lieu of the classification-based model.

4 Experiments

4.1 Experimental Setup

We evaluated our approach using three public datasets, namely NinaPro dataset [Atzori *et al.*, 2014] (sub-dataset 1), CapgMyo dataset [Geng *et al.*, 2016], and csl-hdemg dataset [Amma *et al.*, 2015]. For all datasets, we linearly transformed the value of sEMG signals to $[0, 255]$.

The NinaPro sub-dataset 1 (DB1) is recorded for the development of hand prostheses. It consists sparse multi-channel sEMG samples of 52 gestures performed by 27 intact subjects. Each sample were recorded at a sampling rate of 100 Hz with 10 sparsely located electrodes placed on upper forearms (forming an image with 1×10 pixels). The signals were filtered and smoothed by the acquisition device. The first 8 components corresponded to the equally spaced electrodes around the forearm at the height of the radiohumeral joint, where the last two components corresponded to electrodes placed on the main activity spots of the flexor digitorum superficialis and the extensor digitorum superficialis, respectively. The 3D hand poses were recorded by a 22-sensor CyberGlove II and synchronized with the sEMG signals.

The CapgMyo dataset consists of high-density sEMG (HD-sEMG) signals, which were recorded at a sampling rate of 1000 Hz using an electrode array with 128 electrodes that covered the upper forearm muscles (forming a grid of 8×16 channels). It consists of 3 sub-databases (DB-a, DB-b and DB-c); The first two sub-databases consist of 8 isometric and isotonic hand gestures obtained from 18 subjects (DB-a) and 10 subjects (DB-b), where the last sub-databases (DB-c) consits of 12 basic fingers movements from 10 subjects. Each subject in DB-b contributed two recording sessions on different days, with an inter-recording interval greater than one week. Each subject performed 10 trials for each gesture.

The csl-hdemg dataset contains HD-sEMG signals of 5 subjects performing 27 finger gestures. Each subject recorded over 5 sessions where 10 trials of each gesture is performed in each session. The sEMG signals were bipolar recorded at a sampling rate of 2048 Hz using an electrode array with 192 electrodes that covered the upper forearm muscles (forming a grid of 7×24 channels).

The deep-learning framework is based on MxNet [Chen *et al.*, 2015]. In all the experiments, the ConvNet was trained using Stochastic Gradient Descent with a batch size of 1000, 28 epoch, and a weight decay of 0.0001. The learning rate started at 0.1 and was divided by 10 after the 16th and 24th epochs. The weights of the ConvNet were initialized as described in [He *et al.*, 2015] when a pre-trained ConvNet was not available. For all experiments that involves a majority voting window, the sliding window advances by one frame.

The following methods are compared in this work. (1) **RF**: Random Forests with a manually designed sEMG feature set [Atzori *et al.*, 2014]. (2) **AtzoriNet** [Atzori *et al.*, 2016]. (3) **GengNet** [Geng *et al.*, 2016]. (4) **T2**: MyoNet with task 2. (5) **T3**: MyoNet with task 3. (6) **T23**: MyoNet with task 2 & 3.

Based on the preliminary experiment, we fixed the temporal distance between two randomly selected frames (δ) in task 2 to be 10 frames (NinaPro DB1), 100 frames (CapgMyo), and 205 frames (csl-hdemg) (equivalent to 10 ms in each dataset). In the experiments on NinaPro DB1, the sub-network of task 2 was branched out from MyoNet at the 4th layer (i.e., Siamese network for task 2 share the first 4 layers with the networks for task 1 and 3). We fix the hyper-parameters of the proposed method in all experiments, where

Table 1: Recognition accuracies (%) of 52 gestures in NinaPro DB1. AtzoriNet used a analysis window of 150 ms and an augmented training set [Atzori *et al.*, 2016].

	RF	AtzoriNet	GengNet	T2	T3	T23
Per-frame			76.1	77.9	77.8	78.1
200 ms	75.3	66.6	77.8	79.4	79.3	79.5

$\alpha = \beta = 1$ and $k = 512$.

The codes are available at `http://zju-capg.org/myo/semi`.

4.2 Evaluation using NinaPro

We followed the evaluation procedure described in previous works [Atzori *et al.*, 2014; Geng *et al.*, 2016]. Our ConvNet was trained with approximately two thirds of the trials of each subject and tested with the remaining one third. It was initialized by pre-training on the union of the training sets of all subjects. The accuracy was calculated as the proportion of correctly recognized frames and averaged over all subjects. As shown in Table 1, the per-frame recognition accuracy of our method is 78.1%, or 2 percentage points improvement over the state-of-the-art [Geng *et al.*, 2016]. Note that the improvement comes without additional runtime cost, because the configurations of GengNet and our MyoNet are the same in the recognition phase. During training phase, the computation load was higher due to the additional unlabeled data and auxiliary networks.

In the above experiments, the gesture labels of all training samples are available (i.e., $\mathcal{U} = \varnothing$). To investigate how many gesture labels are required for stable recognition accuracy, we randomly selected a $1/n$ subset ($n \in \{8, 16, 32, 64, 128\}$) of the training set as $\mathcal{L}$ and used the remaining samples without gesture labels as $\mathcal{U}$ to train the model. As shown in Fig. 3(a), the resulting recognition accuracy achieved by semi-supervised training with $1/32$ labeled data is comparable to the supervised learning counterpart with fully labeled training set. Note that the recognition accuracy is also significantly improved by only predicting temporal order of sEMG frames (i.e., task 2). This suggests that the improvements of accuracy could be achieved almost without additional cost, as the semi-supervised scheme T2 can be easily adapted to any deep learning-based gesture classifiers and doesn't require additional data of other modalities.

We also evaluated different configurations of the weight hyper-parameters and the scheme of task 2 (see Table 2). Here, we denote **T2'** as the network for a regression-based task 2, where a sub-network with two fully-connected layers was trained to predict the difference sEMG image with an instantaneous sEMG image. As shown in Table 2, our method is not sensitive over a wide range of weight hyper-parameters.

Here, we evaluated the impact of four types of hand pose representation for the regression-based task 3: (1) The 22 channels of raw data recorded by data glove; (2) The raw data and its difference in time (44 channels in total); (3) The 19 channels of joint angles (and its difference in time) calculated by mapping the raw data to a 3D virtual hand [Saric, 2011]; (4) The 60 channels of joint positions in the local frame (and its difference in time) calculated by mapping the raw data to the 3D virtual hand. All representations were independently normalized with z-score normalization on data part (Raw) and difference part (Diff). As shown in Table 3, the model based on statistics (clustering) of hand pose outperforms the models that predict vector representations of hand pose, and predicting the temporal changes of hand pose is helpful.

Table 2: Per-frame recognition accuracies (%) of various hyper-parameter settings. The training of T2' (i.e., MyoNet with a regression-based task 2) did not converge when $\alpha = 10$.

Param	Network	0.1	1	10
α	T2	77.6	77.9	77.6
α	T2'	77.4	77.5	Did not converge
β	T3	77.5	77.8	76.8

Table 3: Per-frame recognition accuracies (%) of various hand pose representations based on T3.

T3	Raw	Raw & Diff	Angle	Pos
77.8	77.5	77.6	77.4	77.4

4.3 Evaluation using CapgMyo

Geng *et al.* [2016] only used the static part of the movement to evaluate the recognition algorithms. In other words, only the middle one-second window of each trial (i.e., 1,000 frames of data, about $1/3$ of the entire trial) was used to ensure that no transition movements are included in training and testing. Here we use the remaining $2/3$ of the training data as $\mathcal{U}$ to perform SSL with task 2.

We followed the evaluation procedure as described in the previous study [Geng *et al.*, 2016]. For each subject, a classifier was trained with 50% of the data (i.e., trials 1, 3, 5, 7 and 9 for that subject) and tested on the remaining trials. The ConvNet was initialized by pre-training on the union of the training sets of all subjects. This procedure was performed on each sub-database. For DB-b, the second session of each subject was used for the evaluation. As shown in Table 4, our method achieved per-frame improvements of 0.2, 0.3 and 0.4 percentage points for the three sub-dataset, respectively.

We also evaluated our method with downsampled gesture labels as that in the experiment for NinaPro. As shown in Fig. 3(b-d), the resulting recognition accuracy was improved by semi-supervised training, especially when the labeled samples are very scarce. These results further confirmed that predicting temporal order of sEMG frames also helps gesture recognition with HD-sEMG.

4.4 Evaluation using csl-hdemg

Lastly, we evaluated our method with more gestures (total of 27 finger gestures) in csl-hdemg. We followed the evaluation procedure as described in the previous works [Amma *et al.*, 2015; Geng *et al.*, 2016]. For each recording session, we performed a leave-one-out cross-validation, in which each of the 10 trials was used in turn as the test set and a ConvNet was trained by using the remaining 9 trials. The ConvNet was initialized by pre-training on the union of the training sets of all subjects in each round. The standard evaluation protocol [Amma *et al.*, 2015] used only a small window of

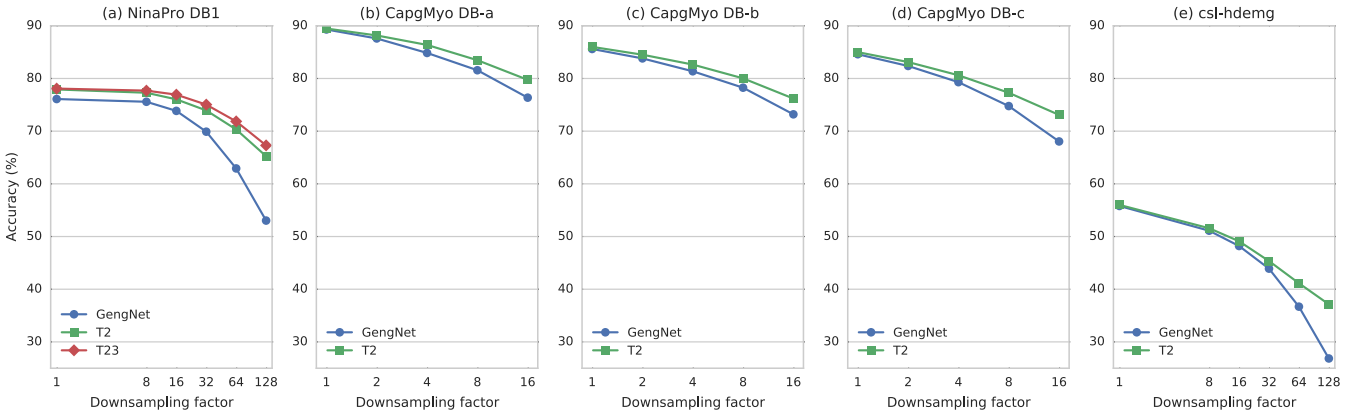

Figure 3: Per-frame recognition accuracies with different downsample factors of labeled training samples.

Table 4: Recognition accuracies (%) on NinaPro, CapgMyo and csl-hdemg datasets. The performance is obtained with majority voting using 200 ms window (i.e., 20 frames) on NinaPro and 150 ms window (i.e., 150 frames) on CapgMyo, where the entire trial were selected for csl-hdemg. Per-frame accuracies are shown in parentheses. * indicates models are trained with 1/16 gesture labels.

	NinaPro	DB-a	DB-b	DB-c	csl-hdemg
GengNet	77.8(76.1)	99.5(89.3)	98.6(85.6)	99.2(84.6)	96.8(55.8)
T2	79.4(77.9)	99.6(89.5)	98.7(85.9)	99.2(85.0)	96.9(56.0)
GengNet*	75.7(73.8)	98.1(76.4)	97.0(73.2)	97.9(68.0)	92.9(48.2)
T2*	77.6(76.0)	98.8(79.7)	97.2(76.2)	98.5(73.1)	94.3(49.1)

each trial, of which the average amplitude of sEMG signals is relatively high, for evaluation. Here we use the unlabeled 512 frames before and 512 frames after the training windows as $\mathcal{U}$ to do semi-supervised learning with task 2.

As shown in Table 4, our method achieved a per-trial recognition accuracy of 96.9%, or 0.1 percentage points improvement, over the latest work [Geng *et al.*, 2016] on csl-hdemg. Compared to NinaPro and CapgMyo, csl-hdemg consists of sEMG signals recorded with much higher sampling rate and higher number of recording sessions for each subject. The standard evaluation protocol used a leave-one-out cross-validation, where each trial was reused 9 times for training. Given that there were sufficient gesture labels in each training set, the performance of GengNet and T2 are almost same. To demonstrate the efficacy of the proposed model, we down-sampled gesture labels as that for the other two datasets. As shown in Fig. 3(e) and Table 4, with a downsample factor of 16, our method achieved a per-trial recognition accuracy (per-frame results shown in parenthesis) of 94.3% (49.1%), an 1.4 (0.9) percentage points improvement over GengNet.

5 Conclusion

Temporal relationship between neighbouring sEMG frames embed information about the underlying muscles firing order, which is invariant with respect to the rate and amplitude of the performed gestures. Furthermore, 3D hand pose describes the fine-grained spatial status of the hand at a specific time. Normally, these information are available without human annotations, which make them an useful supervisory signals for gesture classifier training.

In this work, we proposed using temporal order of sEMG frames and, optionally, statistics of 3D hand pose to learn feature representations of instantaneous sEMG signals. We presented a semi-supervised learning framework with a novel Siamese architecture, which learns a shared ConvNet by predicting the statistics of 3D hand pose with a sEMG frame and predicting the temporal order of the two neighbouring sEMG frames. Our method improves recognition accuracies on both sparse multi-channel sEMG and high-density sEMG, especially when the labeled training data are very scarce.

Unlike typical multi-modal gesture recognition methods that require multi-modality input in the recognition phase, our semi-supervised approach uses multi-modality data only in the training phase, and thus without additional runtime cost. Moreover, our method also works on single-modality training data, thus making it easy to be adapted to any deep learning-based gesture classifiers. In future work, we plan to (1) extend our framework using temporal models (e.g., RNNs) and accommodate dynamic transitional motions, and (2) extend our framework to recognize gestures in inter-session scenario in which the sEMG signals used for training and validation are recorded in different sessions.

Acknowledgments

This work is supported by the National Key Research and Development Program of China (No.2016YFB1001300), the National Natural Science Foundation of China (No.61379067), and the National Research Foundation, Prime Ministers Office, Singapore under its International Research Centre in Singapore Funding Initiative.

References

[Amma *et al.*, 2015] Christoph Amma, Thomas Krings, Jonas Böer, and Tanja Schultz. Advancing muscle-computer interfaces with high-density electromyography. In *SIGCHI*, pages 929–938, 2015.

[Atzori *et al.*, 2012] M Atzori, A Gijsberts, S Heynen, A M Hager, O Deriaz, P van der Smagt, C Castellini, B Caputo, and H Muller. Building the Ninapro database: a resource for the biorobotics community. In *IEEE RAS EMBS International Conference on BioRob*, pages 1258–1265, 2012.

[Atzori *et al.*, 2014] Manfredo Atzori, Arjan Gijsberts, Claudio Castellini, Barbara Caputo, Anne-Gabrielle Mittaz Hager, Simone Elsig, Giorgio Giatsidis, Franco Bassetto, and Henning Müller. Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Scientific Data*, 1, 2014.

[Atzori *et al.*, 2016] Manfredo Atzori, Matteo Cognolato, and Henning Müller. deep learning with convolutional neural networks applied to electromyography data: a resource for the classification of movements for prosthetic hands. *Frontiers in Neurorobotics*, 10, 2016.

[Bengio *et al.*, 2014] Yoshua Bengio, Eric Laufer, Guillaume Alain, and Jason Yosinski. Deep generative stochastic networks trainable by backprop. In *ICML*, 2014.

[Caruana, 1997] Rich Caruana. Multitask learning. *Machine Learning*, 28:41–75, 1997.

[Chen *et al.*, 2015] Tianqi Chen, Mu Li, Yutian Li, Min Lin, Naiyan Wang, Minjie Wang, Tianjun Xiao, Bing Xu, Chiyuan Zhang, and Zheng Zhang. MxNet: a flexible and efficient machine learning library for heterogeneous distributed systems. In *NIPS Workshop*, 2015.

[Cheng *et al.*, 2016] Yanhua Cheng, Xin Zhao, Rui Cai, Zhiwei Li, Kaiqi Huang, and Yong Rui. Semi-supervised multimodal deep learning for RGB-D object recognition. In *IJCAI*, pages 3345–3351, 2016.

[Doersch *et al.*, 2015] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, pages 1422–1430, 2015.

[Farina *et al.*, 2014] Dario Farina, Ning Jiang, Hubertus Rehbaum, Aleš Holobar, Bernhard Graimann, Hans Dietl, and Oskar C Aszmann. The extraction of neural information from the surface EMG for the control of upper-limb prostheses: emerging avenues and challenges. *IEEE Trans. Neural Syst. Rehabil. Eng.*, 22(4):797–809, 2014.

[Fowler, 1983] Carol A Fowler. Converging sources of evidence on spoken and perceived rhythms of speech: cyclic production of vowels in monosyllabic stress feet. *Journal of Experimental Psychology: General*, 112(3):386, 1983.

[Geng *et al.*, 2016] Weidong Geng, Yu Du, Wenguang Jin, Wentao Wei, Yu Hu, and Jiajun Li. Gesture recognition by instantaneous surface EMG images. *Scientific Reports*, 6:36571, 2016.

[Guo *et al.*, 2016] Yuchen Guo, Guiguang Ding, Yue Gao, and Jianmin Wang. Semi-supervised active learning with cross-class sample transfer. In *IJCAI*, 2016.

[He *et al.*, 2015] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In *ICCV*, pages 1026–1034, 2015.

[Ioffe and Szegedy, 2015] Sergey Ioffe and Christian Szegedy. Batch normalization: accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456, 2015.

[Joachims, 2002] Thorsten Joachims. Optimizing search engines using clickthrough data. In *SIGKDD*, 2002.

[Konrad, 2005] Peter Konrad. The ABC of EMG, 2005.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105. 2012.

[Leng *et al.*, 2013] Yan Leng, Xinyan Xu, and Guanghui Qi. Combining active learning and semi-supervised learning to construct SVM classifier. *Knowl.-Based Syst.*, 44:121–131, 2013.

[Mobahi *et al.*, 2009] Hossein Mobahi, Ronan Collobert, and Jason Weston. Deep learning from temporal coherence in video. In *ICML*, pages 737–744, 2009.

[Owens *et al.*, 2016] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *LNCS*, volume 9905, pages 801–816, 2016.

[Pinto *et al.*, 2016] Lerrel Pinto, Dhiraj Gandhi, Yuanfeng Han, Yong-Lae Park, and Abhinav Gupta. The curious robot: learning visual representations via physical interactions. In *LNCS*, volume 9906, pages 3–18, 2016.

[Rasmus *et al.*, 2015] Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko. Semi-supervised learning with ladder networks. In *NIPS*, pages 3546–3554, 2015.

[Saponas *et al.*, 2008] T Scott Saponas, Desney S Tan, Dan Morris, and Ravin Balakrishnan. Demonstrating the feasibility of using forearm electromyography for muscle-computer interfaces. In *SIGCHI*, pages 515–524, 2008.

[Saric, 2011] Marin Saric. Libhand: a library for hand articulation, 2011.

[Shahshahani and Landgrebe, 1994] Behzad M Shahshahani and David A Landgrebe. The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon. *IEEE Trans. Geosci. Remote Sens.*, 32(5):1087–1095, 1994.

[Srivastava *et al.*, 2014] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958, 2014.

[Stewart and Ermon, 2017] Russell Stewart and Stefano Ermon. Label-free supervision of neural networks with physics and domain knowledge. In *IJCAI*, 2017.

[Taigman *et al.*, 2014] Y Taigman, Ming Yang, M Ranzato, and L Wolf. DeepFace: closing the gap to human-level performance in face verification. In *CVPR*, 2014.

[Wang and Gupta, 2015] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *ICCV*, pages 2794–2802, 2015.

[Wang *et al.*, 2016] Xibin Wang, Junhao Wen, Shafiq Alam, Zhuo Jiang, and Yingbo Wu. Semi-supervised learning combining transductive support vector machine with active learning. *Neurocomputing*, 173:1288–1298, 2016.