An incremental cross-modal transfer learning method for gesture interaction

Junpei Zhong [a,*], Jie Li [b], Ahmad Lotfi [c], Peidong Liang [d], Chenguang Yang [e]

[a] *The Hong Kong Polytechnic University, Kowloon, Hong Kong*
[b] *Chongqing Technology and Business University, Chongqing, 400067, China*
[c] *Nottingham Trent University, Nottingham, NG11 8NS, United Kingdom*
[d] *Quanzhou-HIT Research Institute of Engineering and Technology, Quanzhou, 362008, China*
[e] *South China University of Technology, Guangzhou, 510640, China*

ABSTRACT

Gesture can be used as an important way for human–robot interaction, since it is able to give accurate and intuitive instructions to the robots. Various sensors can be used to capture gestures. We apply three different sensors that can provide different modalities in recognizing human gestures. Such data also owns its own statistical properties for the purpose of transfer learning: they own the same labeled data, but both the source and the validation data-sets have their own statistical distributions. To tackle the transfer learning problem across different sensors with such kind of data-sets, we propose a weighting method to adjust the probability distributions of the data, which results in a more faster convergence result. We further apply this method in a broad learning system, which has proven to be efficient to learn with the incremental learning capability. The results show that although these three sensors measure different parts of the body using different technologies, transfer learning is able to find out the weighting correlation among the data-sets. It also suggests that using the proposed transfer learning is able to adjust the data which has different distributions which may be similar to the physical correlation between different parts of the body in the context of giving gestures.

1. Introduction

Recently, robots that collaborate with human users have been deployed widely. Since it has been reported that human workers have unique problem-solving skills and sensory-motor capabilities, human–robot collaboration can compensate the disadvantages of human workers such as their weakness in force and precision. But due to the limited autonomous ability of the robots, instructions, direct mapping, or instant controls, still have to be made by the human users, so it has been suggested that a friendly and easy communication should be developed by less-experienced users. Furthermore, such communication should be intuitive and dis-ambiguous.

Among various approaches about human–robot interaction, using gesture for demonstration or commands is one of the major directions for the future development of human–computer interaction. The way of using gesture to communicate between human and robots has been developed in order to make the communication between human workers and robots more effective and efficient. At present, gesture recognition that can be used for human–computer interaction is mainly based on the detection of physical movements of arms. For instance, using camera(s) is able to detect two-dimensional movements. Two-dimensional gesture recognition basically does not involve any depth information. But it is still enough to solve the simple basic gesture operation for the user using computer vision with basic geographic transformation [1]. The 2-D technology not only recognizes the hand shape, but also recognizes some simple two-dimensional gestures, such as basic interactive gestures with objects such as waving and dragging the camera. This kind of gesture recognition technology needs simple hardware requirements, but it can obtain abundant human–computer interaction content thanks to more advanced computer vision algorithms.

Besides of the gesture recognition based on two-dimensional computer vision, we can also use the 3D skeleton model to accomplish the task (e.g. [2,3]), although it requires a large number of parameters to describe the entity with a skeleton. The skeletal model takes advantage of the angle of the joint and the length of each segment as parameters, thus it greatly improves the accuracy and reduces the amount of calculation. It can be compared with the template database of the skeleton to identify the type of gesture. Since only the key points are used to describe the model, important feature points of the hand can be extracted.

With the development of sensing technologies and computer vision technologies, there are still open challenges for gesture recognition:

- how to process the data in real-time with incremental learning capability;
- how to improve the accuracy of gesture recognition.

All the incremental transfer learning methods should keep a balance of this trade-off. While the first challenge is particularly essential for human-in-the-loop learning for interaction, while the system is able to update the parameters or even the architecture on the run. To guarantee the real-time requirement during incremental learning, some algorithms have to lower the accuracy of the system. In this paper, we proposed a framework and neural learning method for transfer learning among different sensory modalities for gesture recognition. The main contributions are as follows:

- A transfer learning framework across different modalities is proposed. Here, the transfer learning is adopted while we assume these modalities have different data distributions, but all of them have the same output labels.
- We propose a weighting method to off-set the difference in data-distributions of different modalities.
- Finally, we apply the broad learning system in this method which also ensures the incremental learning requirement.

In the rest of this article, we will introduce the related work about using transfer learning in gesture recognition in the next section. The theoretical background about transfer learning using different sensor will be formulated in Section 3. Then we will apply the broad learning system to solve this problem in Section 4. Experiments to examine this method will be introduced in Section 5. The discussions and summaries will be given in Section 6.

2. Related work

2.1. Transfer learning

Transfer learning is a research question about how to fine-tune a model which was previously trained based on a data-set A, for another relevant but different data-set B. Here the term "relevant" means that the two data-sets are having data from the same source of data, share the same data space or having the same labels. But probably they have different distributions. Transfer Learning is particularly useful when the data-set is not big enough to support the training of the model, but it exists another relevant data-set to pre-trained the model (e.g. [4–7]). Another advantage of transfer learning is that people can save computational time for the training process. For instance, in image classification, the most well-studied field in machine learning, the parameters the classification model for the target data-set (which usually has a small number of labeled data) can be fine-tuned using transfer learning. And some parameters of the deep learning model, which usually belong to the lower level of the deep neural network, can be fixed during the fine-tuning. Thus the knowledge obtained during the pre-training by a large number of labeled data set (e.g., ImageNet [8]) can be still stored in the lower level of the neural network.

Domain adaptation is an overlapped but a different field from transfer learning, in the sense that domain adaptation can be seen as a sub-field of transductive transfer learning. In such cases, when the source and target data sets of transfer learning is dealing with the same task (e.g. image classification), the two data sets come from different domains or sources (e.g. different data-sets, or even different sensors). Basically, these two data-sets are from different sources following different marginal distributions. To eliminate such differences, most methods [9–11] uses optimization method to maximize the predicted distributions on the target domain data, by adapting the weights for either the source data set or the target data set. This is realized by comparing the distributions between each source domain and the target domain, and thus to learn a new classifier. This kind of learning can be done by the ensemble method [9], which constructs different source-domain classifiers by estimating the weights from the comparison between the trained classifiers from the source dataset and the target instances. The prediction of the distribution of the target data-set can be also used. This can be approximated by a consensus regularization framework on both data sets [10].

The domain adaptation method may be also used when different sources of inputs, even they have different forms of joint distributions [12]. For example, the multi-source domain adaptation is use to detect levels of fatigue of different subjects [13] by the Surface ElectroMyoGraphy (sEMG) signals. For indoor activity recognition, [14] conducts experiments on multiple devices are used to imply the subject's activities and locations. In this optimization framework, these devices do not necessarily have the same signal distributions. Domain adaptation is a useful to deal while the training data and testing data are with different distributions. But while different batches of training sets are also with different distributions, the training cost is high especially in a time-sensitive scenario.

2.2. Incremental learning

Incremental learning can be also referred to as an adaptive learning algorithm that is capable of learning from a batch of information belonging to different, or even novel classes of data without forgetting the trained knowledge. It can be also called continuous learning or lifelong learning. In general, besides using the techniques of transfer learning where the model parameters are updated with the new data-set, an incremental learning framework should avoid the catastrophic forgetting and to follow the following three properties:

- the model should be able to detect and start to train itself while it receives a novel stream of data which is different from the consolidated data;
- the remaining knowledge should be kept in any forms without forgetting (i.e. catastrophic forgetting) [15].
- during the training process, its computational and memory requirements should remain bounded.

Different from transfer learning, the incremental learning concerns about how to update a model within the limited time and avoid the catastrophic forgetting problem, the over-fitting problem and the computational boundary problems. For instance, the avoid the over-fitting, adding a regularization during learning (e.g. [16,17]), which is the common way to avoid over-fitting, is the most straightforward way to allow incremental learning. By adding different levels of regularization, it provides more plasticity on the synaptic weights during new knowledge comes and, in the meanwhile, still keeps the existing partial representation on some of the weights. This regulation can be applied in the loss function to have the penalties into a large update of the new information [18–20], so as to preserve the previously learned input–output mappings by computing additional activation.

Another methodology in incremental learning focuses on the modification and extension of the network structures, which is similar to the generative learning. They incorporate novelty detection and use different representational resources for new samples. For instance, the growing-like networks [21,22] are able to extend the architectures by allocating novel sub-networks or neurons with a fixed capacity to be trained with the novel information recorded in the new weights. This idea is not only about extension by also subtraction/merging, while the similar features are learnt in redundant connections [23,24]. Therefore, the refined connectionist models and the optimization methods have to be carefully designed to act as the memories or the sub-systems. Some of the learning method can selectively learn such process and form particular sub-systems, which allow both long-time learning (by allocating similar features) and short-term learning (e.g. by growing) [25]. The new-grown units and their parameters are retrained using the group sparse regularization. Although it claims that it allows the incremental learning, the computational cost for such a method could be still demanding.

Therefore, since the multiple requirements the incremental learning are to balance the trade-off between the computational costs and the training performance, a different neural-architecture-optimized approach should be utilized. The optimization method needs to calculate fast so that the incremental variants can easily be formulated, whereby the reservoir-like neurons represents the non-linear mappings of the new data-sets.

2.3. *Multi-modal classification and interaction*

A multi-modal interaction framework usually consists various communication channels between humans and computers using different sensors. It has been developed and used extensively since different sensors can capture various types of information for human–robot interaction. Such information may included torso movement, arm gestures, etc, all of which can be useful for non-verbal communication. The non-verbal communication is as crucial as the verbal communication such as speech, since quite a large portion of interactions, either human–human or human–robot, incorporates both directions of non-verbal communication [26]. And multi-modal human–robot interaction, in the scenarios of human–robot collaboration and tele-operation, has proven to be more efficient than using speech recognition and touch screen commands [27,28].

In both service robots and industrial robots, the non-verbal communication facilitates their ways of interactions in the forms of interactive control and human–robot communication. The interactive control for industrial robots, thanks to the high precision of the novel contact-free sensors, such as LeapMotion and Kinect, there have been increasing tele-operating robotic applications. Particularly, in the case of direct human–robot cooperation or collaboration, the gesture-based user interface is more straightforward and safe (e.g. [29,30]). Using multi-modal signals such as the speech commands, hand gestures as well as body position provides a complementary way to order the robot, thus the users do not have to explicitly tell the instructions [31]. Furthermore, one of the fields where such tele-operation is essential is the medical robot, where we aim to minimize the risk of infections [32]. Besides the usage in robot control, the non-verbal interaction can be also a way for the robots to detect the intention and internal status of the users [33]. The hand gesture and bodily movement interaction is reported the most natural method way for drone navigation in the presented flight-tests in controlled indoor environments.

3. Method

3.1. *Problem formulation*

Suppose there is a domain $\mathcal{D}_s$ which is composed of an n dimensional feature space $\mathcal{X}$. Assume that we have a set of source training data in the $\mathbb{X}$ that comes from the first modality:

$$\mathcal{X}_s = x_s^1, x_s^2, \dots, x_s^N \tag{1}$$

which fulfills a marginal probability distribution $P(x)$.

Similarly, the test set belongs to a domain $\mathcal{D}_t$ that comes from another modality, with the m dimensional feature space:

$$\mathcal{X}_t = x_t^1, x_t^2, \dots, x_t^M \tag{2}$$

where $M \neq N$, since the inputs of source and test sets may differ.

We also assume that there are two separate classifiers in two feature spaces, $\mathbb{C}_s$ and $\mathbb{C}_t$, for the source data-set X_s and the test data-set X_t, respectively.

The output of both X_s and X_t are the same output labels. For example, the same set of commands for interaction:

$$\mathcal{Z}_t = z_t^1, z_t^2, \dots, z_t^O \tag{3}$$

We denote

$$\mathcal{X}_{ls} \xrightarrow{\mathbb{C}_s} z^o \tag{4}$$

and

$$\mathcal{X}_{lt} \xrightarrow{\mathbb{C}_t} z^o \tag{5}$$

where z is a predicted label as an output of the two classifiers, and $z \in \mathcal{Z}$. The given sets are $X_{ls} \in X_s$ and $X_{lt} \in X_t$. $\mathbb{C}_s$ and $\mathbb{C}_s$ indicate the known mappings from the source data-set and the test data-set to the labels, respectively.

We then define the complement of the sets $\mathcal{X}_{ls}$ and $\mathcal{X}_{lt}$ are the unlabeled data $\mathcal{X}_{us}$ and $\mathcal{X}_{ut}$, where

$$\mathcal{X}_s = \mathcal{X}_{us} \cup \mathcal{X}_{ls} \tag{6}$$

and

$$\mathcal{X}_t = \mathcal{X}_{ut} \cup \mathcal{X}_{lt} \tag{7}$$

From this formulation of the background, we can derive the following optimization target based on the labeled and the unlabeled data we have.

3.2. *Optimization target*

In the multi-modal transfer learning applications, given labeled source data-set: $\{\mathcal{X}_{ls}, \mathcal{Z}\} = \{(x_s^1, z^1), \dots, (x_s^t, z^t)\}$. During the incremental learning stage, there comes a few labeled target domain $\{\mathcal{X}_{lt}, \mathcal{Z}\} = \{(x_t^1, z^1), \dots, (x_t^t, z^t)\}$. Since the modality of both source and test sets differ, we also have the assumptions that $P_s(y|x_s) \neq P_t(y|x_t)$ and $P_s(z|x_s) \neq P_t(z|x_t)$, so we have to learn a domain adaptation in which the distribution differences between $P_s(x_s)$ and $P_t(x_t)$.

We then use $\mathcal{H}(\cdot)$, which is the cross entropy loss between the predictions of the model $\mathcal{Z}$ and the ground-truth labels † to represent the minimization target often uses the cross entropy loss between the predictions of the model. Based on the definition of cross entropy, the optimization target thus is:

$$\begin{aligned} \mathcal{J}_w &= \min \mathcal{H}(P(y|x_t), P_t(z_t|x_t)) && (8) \\ &= \min_{\mathbb{T}} \|E_{P(x_s,y)}[\mathbb{T}(x_t), y] - E_{P(x_t,z)}[\mathbb{T}(x_t), z]\|^2 && (9) \\ &\approx \min_{\mathbb{T}} \|E_{P(x_s,z)}[\mathbb{T}(x_s), z] - E_{P(x_t,z)}[\mathbb{T}(x_t), z]\|^2 && (10) \end{aligned}$$

where $\mathbb{T}$ is a function composition of $\mathbb{C}$. The approximation is established according to the assumption that the distribution of predicting a particular label given the source examples is approximately the same as that of the target examples. In general, this transformation function $A = \mathbb{T}^T$ should fulfill the following statistical properties:

- the reconstruction error of the input data should be minimized, which implies that we should maximize the variance of the embedded data, which is closely related to dimension reduction methods such as PCA.

$$\underset{A^T X}{\operatorname{argmax}}\, tr(A^T X H X^T A) \tag{11}$$

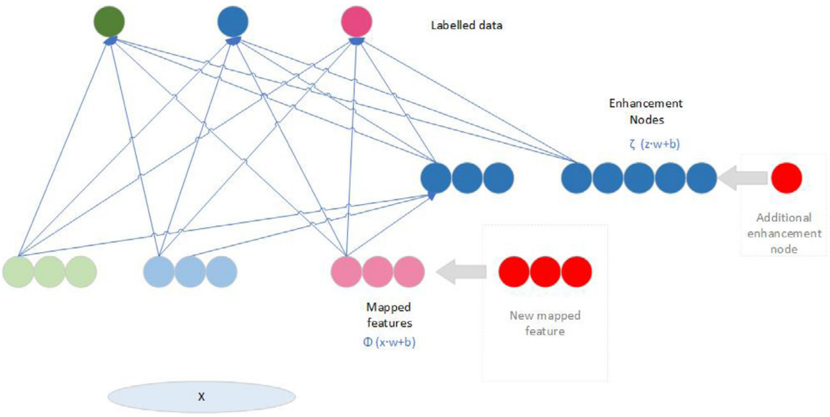


Fig. 1. Broad learning system.

- the marginal distributions difference between different domains $P_s(X_s)$ and $P_t(X_t)$ should be minimized, which implies :

$$\min \| \frac{1}{N}\sum_{i=1}^{N} A^T X_s - \frac{1}{M}\sum_{i=1}^{M} A^T X_t \| \tag{12}$$

3.3. Domain adaptation based on importance weighting

In Eq. (10), note that there is few unlabeled data in the target domain $\mathcal{X}_{ut}$. So the functional optimization problem $P_t(z_t|x_t)$ cannot be estimated exactly.

Remark 1. Data Distribution Adaptation

The optimization target Eq. (10) can be further derived as

$$\mathcal{J}_w = \min_{\mathbb{T}} \|E_{P(x_s,z)}[\mathbb{T}(x_s), z] - E_{P(x_t,z)}[\mathbb{T}(x_t), z]\|^2 \tag{13}$$

$$\approx \min_{T} \|E_{P(x_s)}[\mathbb{T}(x_s)] - E_{P(x_t)}[\mathbb{T}(x_t)]\|^2 + \|E_{\mathbb{C}_s(z|x_s)}[z|\mathbb{T}(x_s)] - E_{\mathbb{C}_t(z|x_t)}[z|\mathbb{T}(x_t)]\|^2 \tag{14}$$

Remark 2. Sample Weighting

However, the distribution of source pre-training data-set D_s may differ from the target data-set D_t. This could be detrimental as the model may emphasize features which are not relevant to the target data-set. We will mitigate this by up-weighting the examples that are most relevant to the target data-set.

$$E_{P(x_2,z)}[\mathbb{T}(x_2), z] = \sum_{x,z} P_t(x, z)\mathcal{L}(f_\phi(x), z) \tag{15}$$

$$= \sum_{x,z} P_s(x, z)\frac{P_t(x, z)}{P_s(x, z)}\mathcal{L}(f_\phi(x), z) \tag{16}$$

$$= \sum_{x,z} P_s(x, z)\frac{P_t(z)P_t(x|z)}{P_s(z)P_s(x|z)}\mathcal{L}(f_\phi(x), z) \tag{17}$$

where we assume the transformation function $\mathbb{T}$ is a function f owning the parameter ϕ.

3.4. Broad learning based transfer learning

Considering the previous problems presented, the trade-off between computational efficiency and the optimization process of the neural structure should be carefully balanced in the incremental learning algorithm. Broad learning has been found to be an efficient optimization method in various applications, so we adopt the broad learning method to do the optimization of aforementioned problems. It is a novel learning architecture which is different from deep learning method. Different from the convolutional networks which delicately learn the features, it uses both randomized weighting features and additional learnable ones which can efficiently encode the inputs. Though the weights are firstly initialized randomly, by only updating the one weighting matrix, it can approximate any non-linear functions. Using those features, we can gradually learn the mapping proposed by the Remark 1 (see Fig. 1). Let x is the raw input of the network. A set of mapped features was first constructed by the weighted multiplication of the inputs

$$a_f = \phi_i(x_i \cdot w_{fi} + b_f) \in \mathcal{A} \tag{18}$$

The first set of mapped feature-sets $Z = \{z_1, z_2, \ldots, z_f\}$ are done by the randomly generated weights. Being different from the usual network, the connecting weight W_{ei} and bias b_{ei} are randomly initialized and fixed afterwards. The non-linear approximation ability of the BLS is realized by the additional enhancement-sets $H = \{h_1, h_2, \ldots, h_j\}$, which maps from the feature-sets Z:

$$e_j = \zeta_j(z_f \cdot w_{jf} + b_j) \in \mathcal{E} \tag{19}$$

In practice, similar as choosing the number of neurons in MLP, the number of i and j should be chosen depend on the size of the data tasks. Assume we have n feature mappings and m groups of enhancement features with each feature mapping and enhancement features generating p nodes and q nodes, respectively. Then the obtained BLS features can be represented as the concatenation of both the features $\mathcal{A}$ and $\mathcal{E}$

$$\mathcal{F} = [\mathcal{A}_n|\mathcal{E}_m] \in \mathbb{R}^{(np+mq)} \tag{20}$$

When we know the labels $\mathcal{Z}$, we have

$$\mathcal{Z} = \mathcal{F} \times W \tag{21}$$

where W are the connecting weights for the network. Also, the original work of BLS [34] proposes that the resulting W should follow the sparse auto-encoder characteristics in order to obtain the efficient representation for all the incoming data.

$$\underset{W}{\text{argmin}} : \|\mathcal{F}W - \mathcal{Z}\|_2^2 + \lambda\|W\|_2^2 \tag{22}$$

where $\lambda\|W\|_2^2$ is the L_2 regularized term.

To solve this problem, we uses an approximation based on a calculation of pseudoinverse. It is a convenient and efficient approach to solve the training problem, with the bounded time and computational requirement:

$$W = (\lambda I + AA^T)^{-1}A^T Y \tag{23}$$

where I is the identity matrix. And it can equivalent to the first problem we proposed (Eq. (14)).

Furthermore, when the incremental learning is necessary, resulting in the new sample the updated feature vector $\mathcal{F}$ becomes

$$\mathcal{F}_{n+1}^{+} = \begin{bmatrix} \mathcal{F}_n^{+} - \mathcal{D}^T \mathcal{Z} \\ \mathcal{B}^T \end{bmatrix} \tag{24}$$

where $\mathcal{D} = \mathcal{F}_n^{+} \cdot f$, and

$$B^T = \begin{cases} C^{+} & C \neq 0 \\ (1 + \mathcal{D}^T \mathcal{D})^{-1} \mathcal{B}^T \mathcal{F} \mathcal{D} & C = 0 \end{cases} \tag{25}$$

where $C = \zeta(\mathcal{A}W + b) - \mathcal{F}\mathcal{D}$.

After all, the new weights becomes:

$$W_{n+1} = \begin{bmatrix} W_n - \mathcal{D}\mathcal{B}^T \mathcal{Z} \\ \mathcal{B}^T \mathcal{Z} \end{bmatrix} \tag{26}$$

As we can see, the update of W should be quite efficient as it only considers the additional value of sample and feature node, which fulfills the third requirement of incremental learning (Section 2.2).

3.5. BLS based multi-modal transfer learning

The proposed BLS based domain adaptation frame contains two stages: (1) BLS feature mapping, (2) output weights learning. In the first stage, we want to generate the BLS features for all samples. We calculate the corresponding features A using the data from source and target domains by the procedure introduced in Section 2. Take the features of X_S, A_S, as example. We first randomly generate W_{ei} and b_{ei} and finetune them using sparse autoencoder with all available training samples X. Then Z_{Si} and H_{Sj} can be calculated using

$$Z_{Si} = \phi_i(X_S \cdot W_{ei} + b_{ei}) \tag{27}$$

$$H_{Sj} = \zeta(Z_{iS} \cdot W_{hj} + b_{hj}) \tag{28}$$

The BLS-SDA aims to learn a classifier using all labeled instances from the source domain, and set this set of very few labeled data from the target domain as a new learning data source.

We adapt the BLS-SDA proposed in [35] but change their way of weighting sample. The optimization function is devised as follows

$$\underset{W}{\text{argmin}} : C_s \cdot \|\mathcal{F}_s W - \mathcal{Z}_s\|_2^2 + C_t \cdot \|\mathcal{F}_t W - \mathcal{Z}_t\|_2^2 + \lambda \|W\|_2^2 \tag{29}$$

We can omit the last regularized term. To determine the values of C_s and C_t, we can compare Eq. (29) with Eq. (17). We can try to cancel out the terms $P_t(z)$ and $P_s(z)$ in Eq. (17), then we obtain

$$E_{P(x_2,z)}[\mathbb{T}(x_2), z] = \sum_{x,z} P_s(x, z) \frac{P_t(x|z)}{P_s(x|z)} \mathcal{L}(f_\phi(x), z) \tag{30}$$

Therefore, using Remark 2, we should have

$$\frac{C_s}{C_t} = \frac{P_t(x|z)}{P_s(x|z)} \tag{31}$$

which means that we can obtain the weights by calculating the proportion of the distributions of the labels. By doing this, probability of a source sample can be adjusted to reflect the probability under the target distribution.

4. Experiments

In this section, we will apply the algorithm obtained from last section in the scenario of recognizing gestures based on sensors with different modalities. Specifically, we incorporate three relevant modalities of gestures: the depth information of arm movement, the electrical activity of muscles and the movements of fingers. Three types of sensors are also used to measure the data of these modalities, which we will introduce in details below.

4.1. Experimental setting

We select the capture the hand and arm gestures data from different modalities, while the user is trying to give specific instructions to the computer/robot. The three devices we used are: Kinect, Leap Motion and Myo Armband sensors. We will briefly introduce their specifications and working principles.

Kinect is a collection of a infrared (IR) projector, an inexpensive depth sensor which receives IR signals, a color camera and a microphone. It is originally developed as a device for human–computer interaction, so it is brought with the Microsoft software development kit. Based on these sensors, it can be used to capture full-body skeletal motion, facial recognition, and voice recognition. In our experiment, we only use the IR depth sensor to track the arm movement. The depth information is captured by the IR projector which exerts the infrared and the monochrome CMOS sensor, which reconstructs the depth information by capturing the IR beams.

Similar as Kinect, **Leap Motion** is an interactive hardware device based on IR sensors. It can precisely measure the hand and finger movements by IR sensors. But different from the Kinect sensor, the Leap Motion is specifically designed to detect and track human hand-gestures, so the error of tracking is about 200 μm about the 3D coordinate of fingertips [36]. It can accurately capture and extract the angles of 14 finger-joints, and their relative positions to the palm.

Being relevant to the body motion, the Electromyography (EMG) evaluates and records the electrical activity produced by skeletal muscles. The EMG signals can be measured by devices worn on the arm. The EMG devices can be used in either medical use or consumer use. For example, **the Myo armband** is usually used for human–computer interaction by detecting the EMG signal of the forearm. It contains 8 channels and it can identify what kinds of the arm gesture by indirectly detecting which muscles are in contraction (see Fig. 2).

4.2. Experimental results

In this section, we compare the performances of the proposed method and other two transfer learning methods. The targets of the experiments is to examine the proposed method in the context of transferring data between two modalities. Since we have three kinds of sensors with three modalities, we will discuss them in the following three sub-sections. The pre-processing is done as follows:

1. the sampling rate of all the data from three modalities of sensors has been unified to 50 Hz. This can be done by comparing the actual sampling rate and select the corresponding data points.
2. The invalid gesture data which is out of range has been deleted.
3. Since the number of dimensions of different modalities differs, we manually duplicate the last dimension of data and add some more dimension(s). As such, we could align them to be the same number of dimensions.

(a) Kinect 2.0 (b) One type of sEMG sensors (c) Leap Motion Sensor

Fig. 2. Three types of sensors.

To compare the performances of transfer learning, the following data-sets are used:

- The database [37] provided by Marin et al. which includes the gesture information of both Kinect (Depth Sensor) and Leap Motion sensors (Finger). (Depth ↔ Finger)
- The multi-modal data-set [38] from Wang et al. includes signals from multiple EMG sensors, Kinect and the Vicon tracker. We will use the EMG and the Kinect sensors in the following experiments. (EMG ↔ Depth)
- Since we have not found an open-source data-set incorporate both LeapMotion and the EMG sensors, we collect the data by recruiting demonstrators to present the instructions to the robots.

We compare the performances of our proposed method and similar methods proposed in [39–42]. The results of all the 6 experiments about the accuracy (of the validation set) and the running time are shown in Tables 1 and 2. The error and the time are obtained via 20 trials to eliminate the differences of initialization. As we can see from the tables, in most tests, our proposed method can achieve similar or better performance than the others. Among the four selected work, some results from [42] are better than the proposed result, but our method takes less time.

4.3. *Result analysis*

In this subsection, we discuss the results regarding different modalities and how do the results related to the physical correlations of these sensors. We will compare the accuracy and the incremental learning capability of the proposed model with other 4 models.

4.3.1. *Arm Movement ↔ Finger*

In [37], the gesture recognition is realized by both Kinect and the Leap Motion sensors. Although Kinect and Leap Motion do not exactly track the same parts of the human body, the author claim that they have complementary characteristics. The features about 3-dimensional hand positions, hand orientation and the coordinate of the hand center are already extracted in the Leap Motion. Particularly, in this data-set, no color information but the depth information is recorded from the Kinect sensor. The two sensors are calibrated as [37] did. In our experiment, different from the authors did in [37], we put the inputs of the original features from either the Leap Motion or the Kinect. Then we train the classifiers. After that, we adjust the weights according to Eq. (31). The outputs of the classifiers are also the ten gestures from American Sign Language (ASL) data-set. The training and accuracy curves are shown in Fig. 3. After the comparison between Figs. 3(a) and 3(b), it seems that the transfer learning from fingers to arm movements are easier than vise versa, which is quite close to our intuition: it is easier to guess the arm movement, which is less complicated, given the finger gestures.

4.3.2. *Muscle ↔ Arm movement*

The EV-action data-set collected in [38] includes Kinect, electromyography and Vicon sensors. Compared with other similar data-sets about actions of the whole human body, this data-set is claimed to be the most accurate and comprehensive one. We only utilized the former two sensors to do the comparison, in order to correspond to the previous data-set. And we only use the first part ("person-individual") of the data-set to do the transfer learning. But different from previous experiment in which only one EMG (Myo) sensor is used, four sensors are used on the arms of the subject. They are attached to the middle of each forearm and the shank muscles. The Fig. 4(b) shows the error and the accuracy while we use our proposed method for training and validation. As we can observe, to learn the mapping between muscle EMG signals and the actions are quite challenge to the system, especially at the beginning of the iterations. But the BLS system perform the incremental learning quite well and converges fast after 500 interactions.

4.3.3. *Muscle ↔ Finger*

Similar the previous two experiments on transfer learning, we would like to construct the data-set incorporating the gesture captured by both EMG and Leap Motion sensors. To our best knowledge, such kind of data-set has not been revealed yet. Therefore, using 10 gestures from the ASL, similar as the experiment 1, we recorded the readings from the Myo and Leap Motion sensors. As we can see there is a larger gap between the training and the validation curves than our previous experiment. This is because the correlation between the muscle and finger movements are not significant, so the algorithm should incrementally learn the distribution of the labeled data. But validation error eventually converges around 700 and 800 iterations. We will discuss the physical correlation among these modalities in the next section (see Fig. 5).

4.3.4. *Incremental learning*

In this subsection, similar as the [34], we increase the dynamical structure to test its incremental learning capability. The following three structures are changed and examined: (1) the feature nodes; (2) the corresponding enhancement nodes; and (3) the additional enhancement nodes. We are tested similar cross-modal scenarios: the incremental learning among the three sensory modalities are examined.

The setting of changing the structure is shown in Tabs. At first, the network is initially number set to have 10 × 9 feature nodes and 7,000 enhancement nodes at the beginning of the incremental learning. Since the dimensions of sensory inputs differ, the following common rules are followed:

Table 1
Validation error of three transfer learning methods.

Methods	A ← F	A → F	A ← M	A → M	F ← M	F → M
Base-line	0.542	0.642	0.691	0.535	0.458	0.414
Our method	0.752	**0.748**	**0.724**	0.646	0.567	**0.581**
2SW-MDA [39]	0.636	0.621	0.712	**0.741**	0.585	0.521
ITL-KRR [40]	0.744	0.729	0.601	0.568	0.502	0.567
SWiRN [41]	**0.797**	0.585	0.589	0.653	**0.776**	0.501
DSL-GSDA [42]	0.684	0.649	0.658	0.599	0.606	0.385

Table 2
Elapsed time of three transfer learning methods (in seconds).

Methods	A ← F	A → F	A ← M	A → M	F ← M	F → M
Base-line	320.3	301.7	1124.4	901.6	681.4	610.6
Our method	**165.5**	**147.8**	206.7	**272.4**	**217.8**	**266.4**
2SW-MDA [39]	193.7	206.7	247.2	324.5	336.1	306.6
ITL-KRR [40]	224.8	215.3	523.7	549.2	522.6	598.3
SWiRN [41]	230.4	302.4	**200.3**	305.5	234.6	301.2
DSL-GSDA [42]	319.7	453.3	302.4	492.9	398.1	284.5

Table 3
Test of incremental learning.

Method	Modalities	No. of feature nodes	No. of enhancement nodes (×1, 000)	Testing accuracy	Training time (s)	Testing time (s)
BL	A ← F	130	130	0.76	162.42	1.54
IBL	A ← F	70 → 90	70 → 90	0.54	72.32	0.92
IBL	A ← F	90 → 110	90 → 110	0.60	93.58	1.35
IBL	A ← F	110 → 130	110 → 130	0.72	156.24	1.48
BL	A → F	150	150	0.75	153.90	1.50
IBL	A → F	90 → 110	90 → 110	0.41	98.19	0.98
IBL	A → F	110 → 130	110 → 130	0.65	135.78	1.27
IBL	A → F	130 → 150	130 → 150	0.76	164.32	1.61
BL	A ← M	100	100	0.64	105.19	1.79
IBL	A ← M	40 → 60	40 → 60	0.45	91.36	0.68
IBL	A ← M	60 → 80	60 → 80	0.58	98.01	1.56
IBL	A ← M	80 → 100	80 → 100	0.70	103.91	1.85
BL	A → M	150	150	0.65	298.23	1.87
IBL	A → M	90 → 110	90 → 110	0.41	187.84	0.93
IBL	A → M	110 → 130	110 → 130	0.54	229.59	1.22
IBL	A → M	130 → 150	130 → 150	0.68	275.87	1.88
BL	F ← M	100	100	0.55	128.21	1.54
IBL	F ← M	40 → 60	40 → 60	0.41	60.92	1.09
IBL	F ← M	60 → 80	60 → 80	0.49	93.83	1.38
IBL	F ← M	80 → 100	80 → 100	0.57	122.29	1.61
BL	F → M	130	130	0.57	276.91	1.64
IBL	F → M	70 → 90	70 → 90	0.38	190.66	0.99
IBL	F → M	90 → 110	90 → 110	0.40	233.52	1.38
IBL	F → M	110 → 130	110 → 130	0.71	267.68	1.70

1. the initial numbers of feature nodes vary and they depend on the sensory inputs;
2. the feature nodes are increased from their initial value at the step of 20, until 100% of its initial value;
3. the corresponding enhancement nodes for the additional feature are increased 250 each, and the additional enhancement nodes are increased at 750 each.

The training time and results of each update are presented in Table 3. We can observe that the incremental learning can be done with increasing number of feature and enhancement nodes during training. But the initial training results are also acceptable.

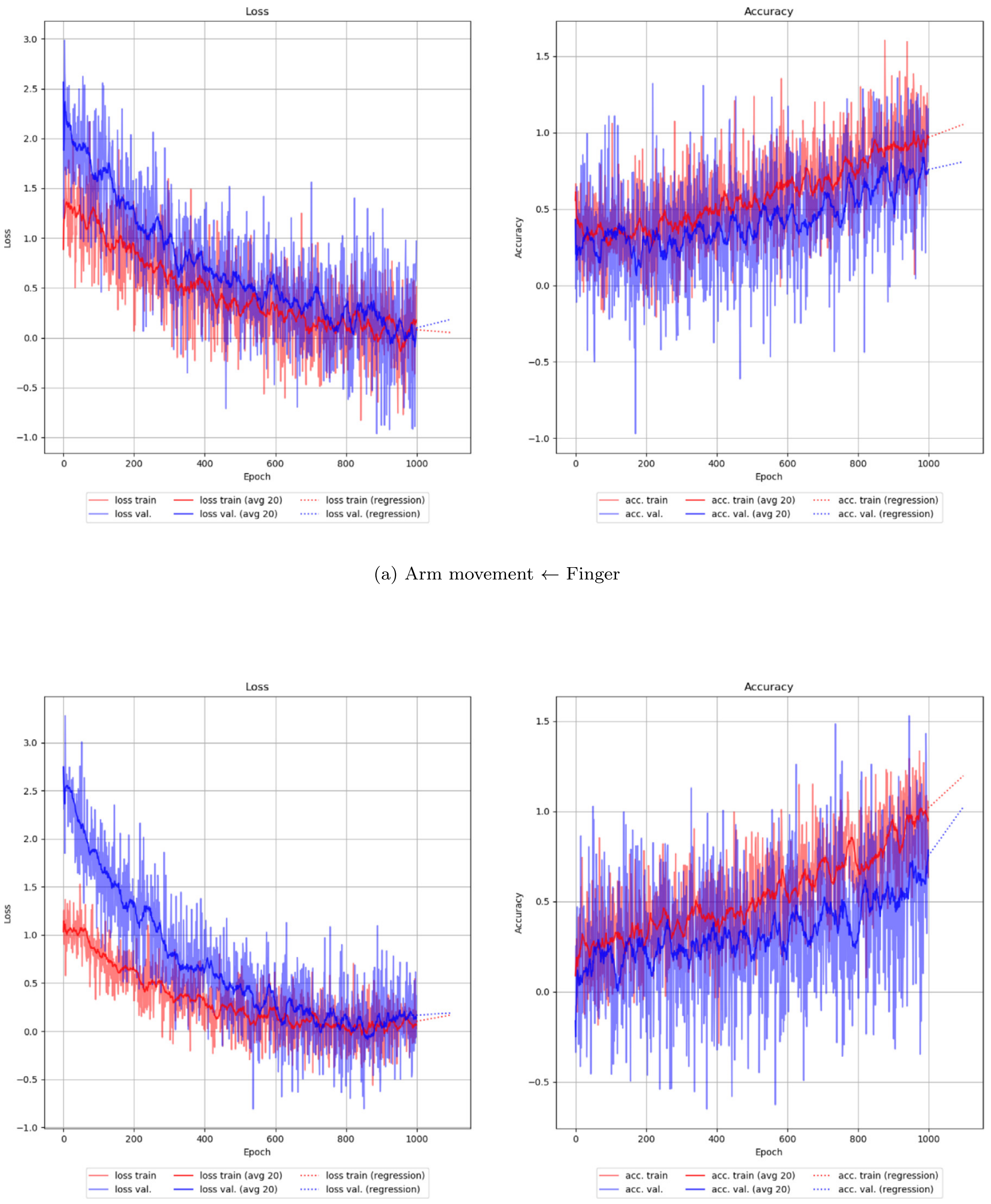
5. Discussions

5.1. Cross-sensor transfer learning

In this paper, we adopt the transfer learning in different sensors. As such, the model trained from the data captured from one sensor, can be used to classify data from another sensor. Besides, the two sensors have different modalities. To our best knowledge, this is quite a novel area in the sense that no much research has been focused on this, besides of [43], who solved the activity recognition using different sensors. But it also held the assumption that the sensors still share the same feature space. We propose a more complicated situation in this paper. But for the future development, we would like to emphasize that there are a few constraints and problems on this research topic:

1. the sensors have differences in both sensory modalities and statistical modalities.
2. the data alignment in both spatial and temporal domains is also a challenge.

To tackle the first problem, it can be divided as two subproblems: (1) using statistical methods, such as, we can do sensor fusion to align and eliminate the uncertainties when the difference in statistical multi-modalities occurs [44]; (2) to tackle



(a) Arm movement ← Finger

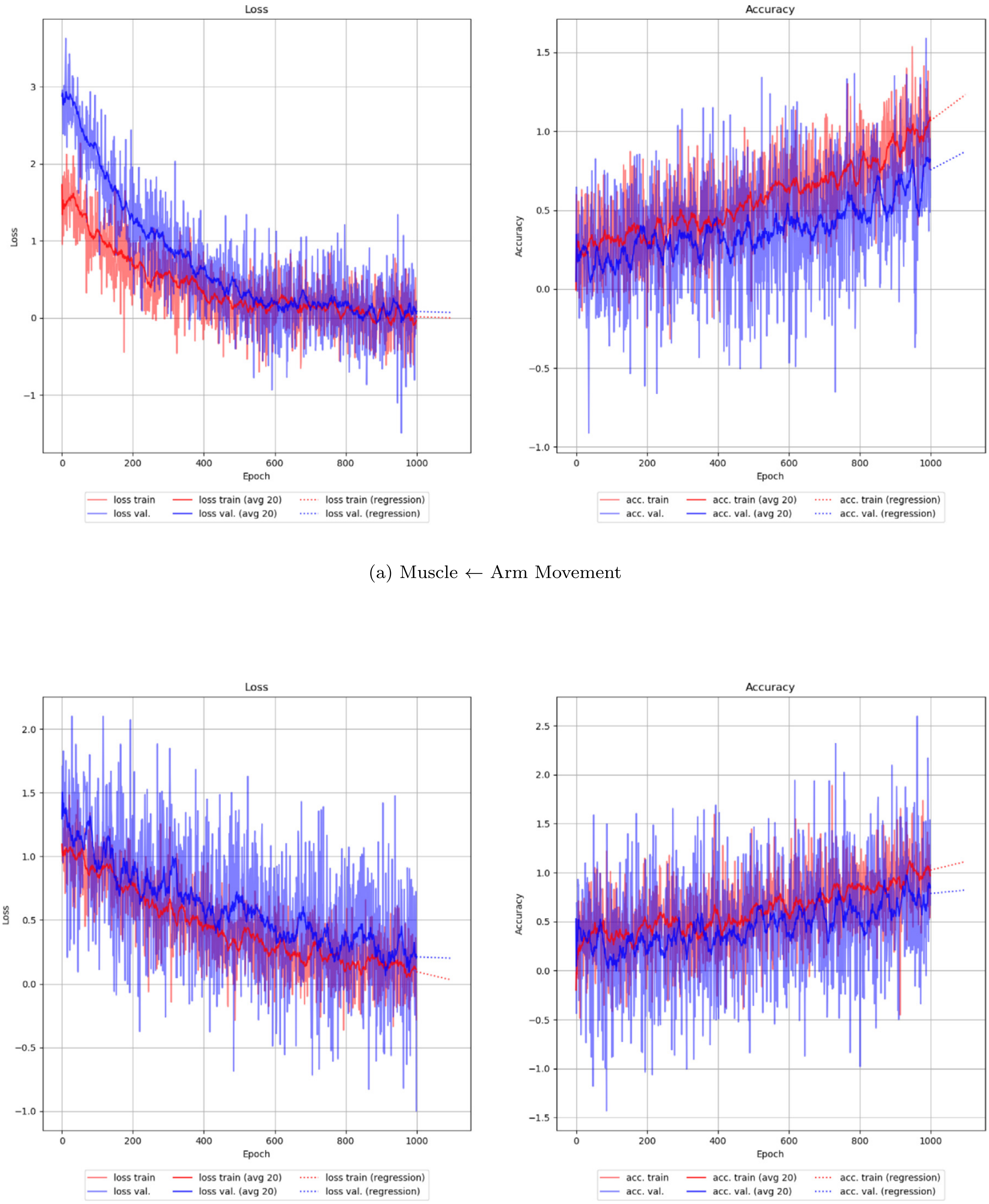
(b) Arm movement → Finger

Fig. 3. The loss and accuracy curves between while doing transfer learning between “Arm movement” and “Finger”.

the differences in sensory modalities, firstly we should convert the signals into the same form (e.g. discrete/continuous, sampling rate, etc.), after which we can also use the same statistical method as previously introduced. To solve the second problem, we should utilize it as the pre-stage of the alignment problem. This problem can be solved with manifold alignment [45], phase correlation [46], etc. On the other hand, we can also solve it with technical methods such as centralized servers and low-latency network.

5.2. Hand and arm movements

In this paper, we utilize the gesture data-sets to compare our proposed methods and other transfer learning methods. The data was captured by three modalities, which focus on different parts of the gesture movements using different principles of measurements (e.g. EMG signal from the muscle, finger movements and arm movements). Nevertheless, the transfer learning methods



(a) Muscle ← Arm Movement

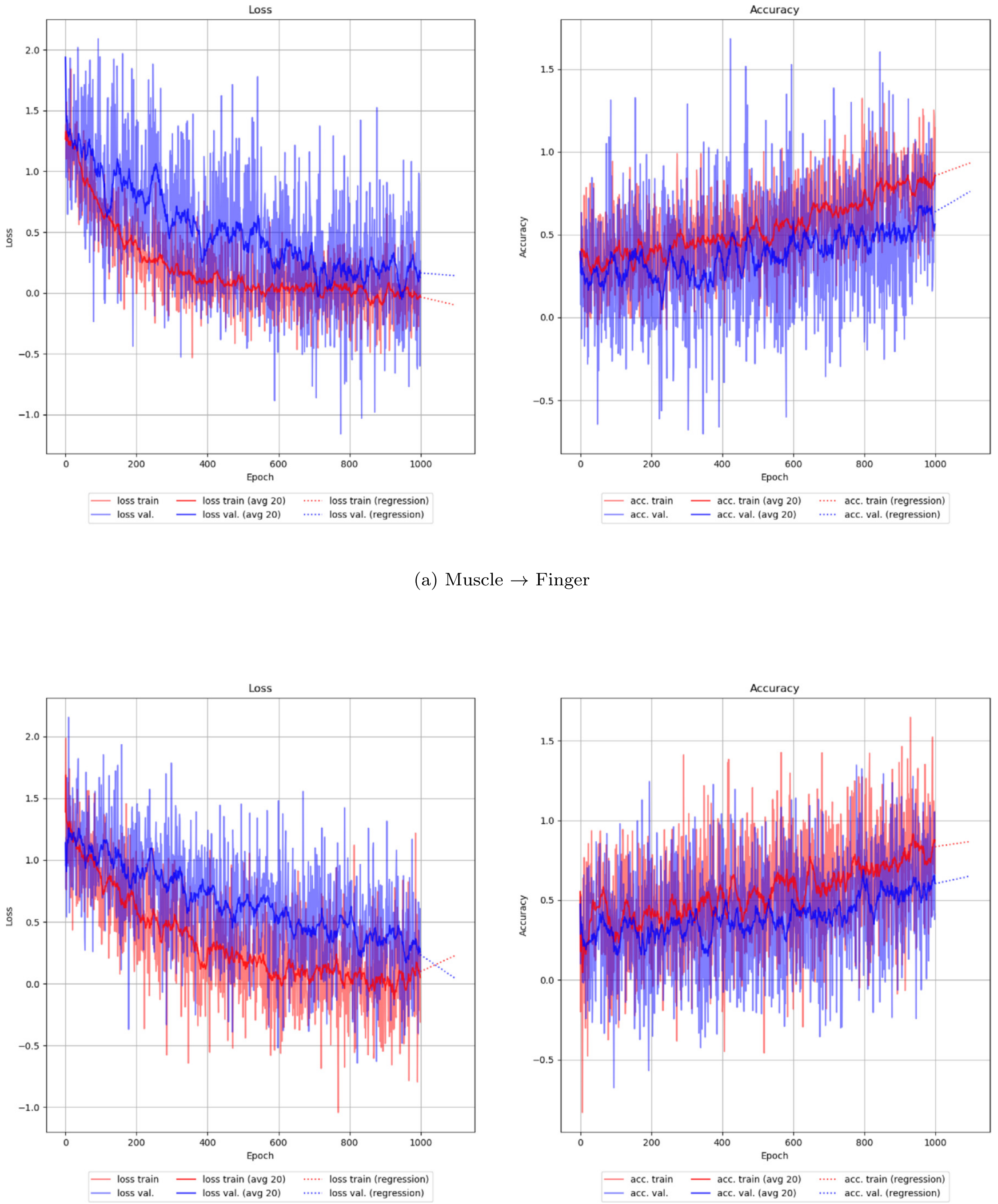
(b) Muscle → Arm Movement

Fig. 4. The loss and accuracy curves between while doing transfer learning between "Muscle" and "Arm Movement".

still work quite well, which exceed the original setting of the transfer learning framework.

In the original setting of transfer learning, applications have been developed to finish tasks such as image recognition. ImageNet [47] has been widely used to pre-train the models which are lately used to classify images which are not included in the data-set. For instance, medical images [48], person detection [49] and action recognition [50]. The pre-training using ImageNet results in the learning in the visual features on the lower levels of the models, which is similar as the biological vision systems [51]. And such pre-training even does not bring significant bias to the results [52].

Our setting for transfer learning is a bit different, in the sense that the training set of data and testing set of data are focusing on different modality of signal. Besides of that, the two modalities are from different parts of the human body, i.e. the original dynamics and primitives of movements are not totally identical



(a) Muscle → Finger

(b) Muscle ← Finger

Fig. 5. The loss and accuracy curves between while doing transfer learning between "Muscle" and "Finger".

as the basic visual features do. Nevertheless, the transfer learning still works, which can be explained as follows:

1. for the *Muscle* ↔ *Finger* transfer learning, study by [53] has pointed out that the sEMG signal of forearms can be identified while the fingers are in movements.
2. for the *Arm Movement* ↔ *Finger* transfer learning. In general, the human gesture system and the ASL system include both the arm and finger movements, in a cross-culture manner [54], although some evidences have suggested that the finger may be involved more than the arm in the terms of its utterance expression [55]. Therefore, it makes a lot of sense that the gestures we used in our experiments that represent some semantic meaning do not only involve the fingers, but the arm movements as well. Such a relationship seems not as obvious as the one between muscle and the fingers, from the view of the performances of the transfer learning. It can still be used as the source data-set mutually for transfer learning.

6. Conclusions

In the context of transfer learning with different modalities for gesture learning, the characteristics of data differ from modalities although they own the same labeled data. Such a difference is caused because different sensors with different technologies are adopted to capture different parts of the bodily gestures. In this paper, we propose to use a weighting method together with the broad learning system, to endow an incremental and more accurate transfer learning method. Experimental results show that such a system is able to balance the trade-off between accuracy and the efficiency in the recognition results.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been partially supported by the Start-up Fund for RAPs under the Strategic Hiring Scheme of PolyU *P*0035417.

References

[1] G. Ushaw, E. Ziogas, J.A. Eyre, G. Morgan, An efficient application of gesture recognition from a 2D camera for rehabilitation of patients with impaired dexterity, in: HEALTHINF, 2013, pp. 315–318.

[2] C. Keskin, A. Erkan, L. Akarun, Real time hand tracking and 3d gesture recognition for interactive interfaces using hmm, in: ICANN/ICONIPP, 2003, Springer Berlin, Germany, 2003, pp. 26–29.

[3] S. Lang, M. Block, R. Rojas, Sign language recognition using kinect, in: International Conference on Artificial Intelligence and Soft Computing, Springer, 2012, pp. 394–402.

[4] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2009) 1345–1359.

[5] H. Larochelle, D. Erhan, Y. Bengio, Zero-data learning of new tasks, in: AAAI. Vol. 1, (2) 2008, p. 3.

[6] X. Glorot, A. Bordes, Y. Bengio, Domain adaptation for large-scale sentiment classification: A deep learning approach, in: Proceedings of the 28th International Conference on Machine Learning, ICML-11, 2011, pp. 513–520.

[7] N. Patricia, B. Caputo, Learning to learn, from transfer learning to domain adaptation: A unifying perspective, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1442–1449.

[8] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252.

[9] J. Gao, W. Fan, J. Jiang, J. Han, Knowledge transfer via multiple model local structure mapping, in: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2008, pp. 283–291.

[10] F. Zhuang, P. Luo, H. Xiong, Y. Xiong, Q. He, Z. Shi, Cross-domain learning from multiple sources: A consensus regularization perspective, IEEE Trans. Knowl. Data Eng. 22 (12) (2009) 1664–1678.

[11] L. Duan, I.W. Tsang, D. Xu, T.-S. Chua, Domain adaptation from multiple sources via auxiliary classifiers, in: Proceedings of the 26th Annual International Conference on Machine Learning, ACM, 2009, pp. 289–296.

[12] J. Jiang, A literature survey on domain adaptation of statistical classifiers, Vol. 3, (1–12) Citeseer, 2008, p. 3, URL: http://sifaka.cs.uiuc.edu/jiang4/domainadaptation/survey.

[13] R. Chattopadhyay, Q. Sun, W. Fan, I. Davidson, S. Panchanathan, J. Ye, Multisource domain adaptation and its application to early detection of fatigue, ACM Trans. Knowl. Discov. Data (TKDD) 6 (4) (2012) 18.

[14] V.W. Zheng, S.J. Pan, Q. Yang, J.J. Pan, Transferring multi-device localization models using latent multi-task learning, in: AAAI, Vol. 8, 2008, pp. 1427–1432.

[15] M. McCloskey, N.J. Cohen, Catastrophic interference in connectionist networks: The sequential learning problem, in: Psychology of Learning and Motivation, Vol. 24, Elsevier, 1989, pp. 109–165.

[16] H.Y. Xiong, Y. Barash, B.J. Frey, BayesIan prediction of tissue-regulated splicing using RNA sequence and cellular context, Bioinformatics 27 (18) (2011) 2554–2562.

[17] W. Zaremba, I. Sutskever, O. Vinyals, Recurrent neural network regularization, 2014, arXiv preprint arXiv:1409.2329.

[18] H. Jung, J. Ju, M. Jung, J. Kim, Less-forgetting learning in deep neural networks, 2016, arXiv preprint arXiv:1607.00122.

[19] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A.A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al., Overcoming catastrophic forgetting in neural networks, Proc. Natl. Acad. Sci. 114 (13) (2017) 3521–3526.

[20] F. Zenke, B. Poole, S. Ganguli, Continual learning through synaptic intelligence, in: Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 3987–3995.

[21] A.A. Rusu, N.C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, R. Hadsell, Progressive neural networks, 2016, arXiv preprint arXiv:1606.04671.

[22] C. Cortes, X. Gonzalvo, V. Kuznetsov, M. Mohri, S. Yang, Adanet: Adaptive structural learning of artificial neural networks, in: Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 874–883.

[23] J.L. Part, O. Lemon, Incremental on-line learning of object classes using a combination of self-organizing incremental neural networks and deep convolutional neural networks, in: Workshop on Bio-Inspired Social Robot Learning in Home Scenarios (IROS), Daejeon, Korea, 2016.

[24] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, C.H. Lampert, Icarl: Incremental classifier and representation learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2001–2010.

[25] J. Yoon, E. Yang, J. Lee, S.J. Hwang, Lifelong learning with dynamically expandable networks, 2017, arXiv preprint arXiv:1708.01547.

[26] M. Argyle, Non-Verbal Communication in Human Social Interaction, Cambridge U. Press, 1972.

[27] S. Gauglitz, C. Lee, M. Turk, T. Höllerer, Integrating the physical environment into mobile remote collaboration, in: Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile Devices and Services, ACM, 2012, pp. 241–250.

[28] S. Lackey, D. Barber, L. Reinerman, N.I. Badler, I. Hudson, Defining next-generation multi-modal communication in human robot interaction, in: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, Vol. 55, (1) SAGE Publications Sage CA, Los Angeles, CA, 2011, pp. 461–464.

[29] Y. Xu, C. Yang, J. Zhong, N. Wang, L. Zhao, Robot teaching by tele-operation based on visual interaction and extreme learning machine, Neurocomputing 275 (2018) 2093–2103.

[30] D. Bassily, C. Georgoulas, J. Guettler, T. Linner, T. Bock, Intuitive and adaptive robotic arm manipulation using the leap motion controller, in: ISR/Robotik 2014; 41st International Symposium on Robotics, VDE, 2014, pp. 1–7.

[31] R.A.S. Fernandez, J.L. Sanchez-Lopez, C. Sampedro, H. Bavle, M. Molina, P. Campoy, Natural user interfaces for human-drone multi-modal interaction, in: 2016 International Conference on Unmanned Aircraft Systems, ICUAS, IEEE, 2016, pp. 1013–1022.

[32] T. Travaglini, P. Swaney, K.D. Weaver, R. Webster III, Initial experiments with the leap motion as a user interface in robotic endonasal surgery, in: Robotics and Mechatronics, Springer, 2016, pp. 171–179.

[33] I.D. Addo, S.I. Ahamed, Applying affective feedback to reinforcement learning in ZOEI, a comic humanoid robot, in: The 23rd IEEE International Symposium on Robot and Human Interactive Communication, IEEE, 2014, pp. 423–428.

[34] C.P. Chen, Z. Liu, Broad learning system: An effective and efficient incremental learning system without the need for deep architecture, IEEE Trans. Neural Netw. Learn. Syst. 29 (1) (2017) 10–24.

[35] L. Yang, S. Song, C.P. Chen, Transductive transfer learning based on broad learning system, in: 2018 IEEE International Conference on Systems, Man, and Cybernetics, SMC, IEEE, 2018, pp. 912–917.

[36] F. Weichert, D. Bachmann, B. Rudak, D. Fisseler, Analysis of the accuracy and robustness of the leap motion controller, Sensors 13 (5) (2013) 6380–6393.

[37] G. Marin, F. Dominio, P. Zanuttigh, Hand gesture recognition with jointly calibrated leap motion and depth sensor, Multimedia Tools Appl. 75 (22) (2016) 14991–15015.

[38] L. Wang, B. Sun, J. Robinson, T. Jing, Y. Fu, EV-action: Electromyography-vision multi-modal action dataset, 2019, arXiv preprint arXiv:1904.12602.

[39] Q. Sun, R. Chattopadhyay, S. Panchanathan, J. Ye, A two-stage weighting framework for multi-source domain adaptation, in: Advances in Neural Information Processing Systems, 2011, pp. 505–513.

[40] J. Garcke, T. Vanck, Importance weighted inductive transfer learning for regression, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2014, pp. 466–481.

[41] X. Zhang, T. Xu, W. Sun, A. Song, Multiple source domain adaptation in micro-expression recognition, J. Ambient Intell. Humaniz. Comput. 12 (8) (2021) 8371–8386.

[42] E. Gholenji, J. Tahmoresnezhad, Joint discriminative subspace and distribution adaptation for unsupervised domain adaptation, Appl. Intell. 50 (7) (2020) 2050–2066.

[43] D.H. Hu, Q. Yang, Transfer learning for activity recognition via sensor mapping, in: Twenty-Second International Joint Conference on Artificial Intelligence, 2011.

[44] M. Kokar, K. Kim, Review of multisensor data fusion architectures and techniques, in: Proceedings of 8th IEEE International Symposium on Intelligent Control, IEEE, 1993, pp. 261–266.

[45] C. Wang, S. Mahadevan, Manifold alignment without correspondence, in: Twenty-First International Joint Conference on Artificial Intelligence, 2009.

[46] C. Dai, Y. Zheng, X. Li, Accurate video alignment using phase correlation, IEEE Signal Process. Lett. 13 (12) (2006) 737–740.

[47] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, Ieee, 2009, pp. 248–255.

[48] A. Menegola, M. Fornaciali, R. Pires, F.V. Bittencourt, S. Avila, E. Valle, Knowledge transfer for melanoma screening with deep learning, in: 2017 IEEE 14th International Symposium on Biomedical Imaging, ISBI 2017, IEEE, 2017, pp. 297–300.

[49] M. Geng, Y. Wang, T. Xiang, Y. Tian, Deep transfer learning for person re-identification, 2016, arXiv preprint arXiv:1611.05244.

[50] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos, in: Advances in Neural Information Processing Systems, 2014, pp. 568–576.

[51] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.

[52] M. Huh, P. Agrawal, A.A. Efros, What makes ImageNet good for transfer learning? 2016, arXiv preprint arXiv:1608.08614.

[53] M. Gazzoni, N. Celadon, D. Mastrapasqua, M. Paleari, V. Margaria, P. Ariano, Quantifying forearm muscle activity during wrist and finger movements by means of multi-channel electromyography, PLoS One 9 (10) (2014) e109943.

[54] J. Blake, G. Vitale, P. Osborne, E. Olshansky, A cross-cultural comparison of communicative gestures in human infants during the transition to language, Gesture 5 (1) (2005) 201–217.

[55] S. Tanaka, T. Inui, Cortical involvement for action imitation of hand/arm postures versus finger configurations: an fMRI study, Neuroreport 13 (13) (2002) 1599–1602.