Transfer Learning on Electromyography (EMG) Tasks: Approaches and Beyond

Di Wu, Jie Yang, *Member, IEEE*, and Mohamad Sawan, *Fellow, IEEE*

***Abstract*— Machine learning on electromyography (EMG) has recently achieved remarkable success on various tasks, while such success relies heavily on the assumption that the training and future data must be of the same data distribution. However, this assumption may not hold in many real-world applications. Model calibration is required via data re-collection and label annotation, which is generally very expensive and time-consuming. To address this issue, transfer learning (TL), which aims to improve target learners' performance by transferring knowledge from related source domains, is emerging as a new paradigm to reduce the amount of calibration effort. This survey assesses the eligibility of more than fifty published peer-reviewed representative transfer learning approaches for EMG applications. Unlike previous surveys on purely transfer learning or EMG-based machine learning, this survey aims to provide insight into the biological foundations of existing transfer learning methods on EMG-related analysis. Specifically, we first introduce the muscles' physiological structure, the EMG generating mechanism, and the recording of EMG to provide biological insights behind existing transfer learning approaches. Further, we categorize existing research endeavors into data based, model based, training scheme based, and adversarial based. This survey systematically summarizes and categorizes existing transfer learning approaches for EMG related machine learning applications. In addition, we discuss possible drawbacks of existing works and point out the future direction of better EMG transfer learning algorithms to enhance practicality for real-world applications.**

***Index Terms*— Transfer learning, electromyography (EMG), machine learning, meta learning, domain-adversarial neural networks (DANN), random forest, model ensemble, fine-tuning, gesture recognition, force regression.**

Manuscript received 4 March 2023; revised 28 June 2023; accepted 10 July 2023. Date of publication 14 July 2023; date of current version 26 July 2023. This work was supported in part by the Zhejiang Key Research and Development Program under Project 2021C03002 and in part by the Zhejiang Leading Innovative and Entrepreneur Team Introduction Program under Grant 2020R01005. *(Corresponding authors: Jie Yang; Mohamad Sawan.)*

Di Wu is with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310007, China, and also with the Center of Excellence in Biomedical Research on Advanced Integrated-on-Chips Neurotechnologies (CenBRAIN Neurotech), School of Engineering, Westlake University, Hangzhou 310024, China (e-mail: wudi@westlake.edu.cn).

Jie Yang and Mohamad Sawan are with the Center of Excellence in Biomedical Research on Advanced Integrated-on-Chips Neurotechnologies (CenBRAIN Neurotech), School of Engineering, Westlake University, Hangzhou 310024, China, and also with the Institute of Advanced Technology, Westlake Institute for Advanced Study, Hangzhou 310024, China (e-mail: yangjie@westlake.edu.cn; sawan@westlake.edu.cn).

Digital Object Identifier 10.1109/TNSRE.2023.3295453

I. INTRODUCTION

THE human motor control system is a complex neural system crucial for daily human activities. One way to study the human motor control system is to record the signal due to muscle fiber contractions associated with human motor activities by means of either inserting needle electrodes into the muscles or attaching electrodes onto the surface of the skin. The signal obtained is referred to as electromyography (EMG). Given the location of the electrodes, EMG is further divided into surface EMG (sEMG) and intramuscular EMG (iEMG). Advancement in the analysis of EMG and machine learning has recently achieved remarkable success enabling a wide variety of applications, including but not limited to rehabilitation with prostheses [1], hand gesture recognition [2] and human-machine interfaces (HMIs) [3].

The current success of applying deep learning onto EMG related tasks is largely confined to the following two assumptions, which are usually infeasible when it comes to real-world EMG related scenarios:

1) **Sufficient amount of annotated training data.** The growing capability and capacity of deep neural networks (DNN) architectures are associated with large amounts of labeled data [4], [5]. Such high quality abundant, labeled data are often limited, expensive, and inaccessible in the domain of EMG analysis. On the one hand, the EMG data acquisition process is a highly physical and time-consuming task that requires several days of collaboration from multiple parties [6]. On the other hand, EMG data annotation associated with biomedical applications such as the diagnosis of neuromuscular disorders requires expert knowledge [7].
2) **Training data and testing data are independent and identically distributed (i.i.d).** The performance of the model is largely affected by the distribution gap between the training and testing datasets. The testing data might also refer to the data generated during actual application usage after model deployment. Take hand gesture recognition, for example. The model is only capable of giving accurate predictions with the exact same positioning of the forearm of the test subject and the exact placement of the electrodes [8], [9].

As the distribution of data changes, models based on statistics need to be reconstructed with newly collected training data. In many real-world applications, it is expensive and impractical to recollect a large amount of training data and rebuild the models each time a distribution change is observed. Transfer learning (TL), which emphasizes the transfer of knowledge across domains, emerges as a promising machine learning solution for solving the above problems. The notion of transfer learning is not new, Woodworth and Thorndike [10] suggested that the improvement over one task is beneficial to the efficiency of learning other tasks given the similarity exists between these two tasks. In practice, a person knowing how to ride a bicycle can learn to ride a motorcycle faster than others since both tasks require balance keeping. However, transfer learning for EMG related tasks has only been gaining attention with the recent development of both DNN and HMIs. Existing surveys provide an overview of DNN for EMG-based human machine interfaces [11], and transfer learning in general for various machine learning tasks [12]. This survey focuses on the intersection of machine learning for EMG and transfer learning via EMG biological foundations, providing insights into a novel and growing area of research. Besides the analysis of recent deep learning works, we make an attempt to explain the relationships and differences between non-deep learning and the deep models, for these works usually share similar intuitions and observations. Some of the previous non-deep learning works contain more biological significance that can inspire further DNN-based research in this field. To consolidate these recent advances, we propose a new taxonomy for transfer learning on EMG tasks, and also provide a collection of predominant benchmark datasets following our taxonomy.

The main contributions of this paper are:

- Over fifty representative up-to-date transfer learning approaches on EMG analysis are summarized with organized categorization, presenting a comprehensive overview to the readers.
- Delve deep into the generating mechanisms of EMG and bridge transfer learning practices with the underlying biological foundation.
- Point out the technical limitations of current research and discuss promising directions on transfer learning on EMG analysis to propose further studies.

The remainder of this paper is organized as follows. We introduce in section II the basics of transfer learning, generation, and acquisition of EMG and EMG transfer learning scenarios. In Section III, we first provide the categorization of EMG transfer learning based on existing works and then introduce in detail. We also give a summary of the commonly used datasets in Section IV. Lastly, we discuss existing methods and the future research direction of EMG transfer learning.

II. Preliminaries

We introduce in this section the definitions of transfer learning, and related concepts and then summarize possible transfer scenarios. Moreover, we introduce the basics of EMG and highlight the correspondence between transfer scenarios and EMG generation and recording mechanisms.

A. Transfer Learning

We first give the definitions of a "domain" and a "task", respectively. Define $\mathcal{D}$ to be a domain that consists of a feature space $\mathcal{X}$ and a marginal probability distribution $P(X)$, where X is a set of data samples $X = [x_i]_{i=1}^n$. In particular, if two domains have different feature spaces or marginal probability distributions, they differ from each other. Given a domain $\mathcal{D} = \{\mathcal{X},\ P(X)\}$, a task is then represented by $\mathcal{T} = \{\mathcal{Y},\ f(\cdot)\}$ where $f(\cdot)$ denotes the objective prediction function and $\mathcal{Y}$ is the label space associated with $\mathcal{X}$. From the probability point of view, $f(x)$ can also be regarded as conditional probability distribution $P(y|x)$. Two tasks are considered different if they have different label spaces of different conditional probability distributions. Then, transfer learning can be formally defined as follows:

Definition 1 (Transfer Learning): Given a source learning task $\mathcal{T}_S$ based on a source domain $\mathcal{D}_S$, transfer learning aims to help improve the learning of the target objective prediction function $f_T(x)$ of the target task $\mathcal{T}_S$ based on the target domain $\mathcal{D}_T$, given that $\mathcal{D}_T \neq \mathcal{D}_S$ or $\mathcal{T}_S \neq \mathcal{T}_T$.

The above definition could be extended to multiple domains and tasks for both source and target. It is worth noticing that the majority of works surveyed in this paper only consider the case where there is one source domain $\mathcal{D}_S$, and one target domain $\mathcal{D}_T$, as by far, this is the most intensively studied transfer setup of the research works in the literature. Based on different setups of the source and target domains and tasks, transfer learning could be roughly categorized into *inductive transfer learning*, *transductive transfer learning* and *unsupervised transfer learning* [13].

Definition 2 (Inductive Transfer Learning): Given a transfer learning task $(\mathcal{D}_S, \mathcal{T}_S, \mathcal{D}_T, \mathcal{T}_T)$. It is a *inductive transfer learning* task where the knowledge of $(\mathcal{D}_S$ and $\mathcal{T}_S$ is used to improve the learning of the target objective prediction function $f_T(x)$ when $\mathcal{T}_S \neq \mathcal{T}_T$. The target objective predictive function can be induced by using a few labeled data in the target domain as the training data.

Definition 3 (Transductive Transfer Learning): Given a transfer learning task $(\mathcal{D}_S, \mathcal{T}_S, \mathcal{D}_T, \mathcal{T}_T)$. It is a *transductive transfer learning* task where the knowledge of $\mathcal{D}_S$ and $\mathcal{T}_S$ is used to improve the learning of the target objective prediction function $f_T(x)$ when $\mathcal{D}_S \neq \mathcal{D}_T$ and $\mathcal{T}_S = \mathcal{T}_T$.

For *transductive transfer learning*, the source and target tasks are the same, while the source and target domain vary. Similar to the setting of transductive learning of traditional machine learning [14], *transductive transfer learning* aims to make the best use of the given unlabeled data in the target domain to adapt the objective predictive function learned in the source domain, minimizing the expected error on the target domain. It is worth to notice that *domain adaptation* is a special case where $\mathcal{X}_S = \mathcal{X}_T$, $\mathcal{Y}_S = \mathcal{Y}_T$, $P_S(y|X) \neq P_T(y|X)$ and/or $P_S(X) \neq P_T(X)$.

Definition 4 (Unsupervised Transfer Learning): Given a transfer learning task $(\mathcal{D}_S, \mathcal{T}_S, \mathcal{D}_T, \mathcal{T}_T)$. It is an *unsupervised transfer learning* task where the knowledge of $\mathcal{D}_S$ and $\mathcal{T}_S$ is used to improve the learning of the target objective prediction function $f_T(x)$ with $\mathcal{Y}_S$ and $\mathcal{Y}_T$ not observed.

Based on the above definition, no data annotation is accessible in both the source and target domain during training. There has been little research conducted on this setting to date, given its fully unsupervised nature in both domains.

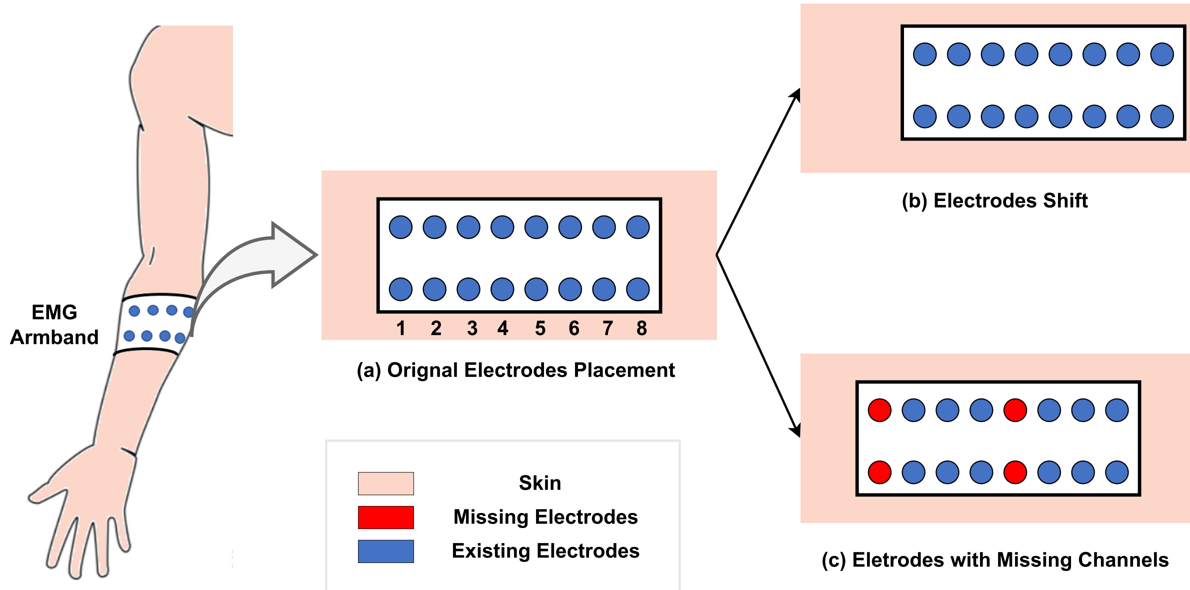

Fig. 1. Illustration of electrode variation. The left-hand side shows an EMG acquisition armband put on the forearm of a subject. (a), (b) and (c) are the net of the armband and the corresponding skin underneath. Colored circles represent electrodes, with two vertically placed electrodes being one bipolar channel. (a) demonstrates the original placement of an eight-channel bi-polar EMG collecting armband on the surface of the skin. (b) shows a shifted placement of the electrodes on the skin compared to (a). (c) is the case where electrode placement is the same as (a), but some channels are missing due to any reason.

B. Transfer Scenarios of EMG

Based on various factors in actual usage scenarios that cause a difference between the source and target domains, the factors that lead to domain differences can be categorized into intrinsic and extrinsic factors. We define the intrinsic factors to be the factors that affect the generation of the EMG signal. Such factors include individual variations, muscle fatigue, contraction force variations, contraction pattern variations, etc. However, in most data acquisition processes surveyed in this paper, subjects are instructed to avoid factors such as muscle fatigue by resting in between multiple data collection sets. Consequently, we only focus on factors that cannot be avoided via deliberately designed data acquisition protocols, such as individual variations. Extrinsic factors refer to the factors that affect the collection of the EMG signal, such as variation in electrode placement, variation in collection devices, variation in downstream task requirement, etc. We summarize the common transfer settings of the works surveyed in this paper as follows:

1) **Inter-subject.** EMG signals have substantial variation across individuals. The variation comes from a different distribution of subcutaneous fat, muscle fiber diameter, and way of performing force. Inter-subject transfer refers to the scenario where data collected from one subject or other subjects is utilized to calibrate the target objective function on a new subject. The task and acquisition devices are assumed to be the same across individuals.
2) **Electrodes Variation.** Electrode variation could be categorized into electrode placement shift and channel variation. Channel variation refers to the situation where some channels are missing during actual use as compared to the number of channels while recording EMG for model training. The placement of electrodes plays a crucial role in EMG applications. However, electrode shift is inevitable from wearing and taking off EMG acquisition devices, whether in the form of armband [11] or sockets [15]. Figure 1 provides a visualization of electrode variation in the case of an eight-channel EMG armband acquisition device. Consider the task of hand gesture and source domain associated with data collected with electrode placement shown in Figure 1(a). A transfer learning setting is formed with the target domain consisting of the same task and data collected with electrode placement shown in Figure 1(b) or with missing channels as in Figure 1(c).
3) **Inter-session.** In real-world applications, models are built with data collected from previous sessions and applied to new sessions. Data distribution varies across sessions due to reasons such as a different way of performing gestures, variation in electrode placement, or simply muscle fatigue. Inter-session transfer refers to the scenario where data collected from previous sessions is utilized to calibrate the target objective function in a new session. The task, acquisition device, and subject are assumed to be the same across sessions.
4) **Modality Variation.** Modality transfer refers to the scenario where data collected on one or a few modalities is utilized to calibrate the target objective function on another or other modalities. The task and subject are assumed to be the same, while devices vary due to modality variation. For the same or relevant tasks, it is possible to utilize the knowledge learned from one modality and facilitate the performance of the objective prediction function on another modality. For example, the transfer learning due to modality variation could be between neurophysiological signals (EEG and EMG) [16].

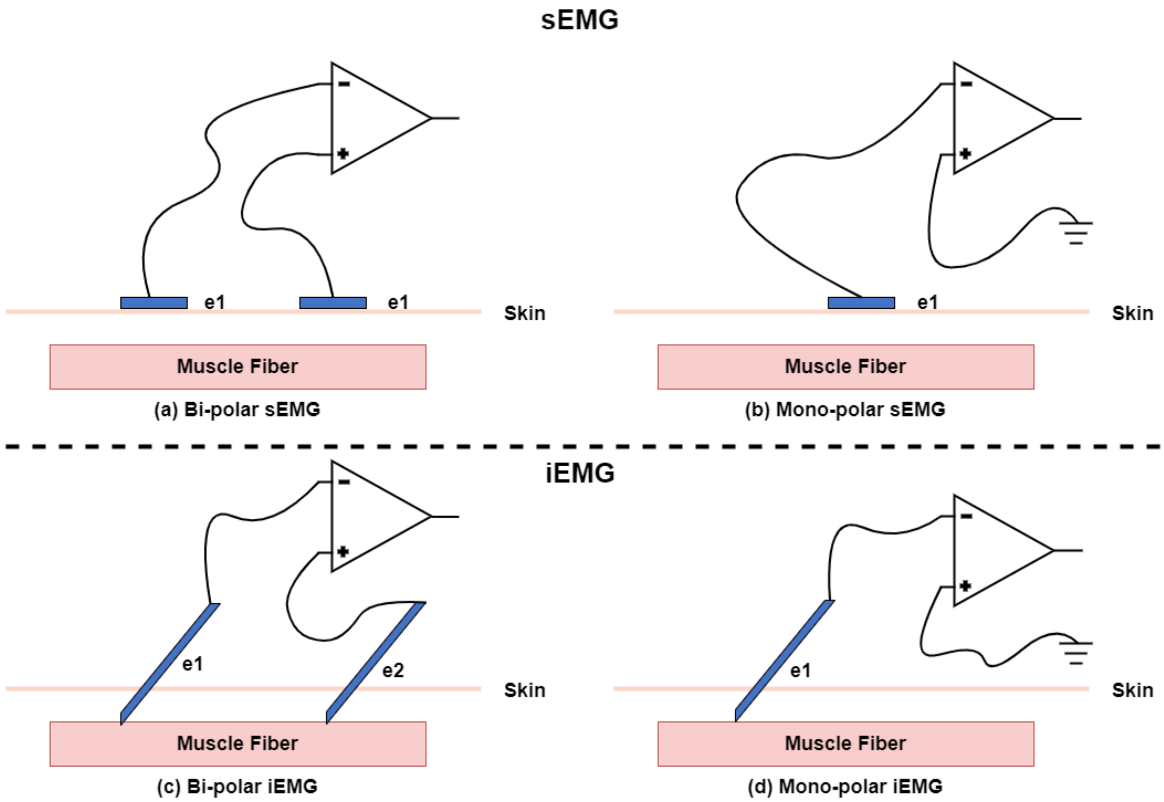

Fig. 2. Demonstration of EMG acquisition. The sEMG acquisition configuration is shown above the dotted line, with the iEMG acquisition configuration shown below the dotted line. The triangle represents an amplifier. For the bi-polar setup as in (a) and (c), two electrodes are placed on the skin surface or inserted into muscle fibers penetrating the skin surface. (b) and (d) show the case of a mono-polar setup with one electrode attached to the skin or muscle fiber and the other electrode connected to the ground or a reference point with no EMG (bones).

5) **Extensive Learning.** Extensive learning refers to the transfer scenario where new input data (target domain) extends either the data or/and the task of the source domain. For instance, the task of the source domain is a C class classification problem, while data collected in the target domain is of $C+K$ classes where K additional classes are incrementally added. The acquisition device and subject are assumed to be the same for both domains.

C. EMG Basics

In the previous section, we categorize the factors that cause domain differences into intrinsic and extrinsic ones. The intrinsic factors are closely related to the generation mechanisms of EMG signals, while extrinsic factors are usually associated with the signal acquisition process. Knowing how EMG is generated and recorded is crucial to understand the biological foundation behind various transfer learning approaches.

a) EMG Generation Mechanism: A *motor unit* (MU) is defined as one motor neuron and the muscle fibers that it innervates. During the contraction of a normal muscle, the muscle fibers of a motor unit are activated by its associated motor neuron. The membrane depolarization of the muscle fiber is accompanied by ions movement and thus generates an electromagnetic field in the vicinity of the muscle fiber. The detected potential or voltage within the electromagnetic field is referred to as the fiber action potential. The amplitude of the fiber action potential is related to the diameter of the corresponding muscle fiber and the distance to the recording electrode. A Motor Unit Action Potential (MUAP) is defined as the waveform consisting of the superimposed (both temporally and spatially) action potentials from each individual muscle fiber of the motor unit. The amplitude and shape of the MUAP are unique indicators of the properties of the MU (functionality, fiber arrangement, fiber diameter, etc.). Individuals can exhibit variations in both muscle fiber arrangement and muscle fiber diameter. These variations contribute to the diverse characteristics of muscles and can influence various factors, including strength, endurance, and muscle recruitment patterns. Consequently, these factors can lead to distinct EMG patterns across individuals, which explains why inter-subject scenario exists. It is important to note that muscle fatigue can lead to distinct EMG patterns, even within the same individual. In order to maintain stable motor movement, MUs are repeatedly activated to sustain muscle contraction. The repeated activation of MU generates a sequence of MUAPs forming a Motor Unit Action Potential Train (MUAPT). The recorded MUAPTs collected from multiple MUs are referred to as the commonly known EMG signal.

b) EMG Signal Acquisition: Based on the number of electrodes used during the recording of MUAPT, the recording techniques could be divided into mono-polar and bi-polar configurations. As shown in Figure 2, based on whether the electrodes are inserted directly into the muscles or placed on the surface of the skin, the collected signal is referred

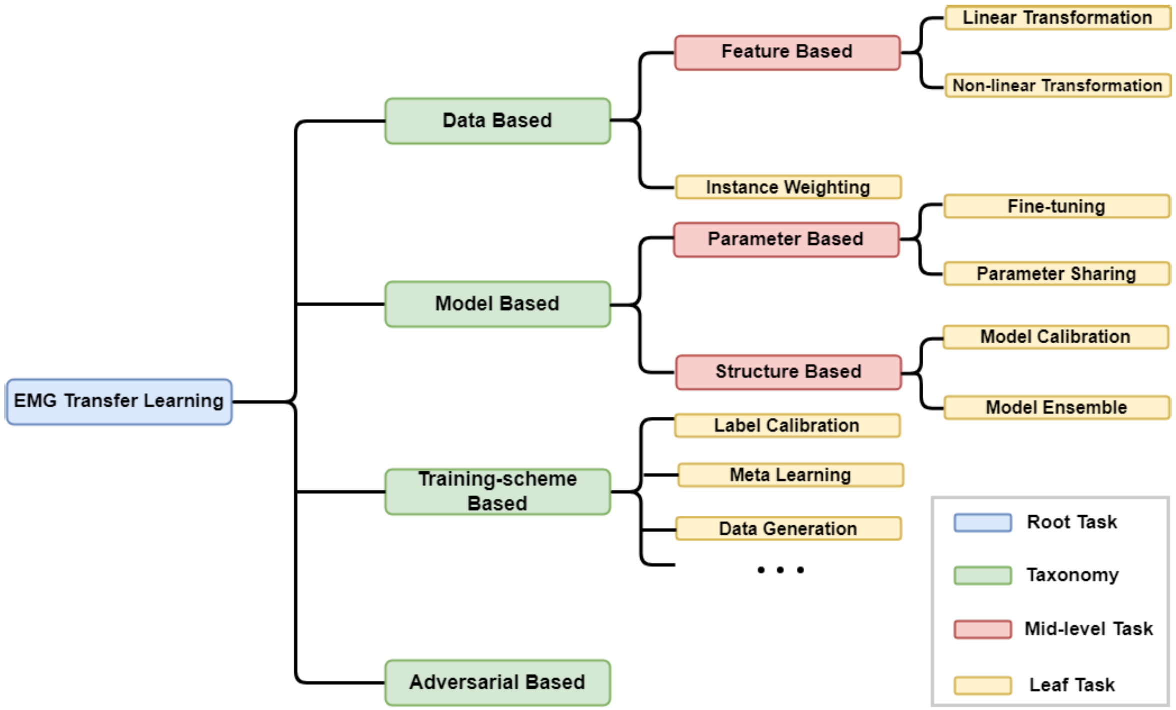

Fig. 3. Overview of categorization of transfer learning on EMG analysis.

to as intramuscular EMG (iEMG) or surface EMG (sEMG), respectively. A thin and sharp needle shaped electrode is quickly and smoothly inserted into the targeted muscle during iEMG acquisition [17]. iEMG is considered to have good spatial resolution due to the small diameter (around 0.5 mm) of the needle electrode. Individual MUAPTs could be identified by visualization. However, the effectiveness of the process of iEMG acquisition is highly dependent on the skill of the electrodiagnostic physician. Moreover, the punctuation procedure bears risks such as skin infection, severe bleeding, and muscle irritation. sEMG, on the other hand, is a non-invasive analysis tool for the human motor system that places electrodes on the surface of the skin [18]. sEMG is widely adopted for Human-Computer Interface (HCI) due to the major advantage of its ease of use and non-invasive nature. If muscle fibers belonging to multiple MUs are within the vicinity of the electrode, all MUAPTs from different MUs will be detected by the electrode. Given the different diameters of the electrode, sEMG is composed of MUAPTs from MUs from the same layer or deep layers, leading to poor spatial resolution as compared to iEMG. Consequently, the picked-up sEMG by the electrodes can be considered as a linear combination of MUAPTs. Each electrode placement pattern corresponds to a unique combination pattern. With electrodes placement varying, the linear combination pattern changes.

III. Transfer Learning in EMG Analysis

In the previous section, we introduced basic concepts on transfer learning on general and EMG generating mechanisms along with EMG acquisition techniques. These preliminaries shed insights on the underlying principles of recent progress in the area of transfer learning on EMG. In this section, we construct a categorization that best summarizes existing research endeavors of transfer learning in EMG analysis. As illustrated in Figure 3, we categorize existing works in EMG related transfer learning into four main lines, *i.e.*, data-based approaches, model-based approaches, training scheme based approaches, and adversarial-based approaches. Recall that the goal of transfer learning is to maximize the performance of the target objective prediction function, which can be achieved by either manipulating the data or modifying the model to reduce domain differences. In addition to the model-based and data-based interpretations, certain transfer strategies are based on specifically designed training schemes, such as mimicking the transfer learning process during training or calibrating labels during the collection of target domain data. We refer to these strategies as training scheme-based approaches. Furthermore, apart from directly minimizing the domain difference, adversarial-based approaches aim to force the neural network to learn hidden EMG representations that contain no domain discriminative information in an adversarial manner. To accomplish this goal, a negative gradient is usually adopted on top of special designs of neural network architectures. While these approaches may appear to fall within the category of model-based approaches, we list them separately due to their unique adversarial nature, which distinguishes them from conventional model-based approaches.

A. Data-Based Perspective

Data-based transfer learning approaches aim to reduce the data distribution difference between the source domain and target domain via data transformation and adjustment. From a data perspective, two approaches are generally employed in order to accomplish the knowledge transfer objective, namely instance weighting and feature based transformation.

1) Feature Based Strategy: Feature-based approaches map each original feature into a new feature representation either by linearly transforming the original feature or non-linearly transforming the original feature to enable knowledge transfer. Recall that in Section II-C, we mentioned that EMG signals are composed of superimposed MUAPTs generated from different MUs in both temporal and spatial domains. Assuming that only attenuation occurs with distance and there is no filtering effect, the relationship between the collected EMG signal and the generated MUAPs of the muscles can be modeled linearly. Therefore, the process of transfer learning aims to establish a mapping from one linear combination pattern to another. In this sense, linear transform is suited for the electrodes variation scenario. If no muscle fatigue and gesture performing pattern difference considered, linear transformation based approaches can be utilized to solve the inter-session scenario. Domain differences caused by factors such as individual variation, muscle contraction patterns can not be modeled linearly, thus non-linear transformation is usually adopted for scenarios such as inter-subject.

a) Linear Transformation: Lin et al. [19] proposed a normalization based approach called Referencing Normalisation to reduce the distribution difference among domains for inter-subject sEMG-based hand gesture classification. In specific, data from the source domain are mapped to the range of the target domain data:

$$\tilde{X_S} = \frac{(X_S - min(X_S))}{max(X_S) - min(X_S)} \cdot (max(X_T) - min(X_T)) + min(X_T), \quad (1)$$

where $\tilde{X_S}$ is the transformed source domain data.

In addition to directly applying a linear transformation to normalize the data to the target domain range, authors [8], [20], [21], [22] attempted to reduce the distribution gap based on statistical features such as covariance and mean. Conventional classifiers such as Linear Discriminant Analysis (LDA) [23], Quadratic Discriminant Analysis (QDA) [24], and Polynomial Classifier (PC) [25] are commonly adopted for sEMG classification tasks. The covariance matrix, mean vector, and the prior are the discriminant variables of LDA and QDA classifiers. Define $\Sigma_S, \Sigma_T, \mu_S, \mu_T$ to be the covariance matrices and mean vectors of data from the source domain and target domain, respectively. The transfer learning process of LDA and QDA based linear classifiers could be defined with a convex interpolation:

$$\tilde{\Sigma} = (1-\alpha) * \Sigma_S + \alpha * \Sigma_T \quad (2a)$$

$$\tilde{\mu} = (1-\beta) * \mu + \beta * \mu_T, \quad (2b)$$

where $\alpha, \beta \in [0, 1]$ are the trade-off parameters to balance the knowledge from the source and target domain, $\tilde{\Sigma}$ and $\tilde{\mu}$ represent the adapted covariance and mean vector. The optimal value for α and β are set empirically or via grid search with a fixed step size. Liu et al. [21] also proposed to use transfer learning on PC for the inter-session transfer scenario on both intact-limbed and amputee subjects. Let $\mathbf{M}$ be the polynomial expansion matrix of the training data, an optimal weight matrix $\mathbf{W}^*$ could be formulated as:

$$\mathbf{W}^* = \underset{\mathbf{W}}{\text{argmin}} \|\mathbf{MW} - \mathbf{Y}\|^2. \quad (3)$$

Similarly, the transfer learning process based on PC is defined as:

$$\tilde{\mathbf{W}} = \sum_{i=1}^{i=K} \beta^i \mathbf{W}^i + \bar{\mathbf{W}}, \quad (4)$$

where $\mathbf{W}^i$ and β^i are the optimal weight matrix for the i^{th} session and the corresponding weight ratio, $\bar{\mathbf{W}}$ represents the optimal weight matrix on the new session and $\tilde{\mathbf{W}}$ represents the adapted weight matrix. It is worth noticing that distance measurements such as Kullback–Leibler divergence [26] could be used to select the source domain that's the most similar to the target domain to avoid negative transfer when there are multiple source domains available [27].

Muscle Synergy Modeling (MSM) [9], [28], [29], [30] has shown great success in terms of modeling the linear relationship between MUAPTs of muscles and the collected EMG signal. Let $x_m(t)$ be the generated MUAPTs from the m^{th} muscle, define $act_i(t) \in \mathbb{R}$ to be the activation signals, $x_m(t)$ could then be expressed as:

$$x_m(t) = \sum_{i=1}^{i=N} g_{mi} \cdot act_i(t), \quad (5)$$

where g_{mi} is the gain factor of muscle m transferred to the i^{th} activation signal. Assuming that only attenuation exists with distance but no filtering effect, the observed EMG signal at the k^{th} electrode (k^{th}channel) is written as:

$$\begin{aligned} y_k(t) &= \sum_{m=m}^{m=M} \sum_{i=1}^{i=N} l_{km} \cdot g_{mi} \cdot act_i(t) \\ &= \sum_{i=1}^{i=N} a_{ki} \cdot act_i(t), \quad (6) \end{aligned}$$

where l_{km} is the factor that reflects the attenuation level from the m^{th} muscle on the k^{th} electrode and a_{ki} is the combined weight factor that models both l_{km} and g_{mi}. The above mixture could be written in matrix form:

$$\mathbf{Y} = \mathbf{A} \cdot \mathbf{F}, \quad (7)$$

where $\mathbf{A} \in \mathbb{R}^{K \times N}$ is the weighting matrix and $\mathbf{F}$ is the synergy matrix. In EMG analysis, $\mathbf{Y}$ is often observed, thus the solving for $\mathbf{W}$ and $\mathbf{F}$ becomes a linear blind source separation (BSS) problem [31]. Non-negative matrix factorization (NMF) [32] finds an approximate solution to the equation (7) with the constraint that all elements are non-negative.

Jiang et al. [33] proposed correlation-based data weighting (COR-W) for inter-subject transfer scenario of elbow torque modeling. In specific, they assume that the target domain data is a linear transformation of the source domain data, $X_T \approx \tilde{X_S} = \mathbf{A}X_S$, where $\tilde{X_S}$ is the transformed source domain data. The underlying assumption is that the synergy matrix remains the same for both domains while the weighting matrix varies. A derived assumption of Jiang et al. is that the

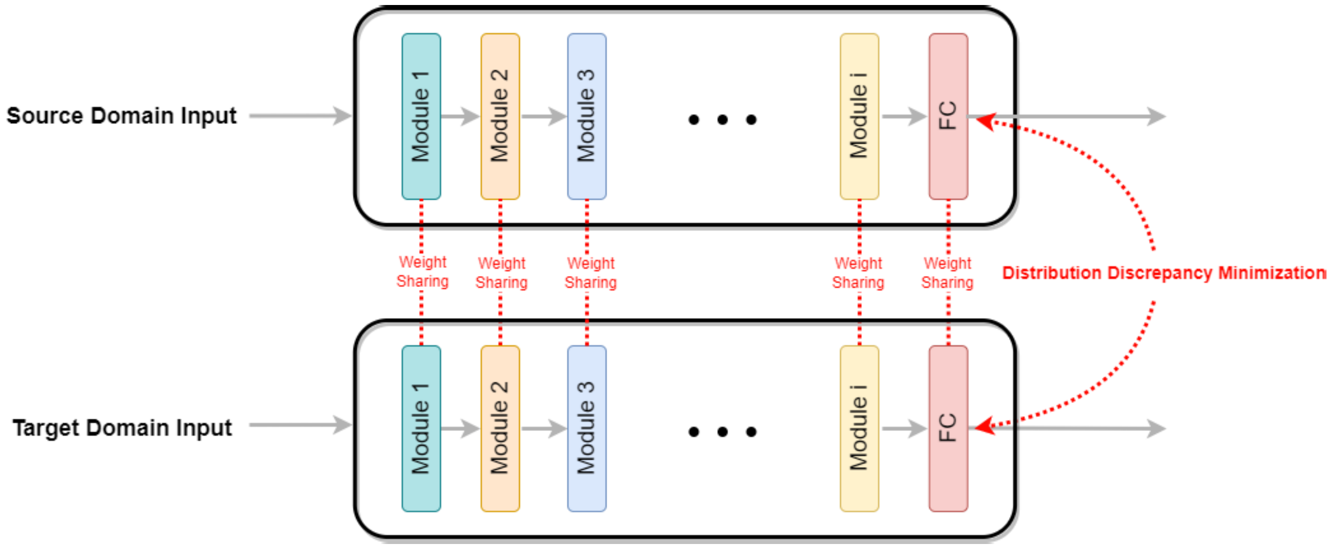

Fig. 4. Demonstration of applying the Siamese architecture for distribution discrepancy minimization. It is worth noticing that the design of neural network architectures varies across works. FC stands for the fully connected layer. The term 'module' refers to a combination of layers that might contain convolution, normalization, or residual connection. The distribution discrepancy measurement is applied to the output of the FC layer just for demonstration. The distribution discrepancy measurement could essentially be applied to deep features output by any module.

covariance matrix of the transformed source domain should also be similar to the covariance matrix of the target domain data. The optimal matrix A^* is estimated by minimizing the discrepancy between $\tilde{\Sigma}_S$ and Σ_T. The transformed source data is then used to re-train the model. Although Jiang et al. proposed for inter-subject transfer scenario, while we argue that the linear assumption might not hold due to variation across subjects. Electrode shift, on the other hand, is reasonably more consistent with the linear assumption in practice. Günay et al. [34] adopted MSM with NMF for knowledge transfer across different tasks. The weighting matrix W calculated on the source domain is kept constant while the synergy matrix is re-estimated on the target domain data using the non-negative least squares (NNLS) algorithm. In contrast to the works that map the source domain data to a new space, another line of work [35], [36], [37] transforms the target domain data so that the source domain objective prediction function is applicable again. Prahm et al. [35] viewed the target domain data as a disturbed version of the source domain data. The disturbance can be expressed as a linear transformation matrix A. The main aim is then to learn and apply an inverse disturbance matrix A^{-1} to the target data such that the disturbance is removed. Prahm et al. [35] adopted Generalized Matrix Learning Vector Quantization (GMLVQ) [38] as the classifier and estimated the optimal A^{-1} using gradient descent on the GMLVQ cost function. The linear transformation that maximizes the likelihood of disturbed data based on the undisturbed data could also be estimated by the Expectation and Maximization (EM) algorithm [37], [39]. Following their previous work [35], [37], Prahm et al. [36] proposed that the linear transformation matrix could be further exploited based on the prior knowledge that the underlying EMG device is an armband with eight uniformly distributed channels. For the electrode shift scenario, Prahm et al. assumed that the disturbed feature from channel j could be linearly interpolated from neighboring channels from both directions with a mixing ratio r. Then the approximation of the linear transformation matrix is reduced to finding an optimal mixing ratio r.

b) Non-linear Transformation: The principle objective of feature transformation is to reduce the data distribution between the source and target domain. Thus, the metrics for measuring distribution difference is essential. Maximum Mean Discrepancy (MMD) [40] is widely adopted in the field of transfer learning:

$$\mathcal{MMD}(X_T, X_S) = \left\| \frac{1}{N^S} \sum_{i=1}^{i=N^S} \Phi(X_S^i) - \frac{1}{N^T} \sum_{j=1}^{i=N^T} \Phi(X_T^j) \right\|^2, \tag{8}$$

where Φ indicates a non-linear mapping to the Reproducing Kernel Hilbert Space (RKHS) [41], N^S and N^T indicate the number of instances in the source and target domain, respectively. Essentially, MMD quantifies the distribution difference via calculating the distance between the mean vectors of the features in an RKHS. In addition to MMD, Kullback–Leibler divergence, Jenson–Shannon (JS) divergence [42] and Wasserstein distance [43] are also common distance measurement criteria. The Siamese architecture [44], [45] is one commonly adopted architecture for DNN related transfer learning, as illustrated in Figure 4. Zou and Cheng [46] proposed a Convolutional Neural Network (CNN) based model named Multiscale Kernel Convolutional Neural Network (MKCNN) for hand gesture recognition. The authors proposed a transfer learning MKCNN (TL-MKCNN), which contains a Distribution Alignment Module (DAM) for inter-subject and inter-session transfer learning scenarios. TL-MKCNN adopts the Siamese architecture, with one network taking inputs from the source domain and the other one taking inputs from the target domain. The Siamese networks share weights with each

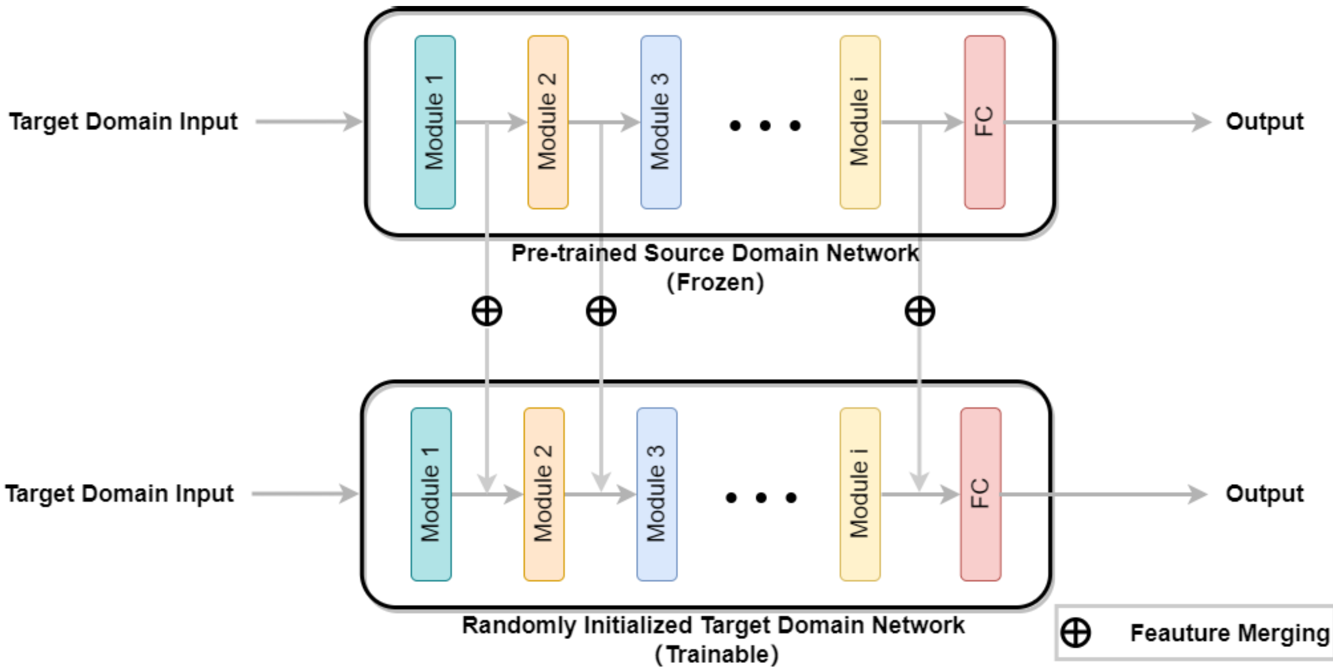

Fig. 5. Illustration of the architecture of the progressive neural network. Frozen indicates that the parameters of the network are fixed, while trainable suggests that the network parameters will be updated during training. The same input is fed to both networks, the intermediate features from each module of the pre-trained network is merged with corresponding intermediate features of the target domain network.

other. DAM applies the JS divergence onto the output of the second layer of the fully connected layers to minimize the distribution difference between the deep features of the data from the source and target domain. Besides the Cross Entropy (CE) loss function for classification, Zou et al. also applies a mean square error (MSE) to minimize the distance of instances to the corresponding class center. The overall loss function to train TL-MKCNN is the sum of JS divergence, CE, and MSE. Bao et al. [47] applied fast Fourier transform (FFT) to data segment and used the spectrum as input to their designed CNN based network. Similar to [46], the MMD loss is applied to the output of the second fully connected layer. A Regression Contrastive Loss is proposed to minimize the distance in the feature space between the source domain instance and the target domain instance of the same category. Normalization tricks are adopted to modify the loss for regression tasks.

Côté-Allard et al. [48], [49] proposed to use the Progressive Neural Network (PNN) [50] to alleviate *catastrophic forgetting* caused by directly fine-tuning the network parameters with data from the target domain. As shown in Figure 5, a source domain network is first trained with data from the source domain. The model parameters of the source domain network are then fixed, while the parameters for the target domain network is randomly initialized. Note that the network structures of both networks are exactly the same except for the model parameters. During the transfer learning process, target domain instances are fed to both networks. The intermediate features of each module of the source domain network is then merged with the corresponding features of the target domain network and fed forward to the next module of the target domain network. The underlying hypothesis is that although distribution variation exists between the source and target domain, generic and robust features could be attracted for more effective representation learning.

Du et al. [51] proposed to adopt Adaptive Batch Normalization (AdaBN) [52] for inter-session transfer learning. AdaBN is a lightweight transfer learning approach for DNNs based on Batch Normalization (BN) [53]. BN was initially proposed to accelerate the convergence of the DNN for faster CNN training. Formally, define $\mathbf{Z} = [\mathbf{z}_i]_{i=1}^{B}$ to be a batch of intermediate features of instances with batch size B, the BN layer transforms $\mathbf{Z}$ as follows:

$$\tilde{\mathbf{z}} = \gamma \cdot \frac{\mathbf{z}_j - \mathbb{E}[\mathbf{Z}_{\cdot j}]}{\sqrt{Var[\mathbf{Z}_{\cdot j}]}} + \beta, \tag{9}$$

where γ and β are learnable parameters, Var stands for variance. The underlying hypothesis is that labeled related knowledge is stored in the network parameters of each layer, and the domain related knowledge is portrayed by the statistics of the BN layers. The transformation ensures that the distribution of each layer remains the same over mini-batches so that each layer of the network receives input of similar distribution regardless of the source or target domain. Different from fine-tuning, AdaBN doesn't require target domain label for knowledge transfer, and only a small fraction of the network parameters need to be updated. In particular, the network is first pre-trained on source domain data. During the training process, the statistics of BN layers are calculated by applying a moving average for all data batches. All network parameters are fixed except for the parameters of BN layers during transfer learning. The update of BN statistics to target domain data could easily be done by a forward pass.

2) Instance Weighting: Instance weighting assumes that similarities exist between source domain and target domain data. This similarity can arise from different individuals as well as from different electrode placements. Instance weighting is suited for any transfer scenario as long as the domain difference is assumed to be small. Consider a special case of domain adaptation where $P_S(y|X) = P_T(y|X)$ and $P_S(X) \neq P_T(X)$ which is referred to as covariate shift [54]. Consider the transfer scenarios that we introduced in Section II-B, collecting abundant data in the target domain is often prohibitive, and thus target domain instances are limited. A natural solution is to assign weights to partial instances from the source domain so that these source domain instances can be used along with limited target domain data. Huang et al. proposed Kernel Mean Matching (KMM) [55] to estimate the instance weights by matching the means of the target and source domain in a Reproducing Kernel Hilbert Space (RKHS). The weighted instances from the source domain are combined with labeled target domain instances to train the target objective prediction function. Li et al. [56] proposed to use TrAdaBoost [57] along with Support Vector Machine (SVM) to improve the motion recognition performance under inter-session scenario. In specific, they first apply TrAdaBoost to weight EMG data of day one and train a target classifier with weighted EMG from day one and EMG collected from another day. TrAdaBoost iteratively adjusts the weights of instances to decrease the negative effect of the instances on the target learner. TrAdaBoost is largely inspired by a boosting algorithm called AdaBoost [58]. AdaBoost iteratively trains weak classifiers with updated weights. The weighting mechanism of AdaBoost is the misclassified instances are given more attention during the training of the next weak learner in the following iteration. The weighting mechanism of TrAdaBoost is to reduce the distribution difference between the source and target domains.

B. Model Based Perspective

From the model perspective, transfer learning approaches can also be interpreted in terms of model parameters and model structures.

1) Parameter Fine-Tuning: Deep neural networks (DNNs) are strong automatic feature extractors. Despite variations in biological characteristics among subjects, differences in electrode placement, and variations in signal modality, parameter fine-tuning assumes that common features could be extracted between the source domain and target domain. It further assumes that the knowledge learned from the source domain can be effectively transferred to the target domain. parameter fine-tuning is well-suited for all transfer scenarios. One intuitive way of transferring knowledge of DNN is to tune the network parameters of the source learner using data from the target domain. Fine-tuning [59] refers to the training process where the network is first trained on one dataset (large-scale) and use the network parameters as initialization to further train on another dataset (small scale). Fine-tuning is a common strategy in the Computer Vision (CV) community where the neural networks are first pre-trained on ImageNet (IN) either in a supervised manner or self-supervised manner and later fine-tuned for various downstream tasks such as classification [60] and object detection [61]. IN-21K is a large scale dataset with 15 million images over 2200 classes. The underlying assumption is that the dataset for downstream tasks (target domain) is of similar data distribution of IN (source domain). Extensive experiments have shown that using IN pre-trained weights and fine-tuning improves performance by a large margin. Inspired by the success of fine-tuning in CV, authors [62], [63], [64] first transform EMG signal to 2D data via Short-time Fourier Transform (STFT) and treat the spectrogram as image input to the neural network. It is worth noticing that these works [62], [63], [64] use IN pre-trained network to improve the performance of the network that takes STFT as input. This line of work do not fall into any of the transfer learning scenarios that we summarized in Section II-B. The neural network learns feature extraction ability (texture and shape) and the ability to localize to the foreground object. IN mainly contains images of natural scenes such as objects, animals, and humans. Since the gap between the source domain (natural scenes) and the target domain (spectrum image) is tremendous, it is questionable as to what knowledge is transferable. Phoo et al. [65] compared the transfer performance of using miniIN (a small subset of IN) as source domain and using IN as source domain to ChestX (X-ray images for chest) [66] as target domain. Experimental results show that pre-training on IN yields no better performance than on miniIN and both yields poor diagnosis accuracy. This suggests that more data does not help improve the generalization ability, given that no more informative knowledge can be extracted from the source domain to benefit the target domain learner. Pre-training the network on the source domain and then using the pre-trained weights to initialize the neural network for further training using the target domain data is another popular fine-tuning strategy for EMG transfer learning [8], [67], [68], [69], [70]. There would be little constraint nor assumption on the transfer scenarios since this transfer process is simple and can be viewed as sequentially train the network with two datasets. When there are EMG data recorded from multiple subjects or sessions, it is possible to combine the data and treat the combined data as the source domain [71], [72]. Or it is also a solution to train a unique model for each subject or session and to select a certain number of models that give the best performance on the target domain [73], [74], the selected models are then fine-tuned on the target dataset to provide final prediction based on majority voting [75]. However, fine-tuning suffers from the *catastrophic forgetting*, meaning that knowledge from the source domain will be forgotten by the neural network rapidly upon the introduction of target domain data [76]. Besides the parameters fine-tuning of DNNs, the parameters of Decision Trees [77] (DTs) could also be fine-tuned for EMG transfer learning [78]. The motivation is that the structure of decision trees for similar tasks should be similar and the domain difference is reflected from different decision threshold values associated with the features. Structure Transfer (STRUT) [79] first discards all the numeric threshold values of learned trees on the source domain data and selects a new threshold value $\tau(\nu)$ for a node ν given that the subset of target examples reach ν in a top-down manner. Any node ν that's empty in terms of target domain data is considered unreachable and will be pruned. Define τ to be the threshold value of feature ϕ at node ν

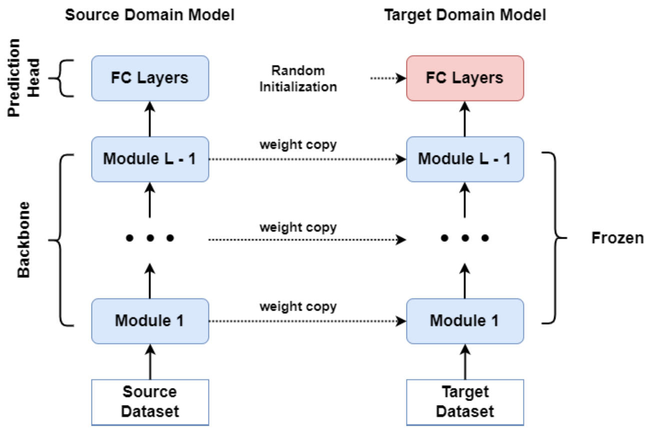

Fig. 6. Illustration of transferring knowledge by sharing the weights of the neural network. The weights of the backbone modules are first copied to the target domain network and frozen. The term 'module' refers to a combination of layers that might contain convolution, normalization, or residual connection. FC stands for the fully connected layer. The weights of the prediction head are randomly initialized and trained from scratch.

that splits any set of labeled data S_v into two subsets, denoted S_L and S_R. P_L and P_R denote the label distribution of S_L and S_R, respectively. STRUT aims to find a new threshold τ' with maximum Divergence Gain (DG) subject to the condition where the new thresholds are local maximums of Information Gain (IG) [77]:

$$\mathcal{DG} = 1 - \frac{\|S_L^T\|}{\|S_v^T\|}\mathcal{JSD}(Q_L^T, Q_L^S) - \frac{\|S_R^T\|}{\|S_v^T\|}\mathcal{JSD}(Q_R^T, Q_R^S), \tag{10}$$

where $\|\cdot\|$ stands for the cardinality, S and T on the superscript stand for the source and target, respectively.

2) Parameter Sharing: The neural network architectures are not specified in Section III-B.1 since parameter fine-tuning tunes all parameters of the network regardless of various network designs. It is stated that fine-tuning the whole network suffers from *catastrophic forgetting* and knowledge learned from the source domain will be quickly forgotten. In most of the works [8], [67], [68], [69], [70] that adopt fine-tuning, the target domain dataset is of the same size as the source domain dataset. Consider the case where the target domain dataset is small compared to the source domain, with forgotten knowledge from the source domain, the neural network is prone to suffer from over-fitting [80]. A possible solution is to freeze partial network parameters and to only update partial parameters during the fine-tuning process [81], [82], [83], [84], [85], [86], [87]. An illustration of knowledge transferring via parameter sharing is provided in Figure 6. A neural network design could be roughly divided into the backbone and the prediction head. The backbone serves as the feature extractor and is usually CNN based or Recurrent Neural Networks (RNN) based. The prediction head is usually composed of fully connected layers and predicts the desired labels based on the deep features extracted by the backbone. Assuming that the extracted deep features are generic for various transfer scenarios, the weight of the backbone could be frozen once pre-trained on the source domain dataset to prevent *catastrophic forgetting*. Only the fully connected layers of the prediction head need to be updated, which reduces transfer training time and guarantees fast convergence.

3) Model Structure Calibration: Besides knowledge transferring via trained parameters, next we explore the possibility of EMG transfer learning from the model structure perspective. Since it is often the case that there is a lack of labeled data in the target domain and as such it might not be sufficient to construct a reliable high performance model solely on the target domain data, optimizing the model structure of a pre-trained model to fit the target domain data is desired. As we mentioned in the previous section that DNNs are believed to be able to extract generic features, thus it is impractical and time consuming to alter or even search for neural network structures using Neural Architecture Search (NAS) [88] for various domains. However, Random Forest (RF) [89] on the other hand, is more suitable for structure calibration since knowledge transfer could be done by pruning or growing the source tree model. Marano et al. [78] proposed to use structure expansion/reduction (SER) [79] for EMG based hand prostheses control. As the name suggests, the SER algorithm contains two phases: *expansion* and *reduction*. Consider an initial random forest that is induced using the source domain data. In the *expansion* phase, SER first calculates all labeled data points in the target domain dataset that reaches node v and then extends node v into a full tree. In the *reduction* phase is performed to reduce the model structure in a bottom-up fashion. Define $\mathcal{E}_{sub}$ to be the empirical error of the subtree with root node v, $\mathcal{E}_{leaf}$ denotes the empirical error on node v if v were to be pruned to a leaf node. The subtree is to be pruned into a node leaf if $\mathcal{E}_{sub} > \mathcal{E}_{leaf}$. SER is performed on each decision tree separately and the resulting random forest is the adapted model for the target domain data.

4) Model Ensemble: Combining data from various sources into a single source domain may not yield satisfactory results since the distributions of these domains might vary greatly from each other. Another commonly adopted strategy for EMG transfer learning is model ensemble. The model ensemble aims to combine a set of weak learners to make the final prediction. Some previously reviewed EMG transfer learning approaches already adopted this strategy. For instance, Kim et al. [73] proposed to train a unique classifier for each subject and further fine-tune the top ten best performing classifiers on a new target subject. The final prediction is the most commonly classified by the ensemble of all ten fine-tuned classifiers. Decision Trees are another popular choice for weak learners. Zhang et al. [90] proposed feature incremental and decremental learning method (FIDE) based on Stratified Random Forest (SRF) for knowledge transfer with missing or added electrodes. In specific, define S_i and S_j to be the electrode sketch score [91] for electrode e_i and e_j, respectively. The distribution difference between electrodes e_i and e_j is defined as:

$$\mathcal{DD}(i, j) = \frac{\rho(S_i, S_j) + \psi(e_i.e_j) + 1}{4}, \tag{11}$$

where $\rho(\cdot)$ stands for the Pearson Correlation Coefficients (PCC) and ψ denotes the inverse of the Euclidean distance between e_i and e_j. K-means [92] is then utilized to cluster the electrodes into K clusters based on the DD. Denote M as the number of weak learners in the ensemble model, SRF is built on the source domain data where $\lceil M/K \rceil$ trees are induced using data collected with electrodes in the corresponding cluster. If electrode i is missing in the target domain data, the missing features could be recovered from the most similar electrode j. If there are incremental electrodes in the target domain dataset, FIDE first selects a set of weak learners to be updated based on a performance score:

$$\mathcal{S}(m) = acc(h_m) + \frac{\#feature_m}{\#feature}, \tag{12}$$

where h_m stands for the m^{th} decision tree, $\#feature_m$ denotes the number of features used by h_m, and $\#feature$ denotes the total number of features. Top $M * \delta$ weak learners are then selected for updated where $\delta \in [0, 1]$. The SER and STRUT algorithms [79] introduced in previous sections are again used for transfer learning on decision trees. Compared to the majority voting way of ensemble, FIDE updates the source domain model to extract new knowledge from target domain data while not abandoning the already learned knowledge.

C. Training-Scheme Based Perspective

In addition to the previously mentioned approaches that can be subsumed into pre-defined paradigms, we also review works that design special training schemes for EMG transfer learning. In theory, this line of work has the potential to be applicable in various transfer scenarios, given a well-designed scheme. The effectiveness of this approach lies in its adaptability across different contexts. By employing a carefully crafted scheme, it becomes feasible to leverage the benefits of this methodology in a wide range of transfer scenarios. Zhai et al. [93] proposed a self re-calibration approach for inter-session hand prosthesis control. In particular, a source domain classifier is first trained with EMG data of existing sessions. Given the target domain data, each EMG data segment x^i is assigned a prediction label y^i by applying a forward pass of the EMG segments. Based on the assumption that temporally adjacent EMG segments are likely to be generated from the same hand movement, the assigned labels are re-calibrated with majority voting:

$$\tilde{y}^i \longleftarrow Majority\ Voting(f_{\mathcal{S}}(x^{i-k}, x^{i-k+1}, \ldots, x^i, \ldots, x^{i+k})), \tag{13}$$

where $f_{\mathcal{S}}$ is the source domain classifier, and k indicates the number of neighboring segments used to re-calibrate the label from both directions in time before and after x^i. Then the target domain data with re-calibrated labels are used to update the source domain classifier. It is worth noticing that such a transfer scheme does not require the target domain labels and can be easily adopted for day-to-day re-calibration.

Meta-learning [94] is another training paradigm that can be used for EMG transfer learning. Meta-learning is commonly known as learning to learn [95]. In contrast to conventional machine learning algorithms that optimize the model over one learning episode, meta-learning improves the model over multiple learning episodes. The meta-learning goal of generalizing the model to a new task of an incoming learning episode with limited samples aligns well with the notion of transfer learning. Intuitively speaking, meta-learning divide the source domain data into multiple learning episodes, with each containing a few samples and mimicking the transfer processing during training so that the model trained has good transferability in terms of the true target domain. Rahimian et al. [96] proposed meta-learning based training scheme called Few-Shot Hand Gesture Recognition (FHGR) for the transfer case where only a minimal amount of target domain data are available for re-calibration. Define an N-way k-shot few shot learning problem, let $\mathcal{T}_j = \{D_j^{train}, D_j^{test}, \mathcal{L}\}$ denote a task associated with the source domain dataset where $D_j^{train} = \{(x_i, y_i)\}_{i=1}^{K\times N}$ and $\mathcal{L}$ is a loss function to measure the error between the prediction and the ground-truth label. Please be aware that the task $\mathcal{T}$ here is a naming convention in the meta-learning area and is of a different meaning than the task that we define for a domain. FHGR aims to predict the labels of D_j^{test} based on the samples seen from D_j^{train} consisting of K samples from each of the N classes over a set of tasks samples from $p(\mathcal{T})$. A Pseudocode in the MAML style [97] is provided in Algorithm 1.

Algorithm 1 MAML Style Meta-Learning for Transfer Learning

Input :
Task distribution: $p(\mathcal{T})$, Loss function: $\mathcal{L}$, learning rate for inner loop: α, learning rate for outer loop: β
Output : Prediction Model: f_Θ,
Initialization :Randomly initialize Θ
while *not done* **do**
  Sample a batch of tasks $\mathcal{T}_i$ from $p(\mathcal{T})$
  for *all task* $\mathcal{T}_i$ **do**
    Evaluate error $\mathcal{L}_{\mathcal{T}_i}(f_\Theta)$ with respect to the D_j^{train}
    Update Θ with gradient descent:
    $\tilde{\Theta} \longleftarrow \Theta - \alpha \cdot \frac{\partial \mathcal{L}_{\mathcal{T}_i}(f_\Theta)}{\partial \Theta}$
  end
  Evaluate error $\mathcal{L}_{\mathcal{T}_i}(f_{\tilde{\Theta}})$ with respect to the D_j^{test}
  Update Θ with gradient descent:
  $\hat{\Theta} \longleftarrow \Theta - \beta \cdot \frac{\partial \mathcal{L}_{\mathcal{T}_i}(f_{\tilde{\Theta}})}{\partial \tilde{\Theta}}$
end

EMG transfer learning could also benefit from data augmentation via generating synthetic data as data from other sessions or subjects (target domain data). Generative Adversarial Networks (GANs) are a famous type of network for data generation without explicitly modeling the data probability distribution. A typical GAN contains a generator G and the discriminator D, which are two neural networks. A random noise vector sampled from a Gaussian or uniform distribution is input to the generator network to produce a sample x_g that should be similar to a real data sample x_r drawn from a true data distribution P_r. Either x_r or x_g is input to the discriminator to get a classification result of whether the input in real or fake. Intuitively, the generator aims to generate fake samples that could confuse the discriminator as much as possible, while the task of the discriminator is to best distinguish fake samples from real ones. The training objective of GAN can be defined as:

$$\mathcal{L}_D = \max_D \mathbb{E}_{x_r}[log D(x_r)] + \mathbb{E}_{x_g}[log(1 - D(x_g))], \tag{14a}$$

$$\mathcal{L}_G = \max_D \mathbb{E}_{x_g}[log(1 - D(x_g))]. \tag{14b}$$

Zanini and Colombini [98] adopted DCGAN [99], which is a convolution-based extension of the very original GAN and style transfer for Parkinson's Disease EMG data augmentation.

Besides GANs, style transfer has also been utilized to augment EMG data. Given a piece of fine artwork, painting, for example, humans have the ability to appreciate the interaction of content and style. "The Starry Night" by Van Gogh is an appealing painting that attracts a lot of re-drawing attention which follows the same drawing style of Van Gogh but with different content. Gatys et al. [100] proposed an algorithm for artistic style transfer that combines content from one painting and the style of another painting. A similar idea could be extended to EMG signals for transfer learning. An EMG signal can also be regarded as the interaction of content and style. The style might refer to the biological characteristics of the subject, such as muscle condition, the filtering effect of a recording device, or simply a session. The content depicts the spikes carrying moving intention from the neural system to the corresponding muscles. Considering that the content of the different muscle movements are the same regardless of any other conditions, the style component then processes the control signals for moving to subject, device, or session specific data. Zanini et al. [98] adopted style transfer [100] to augment Parkinson's Disease EMG data of different patterns. Specifically, given a content EMG signal e_c and a style image e_s, the algorithm aims to find an EMG signal e that's of the same content as e_c and of the same style as e_s. Mathematically, the transferring process minimizes the following loss function:

$$\mathcal{L}_c(e, e_c) = \sum_l \left\| \mathcal{F}^l(e_c) - \mathcal{F}^l(e) \right\|^2, \tag{15a}$$

$$\mathcal{L}_s(e, e_s) = \sum_l \left\| \mathcal{G}(\mathcal{F}^l(e_c)) - \mathcal{G}(\mathcal{F}^l(e)) \right\|^2, \tag{15b}$$

where $\mathcal{F}(\cdot)$ is the output feature of the l^{th} layer of the neural network, $\mathcal{G}$ stands for the Gram matrix [101]. The content component and style component are controlled by two hyper-parameters.

$$\mathcal{L} = \alpha * \mathcal{L}_c + \beta * \mathcal{L}_s \tag{16}$$

Besides directly generating EMG data, Suri and Guptaet al. [102] proposed to synthesize extracted features of EMG signals with an LSTM network [103] to mimic EMG data from other subjects or different sessions. Different from GAN and style transfer based EMG augmentation that is directed by loss functions that either measure the authenticity or similarity, the method proposed by Suri et al. simply relies on the assumption that extracted features are robust and that EMG signal generated by altering features are correlated to the recorded real data.

D. Adversarial Based Perspective

Recall that in Section III-A.1, we introduce non-linear feature based approaches that reduce the data distribution by explicit deep feature transformation. In this section, we review a set of methods that force the neural network to learn hidden EMG representations that contain no discriminative information in terms of the origin of the data for domain generic feature extraction. Hence, this category of methods is well-suited for scenarios involving inter-session, inter-subject, and electrode variation. For instance, in the context of gesture recognition, it is important to note that muscle arrangement and diameter can differ among individuals. However, when performing a specific gesture, the same set of muscles must contract. Adversarial-based approaches aim to identify these common patterns while explicitly discarding the different patterns between the source and target domains. With this objective, Domain-Adversarial Neural Networks (DANN) [104] is a type of neural network that contains a backbone $\mathcal{F}(\cdot; \theta_\mathcal{F})$ parameterized by $\theta_\mathcal{F}$ for feature extraction and two prediction heads: one for predicting the task label and another for predicting the origin of the data (source or target domain). We refer to the prediction head for the source domain task as the task prediction head $\mathcal{P}_t(\cdot; \theta_t)$ and refer to the prediction head for domain classification as domain prediction head $\mathcal{P}_d(\cdot; \theta_d)$. The parameters of the network are optimized in a way that the learned deep feature minimizes the loss for the task prediction head while maximizing the loss for the domain prediction head. The domain prediction head works adversarially to the task prediction head hence the name DANN. Formally, the overall loss function for optimizing $\theta_\mathcal{F}$, θ_t and θ_d is defined as:

$$\mathcal{E}(\theta_\mathcal{F}, \theta_t, \theta_d) = \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_t(\theta_t, \theta_\mathcal{F})^i - \lambda(\frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_d(\theta_d, \theta_\mathcal{F})^i + \frac{1}{m}\sum_{j=1}^{m} \mathcal{L}_d(\theta_d, \theta_\mathcal{F})^j), \tag{17}$$

where $\mathcal{L}_t$ denotes the loss function for the source domain prediction task, $\mathcal{L}_d$ denotes the loss function for the domain classification, λ is a balance factor, n and m indicate the number of the source domain data and target domain data, respectively. The parameters $\theta_\mathcal{F}$, θ_t and θ_d and then are updated using gradient descent:

$$\begin{aligned} \theta_\mathcal{F} &\longleftarrow \theta_\mathcal{F} - \beta(\frac{\partial \mathcal{L}_t}{\partial \theta_\mathcal{F}} - \lambda(\frac{\partial \mathcal{L}_d}{\partial \theta_\mathcal{F}})), \\ \theta_t &\longleftarrow \theta_t - \beta\frac{\partial \mathcal{L}_t}{\partial \theta_t}, \\ \theta_d &\longleftarrow \theta_d - \beta\lambda\frac{\partial \mathcal{L}_d}{\partial \theta_d}, \end{aligned} \tag{18}$$

where β is the learning rate. We provide an illustration of data and gradient flow of DANN in Figure 7.

Côté-Allard et al. [113] proposed to use DANN for multi-domain for inter-session EMG transfer learning. During training, each mini-batch contains randomly sampled EMG segments from one session. Each mini-batch is assigned a class index indicating different sessions for the domain predicting

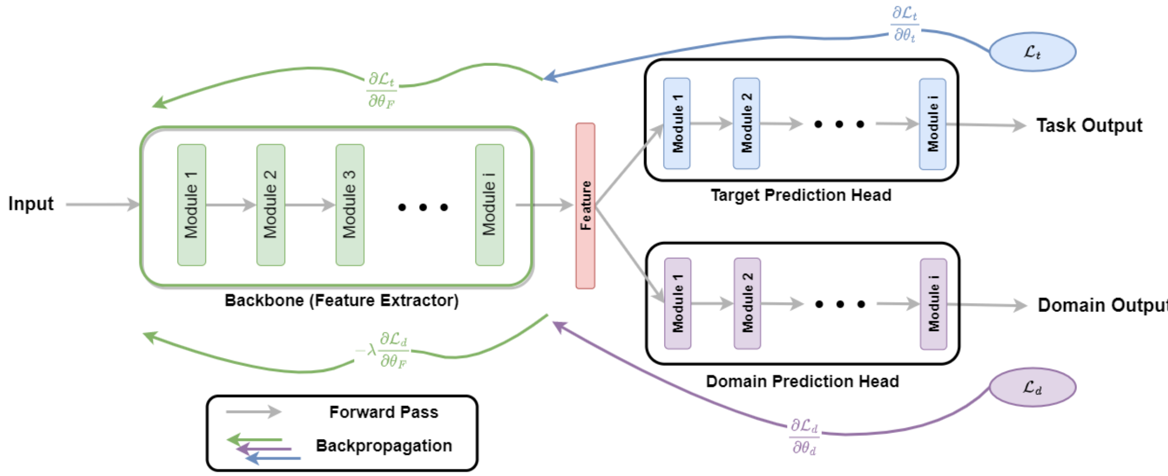

Fig. 7. Illustration of a typical DANN. A backbone of any arbitrary design for feature extraction is marked in green, while the task prediction head and domain prediction head are marked in blue and purple, respectively. The output deep feature from the backbone is fed to both heads for loss calculation with respect to the ground truth label. The gradient of $\mathcal{L}_t$ is backpropagated through the task prediction head and the backbone for parameter update. The domain prediction head is updated by the gradient of $\mathcal{L}_d$. The negative gradient from $\mathcal{L}_d$ also flows back to the backbone for parameter update.

labels. A gradient reversal layer [104] is adopted for easy implementation of negative gradient flow from the domain prediction loss to the backbone. Note that the task prediction head is only updated with loss from the source domain data. In a contemporaneous work, Côté-Allard et al. [114] also explored using Virtual Adversarial Domain Adaptation (VADA) [115] together with Decision-boundary Iterative Refinement Training with a Teacher (DIRT-T) [115] for adversarial based EMG transfer learning. VADA is an extension of DANN that incorporates locally-Lipschitz constraint via Virtual Adversarial Training (VAT) [116] to punish the violation of the cluster assumption during training. On top of the trained model by VADA, DIRT-T aims to optimize the decision boundary on the target domain data by fine-tuning the model. In specific, the model parameter from the previous iteration is treated as the teacher model, the optimization goal is to seek a student model that is close to the teacher model while minimizing the cluster assumption violation. Following the work of Côté-Allard et al., other DANN related EMG transfer learning research endeavors [105], [117] were made for various transfer scenarios.

Han et al. [118] further proposed Disentangled Adversarial Autoencoder (DAA), which disentangles the learned latent representation into adversary and nuisance blocks to model task-related features and domain-related features disjointly. Based on the autoencoder (AE) [119] structure, the encoder $\mathcal{F}(\cdot;\theta)$ maps the input signal x into a latent representation $z = [z_a, z_n]$ where z_a and z_n stand for the adversary and the nuisance sub-representation, respectively. z_a is expected to contain only the task relevant feature but no domain-specific information i_d. On the other hand, the encoder embeds sufficient domain-specific data into z_n. The decoder $\mathcal{G}(\cdot;\eta)$ reconstructs the original input signal based on latent representation z. Similar to DANN, DAA also adopts two prediction heads: adversarial prediction head $\mathcal{P}_a(\cdot;\phi)$ and nuisance prediction head $\mathcal{P}_n(\cdot;\psi)$. Formally, the overall loss to train DAA is defined as:

$$\mathcal{L}(\theta,\phi,\psi,\eta) = -\lambda_n \mathbb{E}[\log p(i_d|z_n)] + \lambda_a \mathbb{E}[\log p(i_d|z_a)] + \mathbb{E}[\|x - \mathcal{G}(\mathcal{F}(x))\|^2], \tag{19}$$

where p stands for the likelihood. As illustrated in Figure 8, the decoder, adversarial prediction head, and nuisance prediction head are discarded after the disentangled feature learning process of DAA. The weight of the encoder is then frozen for feature extraction, and a task prediction head with random weight initialization is placed on top of the encoder for specific downstream tasks. Based on their previous work [118], Han et al. later proposed a soft version of the latent representation disentanglement [120].

IV. Summary of Common Benchmarks

We summarize common EMG datasets [6], [48], [51], [108], [109], [110], [111], [112], [113] that could be used for transfer learning and provide dataset statistics in Table I, including task category, number of subjects, number of recording device channel, sampling frequency, number of gesture classes, and corresponding citations.

V. Experimental Performance Results

For transfer scenarios introduced in Section II-B, modality variation and extensive learning mainly address specific customized application cases with in-house data, no existing works are with identical experimental configurations and are thus hard to compare. Studies focusing on the scenario of electrode variation [8], [9] adopt different data acquisition protocols and shift the electrodes in different manners (shift by angle or distance). Thus, we demonstrate the performance of representative approaches that adopt publicly available benchmarks and omit the works that utilize in-house datasets. We provide the performance comparison on benchmarks that are selected by at least five works. The reported metrics

TABLE I
SUMMARY AND STATISTICS OF COMMON EMG DATASETS FOR TRANSFER LEARNING

Dataset	Category	# Subject	# Channels	Sampling Frequency	# Class	Citation
In-house Data	-	-	-	-	-	[56], [19], [8], [33], [22], [37], [20], [36], [35], [21], [34], [47], [70], [68], [72], [71], [62], [63], [81], [84], [98], [102], [105], [106] [9],
NinaPro DB1 [6]	Hand Gesture Recognition	27	10	100 HZ	53	[90], [82], [85], [107]
NinaPro DB2 [6]	Hand Gesture Recognition	40	12	2000 HZ	50	[46], [67], [73], [83], [86], [78], [96], [98], [93]
NinaPro DB3 [6]	Hand Gesture Recognition	11	12	2000 HZ	50	[46], [67], [73], [83], [78], [93]
NinaPro DB4 [108]	Hand Gesture Recognition	10	12	2000 HZ	53	[46], [67]
NinaPro DB5 [108]	Hand Gesture Recognition	10	16	200 HZ	53	[27], [48], [46], [90], [87]
NinaPro DB6 [109]	Hand Gesture Recognition	10	16	2000 HZ	8	[46], [78]
Myo Dataset [48]	Hand Gesture Recognition	36	8	200 HZ	7	[48], [49]
CSL-HDEMG [110]	Hand Gesture Recognition	5	192	2048 HZ	27	[51], [90]
CapgMyo Database [51]	Hand Gesture Recognition	23	128	1000 HZ	7	[51], [90], [82], [85], [107]
SS-STM [111]	Hand Gesture Recognition	25	8	200 HZ	22	[74]
Multiple Speed Walking [112]	Muscle Force Prediction	8	8	1080 HZ	-	[69]
Physical Action Dataset	Physical Action Recognition	4	8	10000 HZ	20	[64]
LongTermEMG [113]	Hand Gesture Recognition	20	10	1000 HZ	11	[113], [114]

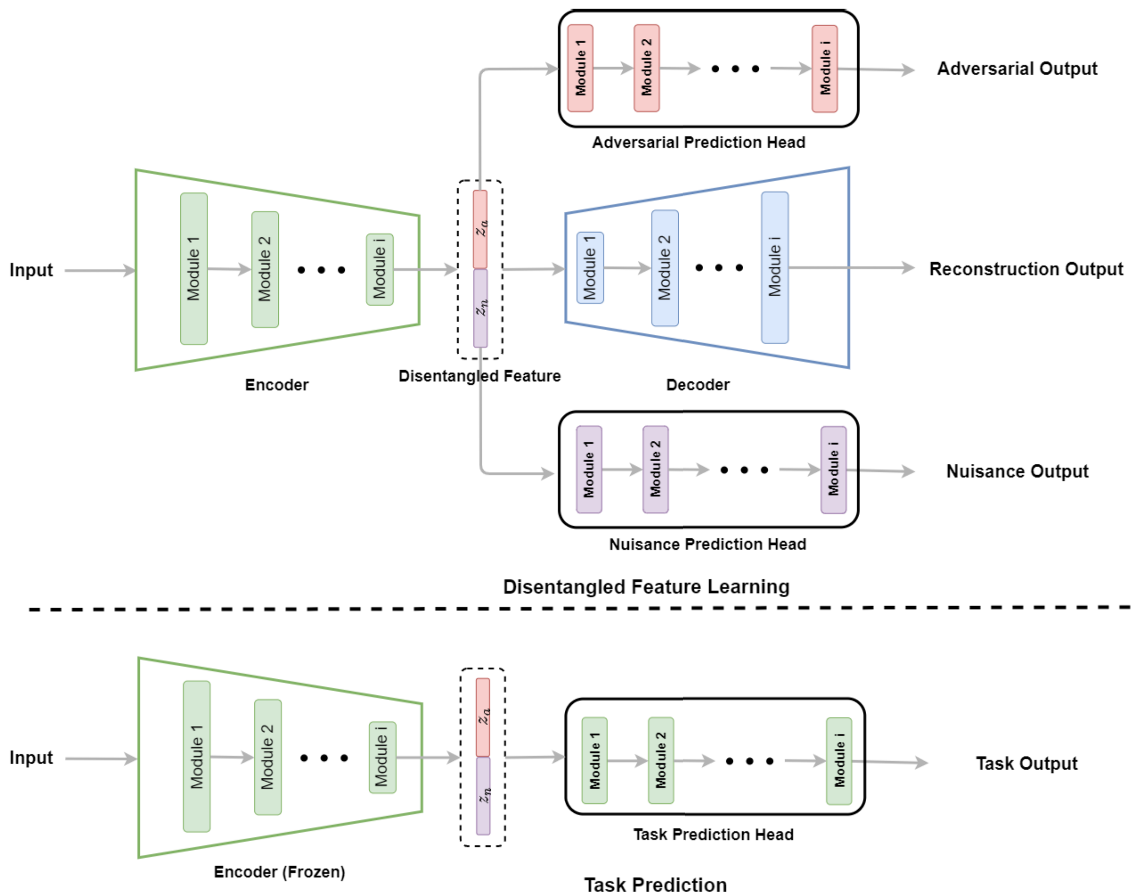

Fig. 8. Illustration of Disentangled Adversarial Autoencoder (DAA). The disentangled feature learning phase is demonstrated above the dotted line, while the task prediction phase is shown below the dotted line. In the disentangled feature learning phase, the input data is mapped into disentangled feature representation z_a and z_n, with each passed to the corresponding prediction head. The overall latent representation is passed to the decoder for signal reconstruction. After feature learning, two prediction heads with the decoder are discarded. A new task prediction head with random weights is introduced on top of the encoder with frozen weight for task prediction.

are taken directly from the original papers. We report the classification result on NinaProDB2 for the **inter-subject** and compare the performance for the **inter-session** scenario on the CapgMyo database. Please note that the recording interval

TABLE II
PERFORMANCE COMPARISON OF EXISTING WORKS WITH SIMILAR EXPERIMENTAL SETUP

Method	Trasnfer Senario	Dataset Used	Window Size	Input	Accuracy (%)
TL-MKCNN [46]	Inter-subject	NinaProDB2	100ms	Raw sEMG	65.3
Lehmler *et al.* [67]	Inter-subject	NinaProDB2	200ms	Raw sEMG	**68.0**
Supportive CNN [73]	Inter-subject	NinaProDB2	200ms	Feature	52.5
Marano *et al.* [78]	Inter-subject	NinaProDB2	200ms	Feature	**68.0**
Fan *et al.* [83]	Inter-subject	NinaProDB2	100ms	Feature	64.1
Du *et al.* [51]	Inter-session	CapgMyo Db-b	150ms	Raw sEMG	88.5
2SRNN [82]	Inter-session	CapgMyo Db-b	150ms	Raw sEMG	83.8
All-ConvNet+TL [107]	Inter-session	CapgMyo Db-b	150ms	Raw sEMG	**94.9**

between the two sessions in Db-b of CapgMyo is greater than one week, and the electrode array placement varies. Consequently, this benchmark can also reflect the effectiveness of various approaches on the **electrodes variation** scenario. Specifically, we compare the performance of five representative methods for the **inter-subject** and three classical works for the **inter-session** in Table II since the compared works are with similar experimental setups. It must be emphasized that although publicly available benchmarks are adopted, data pre-processing techniques and dataset split vary across different works leading to comparisons not entirely fair. The best performance for each transfer scenario on each dataset is marked in bold in the table. It is worth noticing that Du et al., 2SRNN and All-ConvNet+TL all adopted the parameter sharing approach. The advantage of All-ConvNet+TL over the other two approaches might be attributed to a better network structure and careful tuning of which parameters to share. For the inter-subject transfer scenario, all methods yield comparable performance except Supportive CNN. Supportive CNN is a model ensemble based approach that ensemble models from different subjects and thus is very sensitive to subject variation across subjects. We assume that the worse performance of Supportive CNN on NinaProDB2 can be explained by huge subject variation across subjects.

VI. DISCUSSION AND FUTURE DIRECTIONS

In this section, we revisit EMG transfer learning approaches based on our categorization and discuss the advantages and drawbacks of each category. Given our discussion, we further point out future directions.

a) Instance Weighting: By applying weights to the data samples from the source domain, instance weighting techniques leverage existing source domain data to enhance the training of models on the target domain. These methods serve to alleviate the issue of data scarcity when the available target domain data is limited. The use of instance weighting enables the augmentation of the target domain dataset, effectively enlarging its size and providing more diverse examples for model training. While instance weighting can be beneficial, it is important to consider its potential drawbacks. One such drawback is that the overall performance of these methods heavily relies on the similarities between the source and target domains. If the source and target domains differ significantly, the target model may suffer from poorly selected and weighted samples from the source domain. This misalignment can lead to suboptimal model performance on the target domain data, as the weights assigned to the source domain samples may not accurately reflect their relevance or representativeness for the target domain.

b) Linear Feature Transformation: Linear feature transformation-based approaches are considered bio-inspired due to their ability to capture the essence of electromyography (EMG) generation and recording through linear assumptions. This approach is particularly advantageous as it is simple to implement and computationally efficient. The transfer process involves applying a linear transformation either to the data or features, which can sometimes be easily accomplished through matrix multiplication. We contend that the linear assumption is applicable in transfer scenarios where there is a correlation between electrode shifts. As discussed in Section II-C, certain non-linear factors, such as the filtering effects of muscle and fat tissues, as well as variations in muscle fiber recruitment patterns across subjects, cannot be adequately modeled using a linear transformation. However, in cases where the subjects and recording devices remain constant, the linear feature transformation approaches can partially capture the effects of electrode shifts. By assuming linearity in electrode shifts, these methods exploit the consistency in the EMG signals generated by the same subject using the same recording devices. Although other non-linear factors are present, these approaches primarily focus on capturing the linear relationship associated with electrode displacement. This assumption allows for a simplified and computationally lightweight transfer process, making these methods highly practical in certain contexts. It is important to note that while linear feature transformation-based approaches offer simplicity and efficiency, they may not fully account for all the complexities of EMG data. Therefore, when applying these methods, researchers should be aware of the limitations imposed by the linear assumption and carefully consider the specific characteristics and nuances of their EMG datasets to ensure the validity and applicability of the transfer learning process.

c) Non-linear Feature Transformation: The non-linearity of this line of work mainly comes from the non-linear activation functions of DNNs. Moreover, DNN-based approaches in transfer learning also benefit from the inherent advantages of DNNs, such as their robust feature extraction abilities. Deep neural networks have the capacity to automatically extract high-level representations and features from raw data, enabling them to discover meaningful patterns and discriminative features that contribute to reducing data distribution discrepancies between domains. Consequently, the non-linear factors, such as subject variation, can be modeled in a black-box fashion. However, one main drawback of DNN-based non-linear transformations is the lack of interpretability. The black-box nature of DNN architectures makes it challenging to understand precisely which features are being extracted and how they contribute to reducing the differences between domains. The lack of interpretability can hinder further algorithmic improvements, as there is no clear biological or intuitive understanding behind the design of the network architecture. Therefore, it's hard to further improve the algorithm since no biological sound clue resides behind the design of the architecture. Addressing the interpretability challenge in DNN-based transfer learning remains an active area of research.

d) Parameter Fine-tuning: Fine-tuning is a straightforward and commonly used approach in transfer learning, as it involves retraining a pre-trained model on the target domain dataset. This simplicity is advantageous in practice, as it requires minimal modifications to the existing model architecture and training process. However, there are challenges associated with fine-tuning, particularly when the data size of the target domain is limited. One major concern is the potential for *overfitting*. When the target domain dataset is small, the fine-tuned model may overly adapt to the specific characteristics and idiosyncrasies of the limited target domain data, leading to reduced generalization performance on unseen data. Another challenge with fine-tuning is the phenomenon known as catastrophic forgetting. As the model is retrained on the target domain data, it has a tendency to rapidly forget the knowledge it acquired during pre-training on the source domain. This can be problematic if the source domain knowledge is still relevant and beneficial for the target domain task.

e) Parameter Sharing: Parameter sharing based approaches are quite similar to fine-tuning, with the distinction that only a subset of network parameters is shared between the source and target models. This approach can help alleviate the issue of *catastrophic forgetting* by retaining certain knowledge through parameter sharing. In parameter sharing-based approaches, it is common practice to share the parameters of the feature extractor while training a task-specific prediction head from scratch. By sharing the feature extraction layers, which capture general and domain-invariant representations, the transferred model can leverage the learned knowledge from the source domain while adapting to the target domain. This allows the model to retain relevant information and avoid completely forgetting during the transfer process. Freezing the shared backbone (feature extraction layers) is often employed when the source domain is believed to be large and has a similar distribution to the target dataset. This practice ensures that the transferred model focuses on learning the task-specific details from the target domain, without altering the well-established feature extraction layers. Freezing the backbone can help stabilize the transfer process and prevent the model from deviating too much from the source domain knowledge. However, it is important to note that freezing the backbone is not always the optimal strategy. Choosing the appropriate strategy for parameter sharing and freezing depends on the specific characteristics of the source and target domains, including their size, distribution, and similarity. It requires careful consideration and experimentation to strike the right balance between leveraging the source domain knowledge and allowing the model to adapt to the target domain for optimal transfer performance.

f) Model Ensemble: Directly combining data of multiple domains might lead to the neural network not converging smoothly due to data distribution differences. To overcome this obstacle and preserve domain-specific information, a more effective approach is to construct individual models tailored to each domain and then combine them through an ensemble technique. By building separate models for each domain, the ensemble approach ensures that the unique information and characteristics of each domain are captured and preserved. This allows the models to focus on learning the specific nuances and patterns within their respective domains, enhancing the transfer of knowledge from each domain to the target task. In the context of EMG applications, it is often assumed that data distributions can vary significantly across different sessions or subjects. These variations can stem from factors such as muscle physiology, electrode placement, and subject-specific characteristics. Therefore, an ensemble of models is particularly advantageous in this scenario, as it promotes diversity among the models. Each model in the ensemble can capture a different aspect or variation of the data, leading to a comprehensive representation of the underlying patterns and improving the overall performance of the ensemble. However, it is important to consider the computational and memory costs associated with model ensembling. As multiple models are involved, each with its own set of parameters, storing them in memory can be demanding. Additionally, during the prediction phase, each data point needs to be processed multiple times, once by each model in the ensemble, to generate the final prediction. This can increase the computational overhead and impact the real-time performance of the system.

g) Model Structure Calibration: The existing model structure calibration-based approaches primarily rely on random forest models, which inherently involve model ensembling. Therefore, they share similar advantages with other ensemble-based methods. Model structure calibration in this context refers to the adjustment of individual decision trees through growing or pruning operations. One drawback of calibration-based models is that feature extraction needs to be performed manually, which is also a limitation of decision trees themselves. This manual feature extraction process can be time-consuming and may require domain expertise. To further advance the field, it would be intriguing to explore the potential of calibrating the model structure of deep neural networks (DNNs) using neural network structure searching tools such as Neural Architecture Search (NAS). NAS algorithms have shown promise in automatically discovering optimal neural network architectures for various tasks. By applying NAS techniques to the calibration of DNN model structures, we can potentially enhance the transfer learning performance by optimizing the network's architecture specifically for the target domain.

h) Label Calibration: This line of work involves using the source model to label unseen data from the target domain, followed by label calibration to refine the target domain labels and update the model. This approach offers several advantages that make it well-suited for real-world applications. One key advantage is that it eliminates the need for manual labeling of target domain data by domain experts. Instead, the source model's knowledge is leveraged to automatically assign labels to the target domain data, making the transfer process more practical and efficient. Furthermore, the transferring mechanism of these methods allows for easy deployment on end devices and can be seamlessly applied to new incoming data through a simple user interface. This makes it highly suitable for real-time or dynamic environments where continuous adaptation to new data is necessary. However, since the source domain model labels the target domain data based on the knowledge it has learned from the source domain, there is a possibility of introducing label noise. This means that the labels assigned to the target domain data may not always accurately reflect the true labels or categories of the target domain. The label calibration procedure aims to mitigate this issue by refining the target domain labels, but it may not completely eliminate the potential for label noise.

i) Data Generation: Generating synthetic EMG data can alleviate the tedious workload associated with data collection and annotation. The time-consuming nature of EMG data collection and labeling, which requires expertise, makes the generation of high-quality synthetic data appealing in terms of practicality and efficiency. Synthetic data can provide a larger and more diverse dataset for training models, enabling better generalization and performance in real-world applications. By generating data that covers a wide range of scenarios and variations, including different muscle activations, signal noise, and electrode positions, the models can learn to adapt to various conditions and improve their robustness. However, there are unique challenges when it comes to evaluating the quality of synthetic EMG data compared to other domains like vision or language. In vision or language tasks, the quality of generated images or texts can be easily assessed through human observation. However, evaluating the quality of EMG signals generated is more complex and subjective. It is crucial to ensure that the synthetic data accurately represents the characteristics and statistical properties of real EMG signals. Poorly generated data can have a negative impact on the transfer learning process. Models trained on such data may not generalize well to real-world scenarios, leading to poor performance and unreliable results.

j) Meta Learning Based: Meta-learning approaches excel in quickly adapting to new tasks with limited labeled data. By leveraging knowledge from related tasks or domains, meta-learning enables models to rapidly learn and generalize to new EMG tasks. This is particularly beneficial in scenarios where collecting labeled data for each specific task is time-consuming or impractical. This makes meta-learning approaches well-suited for few-shot learning scenarios. However, meta-learning relies on the assumption that the source tasks or domains are similar to the target task or domain. If the dissimilarity between tasks or domains is significant, the transferability of learned knowledge may be limited. Also, meta-learning often requires substantial computational resources, including memory and processing power, to train and update models. End devices with limited resources, such as smartphones or wearable devices, may struggle to handle the computational demands of meta-learning algorithms.

k) Adversarial Learning Based: EMG signals can exhibit variations due to factors like electrode placement, muscle morphology, and individual differences. Adversarial approaches aim to learn domain-invariant representations, allowing models to focus on the underlying patterns and characteristics of EMG signals rather than being influenced by domain-specific variations. This helps in generalizing well to different subjects, sessions, or conditions, thereby enhancing the transferability of learned knowledge. This enables the transfer of knowledge across domains, even when they exhibit substantial distribution differences. However, adversarial training involves balancing the trade-off between the adversarial loss and the task-specific loss. Finding appropriate hyperparameter settings for effective training can be challenging, and suboptimal choices may lead to poor performance or instability during training. More importantly, adversarial-based methods often require additional components, such as domain classifiers or adversarial networks, leading to more complex training procedures. This complexity can increase the computational and memory requirements, making the training process more time-consuming and resource-intensive.

Based on the advantages and drawbacks discussed for each transfer learning approach in the context of EMG applications, along with the requirements of real-world EMG applications, the transfer learning algorithm should bear the following characteristics:

1) **Bio-Inspired.** The working mechanism of muscles is relatively well studied and straightforward compared to that of the brain. We point out that the activation patterns of the muscles, relative location between muscles and electrodes, and individual biological characteristics should be explicitly modeled into the neural network to embed the network with A priori knowledge. AlphaFold [121] is a successful attempt at protein structure prediction with protein A priori knowledge guided network structure design.
2) **Hardware-friendly.** Ideally, the re-calibration should be done on end devices rather than on cloud servers. With wearable or even implantable devices, memory and computation resources are highly restricted. Most current DNN based transfer learning approaches fail to take the hardware constraints into consideration. Future works should incorporate a hardware resource perspective into algorithm design (hardware-software co-design).
3) **User-friendly.** The transfer learning algorithm should effectively support real-time processing and be lightweight in terms of data collection procedures. This ensures its feasibility and suitability for real-world applications, enabling timely and efficient decision-making without imposing excessive data requirements or user involvement. Future works, thus, should put more attention on transfer learning algorithms that work with limited target domain data and annotation. For instance, given a hand gesture classification task with more than 20 classes, the algorithm is considered user-friendly if the user is required to perform the most simple gesture once for system re-calibration.

References

[1] M. Ghassemi et al., "Development of an EMG-controlled serious game for rehabilitation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 2, pp. 283–292, Feb. 2019.

[2] J. Qi, G. Jiang, G. Li, Y. Sun, and B. Tao, "Surface EMG hand gesture recognition system based on PCA and GRNN," *Neural Comput. Appl.*, vol. 32, no. 10, pp. 6343–6351, May 2020.

[3] T. S. Saponas, D. S. Tan, D. Morris, R. Balakrishnan, J. Turner, and J. A. Landay, "Enabling always-available input with muscle-computer interfaces," in *Proc. 22nd Annu. ACM Symp. User Interface Softw. Technol.*, Oct. 2009, pp. 167–176.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1097–1105.

[5] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[6] S. Pizzolato, L. Tagliapietra, M. Cognolato, M. Reggiani, H. Müller, and M. Atzori, "Comparison of six electromyography acquisition setups on hand movement classification tasks," *PLoS ONE*, vol. 12, no. 10, Oct. 2017, Art. no. e0186132.

[7] J. Yousefi and A. Hamilton-Wright, "Characterizing EMG data using machine-learning tools," *Comput. Biol. Med.*, vol. 51, pp. 1–13, Aug. 2014.

[8] A. Ameri, M. A. Akhaee, E. Scheme, and K. Englehart, "A deep transfer learning approach to reducing the effect of electrode shift in EMG pattern recognition-based control," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 2, pp. 370–379, Feb. 2020.

[9] S. Muceli, N. Jiang, and D. Farina, "Extracting signals robust to electrode number and shift for online simultaneous and proportional myoelectric control by factorization algorithms," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 3, pp. 623–633, May 2014.

[10] R. S. Woodworth and E. Thorndike, "The influence of improvement in one mental function upon the efficiency of other functions. (I)," *Psychol. Rev.*, vol. 8, no. 3, p. 247, 1901.

[11] D. Xiong, D. Zhang, X. Zhao, and Y. Zhao, "Deep learning for EMG-based human-machine interaction: A review," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 3, pp. 512–533, Mar. 2021.

[12] F. Zhuang et al., "A comprehensive survey on transfer learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, Jan. 2021.

[13] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[14] T. Joachims et al., "Transductive inference for text classification using support vector machines," in *ICML*, vol. 99, 1999, pp. 200–209.

[15] A. Fleming, N. Stafford, S. Huang, X. Hu, D. P. Ferris, and H. H. Huang, "Myoelectric control of robotic lower limb prostheses: A review of electromyography interfaces, control paradigms, challenges and future directions," *J. Neural Eng.*, vol. 18, no. 4, Aug. 2021, Art. no. 041004.

[16] D. Wu, S. Li, J. Yang, and M. Sawan, "neuro2vec: Masked Fourier spectrum prediction for neurophysiological representation learning," 2022, *arXiv:2204.12440*.

[17] J. R. Daube and D. I. Rubin, "Needle electromyography," *Muscle Nerve, Off. J. Amer. Assoc. Electrodiagnostic Med.*, vol. 39, no. 2, pp. 244–270, 2009.

[18] H. J. Hermens, B. Freriks, C. Disselhorst-Klug, and G. Rau, "Development of recommendations for SEMG sensors and sensor placement procedures," *J. Electromyogr. Kinesiol.*, vol. 10, no. 5, pp. 361–374, Oct. 2000.

[19] Y. Lin, R. Palaniappan, P. De Wilde, and L. Li, "A normalisation approach improves the performance of inter-subject sEMG-based hand gesture recognition with a ConvNet," in *Proc. 42nd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2020, pp. 649–652.

[20] M. M.-C. Vidovic, H. Hwang, S. Amsüss, J. M. Hahne, D. Farina, and K. Müller, "Improving the robustness of myoelectric pattern recognition for upper limb prostheses by covariate shift adaptation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 9, pp. 961–970, Sep. 2016.

[21] J. Liu, X. Sheng, D. Zhang, J. He, and X. Zhu, "Reduced daily recalibration of myoelectric prosthesis classifiers based on domain adaptation," *IEEE J. Biomed. Health Informat.*, vol. 20, no. 1, pp. 166–176, Jan. 2016.

[22] S. Kanoga and A. Kanemura, "Assessing the effect of transfer learning on myoelectric control systems with three electrode positions," in *Proc. IEEE Int. Conf. Ind. Technol. (ICIT)*, Feb. 2018, pp. 1478–1483.

[23] P. Kaufmann, K. Englehart, and M. Platzner, "Fluctuating EMG signals: Investigating long-term effects of pattern matching algorithms," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol.*, Aug. 2010, pp. 6357–6360.

[24] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, vol. 2. New York, NY, USA: Springer, 2009.

[25] Y. Al-Assaf, "Surface myoelectric signal analysis: Dynamic approaches for change detection and classification," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 11, pp. 2248–2256, Nov. 2006.

[26] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.

[27] S. Kanoga, T. Hoshino, and H. Asoh, "Subject-transfer framework with unlabeled data based on multiple distance measures for surface electromyogram pattern recognition," *Biomed. Signal Process. Control*, vol. 74, Apr. 2022, Art. no. 103522.

[28] A. B. Ajiboye and R. F. Weir, "Muscle synergies as a predictive framework for the EMG patterns of new hand postures," *J. Neural Eng.*, vol. 6, no. 3, Jun. 2009, Art. no. 036004.

[29] C. W. Antuvan, F. Bisio, F. Marini, S.-C. Yen, E. Cambria, and L. Masia, "Role of muscle synergies in real-time classification of upper limb motions using extreme learning machines," *J. Neuroeng. Rehabil.*, vol. 13, no. 1, pp. 1–15, Dec. 2016.

[30] N. Jiang, K. B. Englehart, and P. A. Parker, "Extracting simultaneous and proportional neural control information for multiple-DOF prostheses from the surface electromyographic signal," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 4, pp. 1070–1080, Apr. 2009.

[31] A. Cichocki, R. Zdunek, A. H. Phan, and S.-I. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-Way Data Analysis and Blind Source Separation*. Hoboken, NJ, USA: Wiley, 2009.

[32] D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 13, 2000, pp. 1–7.

[33] X. Jiang, B. Bardizbanian, C. Dai, W. Chen, and E. A. Clancy, "Data management for transfer learning approaches to elbow EMG-torque modeling," *IEEE Trans. Biomed. Eng.*, vol. 68, no. 8, pp. 2592–2601, Aug. 2021.

[34] S. Y. Günay, M. Yarossi, D. H. Brooks, E. Tunik, and D. Erdoğmuş, "Transfer learning using low-dimensional subspaces for EMG-based classification of hand posture," in *Proc. 9th Int. IEEE/EMBS Conf. Neural Eng. (NER)*, Mar. 2019, pp. 1097–1100.

[35] C. Prahm, B. Paassen, A. Schulz, B. Hammer, and O. Aszmann, "Transfer learning for rapid re-calibration of a myoelectric prosthesis after electrode shift," in *Converging Clinical and Engineering Research on Neurorehabilitation II*. New York, NY, USA: Springer, 2017, pp. 153–157.

[36] C. Prahm et al., "Counteracting electrode shifts in upper-limb prosthesis control via transfer learning," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 5, pp. 956–962, May 2019.

[37] B. Paaßen, A. Schulz, J. Hahne, and B. Hammer, "Expectation maximization transfer learning and its application for bionic hand prostheses," *Neurocomputing*, vol. 298, pp. 122–133, Jul. 2018.

[38] P. Schneider, M. Biehl, and B. Hammer, "Adaptive relevance matrices in learning vector quantization," *Neural Comput.*, vol. 21, no. 12, pp. 3532–3561, Dec. 2009.

[39] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Process. Mag.*, vol. 13, no. 6, pp. 47–60, Nov. 1996.

[40] K. M. Borgwardt, A. Gretton, M. J. Rasch, H.-P. Kriegel, B. Schölkopf, and A. J. Smola, "Integrating structured biological data by kernel maximum mean discrepancy," *Bioinformatics*, vol. 22, no. 14, pp. e49–e57, Jul. 2006.

[41] A. Berlinet and C. Thomas-Agnan, *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. New York, NY, USA: Springer, 2011.

[42] I. Dagan, L. Lee, and F. Pereira, "Similarity-based methods for word sense disambiguation," 1997, *arXiv:cmp-lg/9708010*.

[43] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 1–8.

[44] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 539–546.

[45] D. Wu, J. Yang, and M. Sawan, "Bridging the gap between patient-specific and patient-independent seizure prediction via knowledge distillation," *J. Neural Eng.*, vol. 19, no. 3, Jun. 2022, Art. no. 036035.

[46] Y. Zou and L. Cheng, "A transfer learning model for gesture recognition based on the deep features extracted by CNN," *IEEE Trans. Artif. Intell.*, vol. 2, no. 5, pp. 447–458, Oct. 2021.

[47] T. Bao, S. A. R. Zaidi, S. Xie, P. Yang, and Z. Zhang, "Inter-subject domain adaptation for CNN-based wrist kinematics estimation using sEMG," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1068–1078, 2021.

[48] U. Côté-Allard et al., "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 4, pp. 760–771, Apr. 2019.

[49] U. Côté-Allard, C. L. Fall, A. Campeau-Lecours, C. Gosselin, F. Laviolette, and B. Gosselin, "Transfer learning for sEMG hand gestures recognition using convolutional neural networks," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2017, pp. 1663–1668.

[50] A. A. Rusu et al., "Progressive neural networks," 2016, *arXiv:1606.04671*.

[51] Y. Du, W. Jin, W. Wei, Y. Hu, and W. Geng, "Surface EMG-based inter-session gesture recognition enhanced by deep domain adaptation," *Sensors*, vol. 17, no. 3, p. 458, Feb. 2017.

[52] Y. Li, N. Wang, J. Shi, J. Liu, and X. Hou, "Revisiting batch normalization for practical domain adaptation," 2016, *arXiv:1603.04779*.

[53] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[54] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," *J. Stat. Planning Inference*, vol. 90, no. 2, pp. 227–244, Oct. 2000.

[55] J. Huang, A. Gretton, K. Borgwardt, B. Schölkopf, and A. Smola, "Correcting sample selection bias by unlabeled data," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 19, 2006, pp. 1–8.

[56] Q. Li, A. Zhang, Z. Li, and Y. Wu, "Improvement of EMG pattern recognition model performance in repeated uses by combining feature selection and incremental transfer learning," *Frontiers Neurorobot.*, vol. 15, p. 87, Jun. 2021.

[57] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for transfer learning," in *Proc. 24th Int. Conf. Mach. Learn. (ICML)*. New York, NY, USA: Association for Computing Machinery, Jun. 2007, pp. 193–200, doi: 10.1145/1273496.1273521.

[58] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997.

[59] J. Donahue et al., "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 647–655.

[60] D. Wu, S. Li, Z. Zang, and S. Z. Li, "Exploring localization for self-supervised fine-grained contrastive learning," in *Proc. 33rd Brit. Mach. Vis. Conf. (BMVC)*. London, U.K.: BMVA Press, Nov. 2022, pp. 1–15. [Online]. Available: https://bmvc2022.mpi-inf.mpg.de/0268.pdf

[61] S. Li, D. Wu, F. Wu, Z. Zang, and S. Z. Li, "Architecture-agnostic masked image modeling—From ViT back to CNN," 2022, *arXiv:2205.13943*.

[62] K. Rezaee, S. Savarkar, X. Yu, and J. Zhang, "A hybrid deep transfer learning-based approach for Parkinson's disease classification in surface electromyography signals," *Biomed. Signal Process. Control*, vol. 71, Jan. 2022, Art. no. 103161.

[63] J. J. Bird, J. Kobylarz, D. R. Faria, A. Ekárt, and E. P. Ribeiro, "Cross-domain MLP and CNN transfer learning for biological signal processing: EEG and EMG," *IEEE Access*, vol. 8, pp. 54789–54801, 2020.

[64] F. Demir, V. Bajaj, M. C. Ince, S. Taran, and A. Şengür, "Surface EMG signals and deep transfer learning-based physical action classification," *Neural Comput. Appl.*, vol. 31, no. 12, pp. 8455–8462, Dec. 2019.

[65] C. P. Phoo and B. Hariharan, "Self-training for few-shot transfer across extreme task differences," 2020, *arXiv:2010.07734*.

[66] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "ChestX-ray8: Hospital-scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2097–2106.

[67] S. J. Lehmler, M. Saif-Ur-Rehman, T. Glasmachers, and I. Iossifidis, "Deep transfer-learning for patient specific model re-calibration: Application to sEMG-classification," 2021, *arXiv:2112.15019*.

[68] J. Kobylarz, J. J. Bird, D. R. Faria, E. P. Ribeiro, and A. Ekárt, "Thumbs up, thumbs down: Non-verbal human–robot interaction through real-time EMG classification via inductive and supervised transductive transfer learning," *J. Ambient Intell. Hum. Comput.*, vol. 11, no. 12, pp. 6021–6031, Dec. 2020.

[69] T. T. Dao, "From deep learning to transfer learning for the prediction of skeletal muscle forces," *Med. Biol. Eng. Comput.*, vol. 57, no. 5, pp. 1049–1058, May 2019.

[70] S. Tam, M. Boukadoum, A. Campeau-Lecours, and B. Gosselin, "Intuitive real-time control strategy for high-density myoelectric hand prosthesis using deep and transfer learning," *Sci. Rep.*, vol. 11, no. 1, pp. 1–14, May 2021.

[71] J. Sloboda, P. Stegall, R. J. McKindles, L. Stirling, and H. C. Siu, "Utility of inter-subject transfer learning for wearable-sensor-based joint torque prediction models," in *Proc. 43rd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Nov. 2021, pp. 4901–4907.

[72] U. Zakia and C. Menon, "Force myography-based human robot interactions via deep domain adaptation and generalization," *Sensors*, vol. 22, no. 1, p. 211, Dec. 2021.

[73] K.-T. Kim, C. Guan, and S.-W. Lee, "A subject-transfer framework based on single-trial EMG analysis using convolutional neural networks," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 1, pp. 94–103, Jan. 2020.

[74] T. Hoshino, S. Kanoga, M. Tsubaki, and A. Aoyama, "Comparing subject-to-subject transfer learning methods in surface electromyogram-based motion recognition with shallow and deep classifiers," *Neurocomputing*, vol. 489, pp. 599–612, Jun. 2022.

[75] L. S. Penrose, "The elementary statistics of majority voting," *J. Roy. Stat. Soc.*, vol. 109, no. 1, pp. 53–57, 1946.

[76] R. M. French, "Using semi-distributed representations to overcome catastrophic forgetting in connectionist networks," in *Proc. 13th Annu. Cogn. Sci. Soc. Conf.*, vol. 1, 1991, pp. 173–178.

[77] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, Mar. 1986.

[78] G. Marano, C. Brambilla, R. M. Mira, A. Scano, H. Müller, and M. Atzori, "Questioning domain adaptation in myoelectric hand prostheses control: An inter- and intra-subject study," *Sensors*, vol. 21, no. 22, p. 7500, Nov. 2021.

[79] N. Segev, M. Harel, S. Mannor, K. Crammer, and R. El-Yaniv, "Learn on source, refine on target: A model transfer learning framework with random forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 9, pp. 1811–1824, Sep. 2017.

[80] D. M. Hawkins, "The problem of overfitting," *J. Chem. Inf. Comput. Sci.*, vol. 44, no. 1, pp. 1–12, Jan. 2004.

[81] X. Chen, Y. Li, R. Hu, X. Zhang, and X. Chen, "Hand gesture recognition based on surface electromyography using convolutional neural network with transfer learning method," *IEEE J. Biomed. Health Informat.*, vol. 25, no. 4, pp. 1292–1304, Apr. 2021.

[82] I. Ketykó, F. Kovács, and K. Z. Varga, "Domain adaptation for sEMG-based gesture recognition with recurrent neural networks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–7.

[83] J. Fan et al., "Improving sEMG-based motion intention recognition for upper-limb amputees using transfer learning," *Neural Comput. Appl.*, vol. 35, pp. 16101–16111, Jul. 2021.

[84] L. Zhang, D. Soselia, R. Wang, and E. M. Gutierrez-Farewik, "Lower-limb joint torque prediction using LSTM neural networks and transfer learning," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 600–609, 2022.

[85] Z. Yu, J. Zhao, Y. Wang, L. He, and S. Wang, "Surface EMG-based instantaneous hand gesture recognition using convolutional neural network with the transfer learning method," *Sensors*, vol. 21, no. 7, p. 2540, Apr. 2021.

[86] P. Tsinganos, J. Cornelis, B. Cornelis, B. Jansen, and A. Skodras, "Transfer learning in sEMG-based gesture recognition," in *Proc. 12th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Jul. 2021, pp. 1–7.

[87] Y. Li, W. Zhang, Q. Zhang, and N. Zheng, "Transfer learning-based muscle activity decoding scheme by low-frequency sEMG for wearable low-cost application," *IEEE Access*, vol. 9, pp. 22804–22815, 2021.

[88] T. Elsken, J. H. Metzen, and F. Hutter, "Neural architecture search: A survey," *J. Mach. Learn. Res.*, vol. 20, no. 1, pp. 1997–2017, 2019.

[89] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[90] Y. Zhang, Y. Chen, H. Yu, X. Yang, R. Sun, and B. Zeng, "A feature adaptive learning method for high-density sEMG-based gesture recognition," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 1, pp. 1–26, Mar. 2021.

[91] E. Liberty, "Simple and deterministic matrix sketching," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2013, pp. 581–588.

[92] J. MacQueen, "Classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probab.*, 1967, pp. 281–297.

[93] X. Zhai, B. Jelfs, R. H. M. Chan, and C. Tin, "Self-recalibrating surface EMG pattern recognition for neuroprosthesis control based on convolutional neural network," *Frontiers Neurosci.*, vol. 11, p. 379, Jul. 2017.

[94] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: A survey," 2020, *arXiv:2004.05439*.

[95] S. Thrun and L. Pratt, "Learning to learn: Introduction and overview," in *Learning to Learn*. Beijing, China: Springer, 1998, pp. 3–17.

[96] E. Rahimian, S. Zabihi, A. Asif, S. F. Atashzar, and A. Mohammadi, "Few-shot learning for decoding surface electromyography for hand gesture recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 1300–1304.

[97] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1126–1135.

[98] R. A. Zanini and E. L. Colombini, "Parkinson's disease EMG data augmentation and simulation with DCGANs and style transfer," *Sensors*, vol. 20, no. 9, p. 2605, May 2020.

[99] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015, *arXiv:1511.06434*.

[100] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," 2015, *arXiv:1508.06576*.

[101] P. Drineas, M. W. Mahoney, and N. Cristianini, "On the Nyström method for approximating a Gram matrix for improved kernel-based learning," *J. Mach. Learn. Res.*, vol. 6, no. 12, pp. 2153–2175, 2005.

[102] K. Suri and R. Gupta, "Transfer learning for sEMG-based hand gesture classification using deep learning in a master- slave architecture," in *Proc. 3rd Int. Conf. Contemp. Comput. Informat. (IC3I)*, Oct. 2018, pp. 178–183.

[103] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000.

[104] Y. Ganin et al., "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, 2016.

[105] E. Campbell, A. Phinyomark, and E. Scheme, "Deep cross-user models reduce the training burden in myoelectric control," *Frontiers Neurosci.*, vol. 15, p. 595, May 2021.

[106] H. Zhang et al., "Transductive learning models for accurate ambulatory gait analysis in elderly residents of assisted living facilities," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 124–134, 2022.

[107] M. Rabiul Islam, D. Massicotte, P. Y. Massicotte, and W.-P. Zhu, "Surface EMG-based inter-session/inter-subject gesture recognition by leveraging lightweight all-ConvNet and transfer learning," 2023, *arXiv:2305.08014*.

[108] M. Atzori, M. Cognolato, and H. Müller, "Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands," *Frontiers Neurorobot.*, vol. 10, p. 9, Sep. 2016.

[109] F. Palermo, M. Cognolato, A. Gijsberts, H. Müller, B. Caputo, and M. Atzori, "Repeatability of grasp recognition for robotic hand prosthesis control based on sEMG data," in *Proc. Int. Conf. Rehabil. Robot. (ICORR)*, Jul. 2017, pp. 1154–1159.

[110] C. Amma, T. Krings, J. Böer, and T. Schultz, "Advancing muscle-computer interfaces with high-density electromyography," in *Proc. 33rd Annu. ACM Conf. Hum. Factors Comput. Syst.*, Apr. 2015, pp. 929–938.

[111] S. Kanoga, T. Hoshino, and H. Asoh, "Semi-supervised style transfer mapping-based framework for sEMG-based pattern recognition with 1- or 2-DoF forearm motions," *Biomed. Signal Process. Control*, vol. 68, Jul. 2021, Art. no. 102817.

[112] M. Q. Liu, F. C. Anderson, M. H. Schwartz, and S. L. Delp, "Muscle contributions to support and progression over a range of walking speeds," *J. Biomech.*, vol. 41, no. 15, pp. 3243–3252, Nov. 2008.

[113] U. Côté-Allard et al., "A transferable adaptive domain adversarial neural network for virtual reality augmented EMG-based gesture recognition," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 546–555, 2021.

[114] U. Côté-Allard et al., "Unsupervised domain adversarial self-calibration for electromyography-based gesture recognition," *IEEE Access*, vol. 8, pp. 177941–177955, 2020.

[115] R. Shu, H. H. Bui, H. Narui, and S. Ermon, "A DIRT-T approach to unsupervised domain adaptation," 2018, *arXiv:1802.08735*.

[116] T. Miyato, S.-I. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1979–1993, Aug. 2019.

[117] M. H. Sohn, S. Y. Lai, M. L. Elwin, and J. P. A. Dewald, "Feasibility of using wearable EMG armbands combined with unsupervised transfer learning for seamless myoelectric control," *bioRxiv*, Jan. 2022.

[118] M. Han, O. Özdenizci, Y. Wang, T. Koike-Akino, and D. Erdogmus, "Disentangled adversarial autoencoder for subject-invariant physiological feature extraction," *IEEE Signal Process. Lett.*, vol. 27, pp. 1565–1569, 2020.

[119] G. E. Hinton and R. Zemel, "Autoencoders, minimum description length and Helmholtz free energy," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 6, 1993, pp. 1–8.

[120] M. Han, O. Özdenizci, T. Koike-Akino, Y. Wang, and D. Erdogmus, "Universal physiological representation learning with soft-disentangled rateless autoencoders," *IEEE J. Biomed. Health Informat.*, vol. 25, no. 8, pp. 2928–2937, Aug. 2021.

[121] J. Jumper et al., "Highly accurate protein structure prediction with AlphaFold," *Nature*, vol. 596, no. 7873, pp. 583–589, Aug. 2021.