https://doi.org/10.1109/TMRB.2025.3604146

Personalized Myoelectric Control for Upper-Limb Exoskeletons Through Meta-Learning: A Few-Shot Learning Approach

Paniz Sedighi, Xingyu Li, *Member, IEEE*, Vivian K. Mushahwar, *Member, IEEE*, and Mahdi Tavakoli, *Senior Member, IEEE*

***Abstract*—Personalization in the myoelectric control of robotic exoskeletons is crucial to ensuring accurate interpretation and adaptation to the unique muscle activity patterns and movement intentions of each user. This approach minimizes the risk of incorrect or excessive force application, significantly reducing the likelihood of user discomfort or injury during operation. This study introduces a model-agnostic meta-learning approach for personalizing a soft upper-limb exoskeleton in industrial settings. The framework incorporates an attention-based CNN-LSTM model that predicts future angular positions of the robot using EMG and IMU signals. The MAML framework demonstrates significant adaptability and personalization, efficiently predicting future angular positions with minimal data, approximately 20-25 seconds per task. This approach effectively reduces the necessity for extensive retraining with new users or in new environments by 50%, showcasing real-time task adaptation capabilities. Our findings confirmed a reduced human effort of nearly 13% in load-bearing tasks. Also, the results show that the exerted torque from the exoskeleton was 24% higher while maintaining higher accuracy. A comparison with other deep learning models further emphasizes the enhanced adaptability and accuracy offered by the meta-learning approach.**

***Index Terms*—Personalization, Meta-Learning, Attention, Soft Exoskeleton, Proportional Myoelectric Control**

I. Introduction

ROBOTIC exoskeletons have emerged as effective tools for rehabilitation, physical assistance, and improving human strength. Upper-limb exoskeletons are also valuable in industrial settings to reduce the risk of injuries and accidents among workers [1], [2]. Achieving optimal results in tasks that involve load-bearing and shared control requires compliant assistance with the intended movements of the user. Efforts to develop Assist as Needed (AAN) strategies to control exoskeletons are directed toward accomplishing this objective [3]–[6].

In recent years, soft upper-limb exoskeleton robots have gained popularity due to their compatibility with the human body, flexibility in joint movement, and reduced weight, thus requiring less energy and unwanted restrictions [7], [8]. In contrast to lower-limb exoskeletons, accurately modeling human–robot interaction (HRI) for upper limbs is crucial owing to the complexity of predicting arm motion [9]. Surface electromyography (sEMG) has emerged as a suitable method for assessing HRI, capable of capturing subtle muscle activities that are closely linked to movement intention [10], [11]. Another benefit of sEMG is its onset prior to movement, offering potential advantages for predicting future positions for proportional myoelectric control [12].

Xu *et al.* [13] introduced an adaptive impedance controller for a soft exoskeleton using reinforcement learning (RL) that dynamically adjusts impedance parameters during different modes. Xiong *et al.* [14] developed an online learning–based control via iterative learning and impedance adaptation without relying on external sensors.

Recent progress in machine-learning (ML) methods has significantly advanced EMG feature extraction. Although many studies have applied neural networks to myoelectric devices, most focus on basic statistical features for discretised single-joint movements [15], often falling short when interpreting continuous motion intention [16]. Recent multimodal work has demonstrated that transfer-learning pipelines can generalise EMG–IMU force estimation across users with minimal calibration effort [17], highlighting the need for algorithms capable of rapid personalisation.

Deep learning (DL) approaches have delivered higher accuracy for multi-DoF exoskeleton control [18]. Convolutional neural networks (CNNs) capture spatial EMG/IMU features [19]–[21], while recurrent neural networks (RNNs)—particularly long short-term memory (LSTM) networks—exploit temporal dependencies and mitigate vanishing-gradient issues [22], [23].

Despite advances in efficient LSTMs, sequence-to-sequence computation remains demanding. Attention mechanisms, introduced to model dependencies irrespective of position, substantially improve such models [24], [25]. Typically paired with LSTMs to emphasise salient inputs—particularly when EMG crosstalk exists—attention has shown promise for movement decoding [26]. In our previous work [27], we generated CNN feature maps and fed them into an attention-based LSTM network, boosting sequential accuracy and reducing computational burden. The attention module also allowed variable input lengths and robustness to sensor dropout.

Traditional data-driven myoelectric control systems often struggle to account for individual differences in muscle char-

This work was supported by the Canada Foundation for Innovation, UAlberta Huawei-ECE Research Initiative (HERI), Government of Alberta, Natural Sciences and Engineering Research Council (NSERC) of Canada, Canadian Institutes of Health Research (CIHR), and Alberta Economic Development (Corresponding author: Paniz Sedighi.)

P. Sedighi, X. Li, and M. Tavakoli are with the Electrical and Computer Engineering Department, University of Alberta, Edmonton, AB T6G 1HQ, Canada (e-mail: sedighi1@ualberta.ca; xingyu@ualberta.ca; mahdi.tavakoli@ualberta.ca). V. Mushahwar is with the Department of Medicine, University of Alberta, Edmonton, AB T6G 2R3, Canada (email: vivian.mushahwar@ualberta.ca). All authors are members of the Institute for Smart Augmentative and Restorative Technologies and Health Innovations (iSMART), University of Alberta, Edmonton, AB T6G 2R3 (email: smartnet@ualberta.ca).

acteristics and physiological signals [28], resulting in deficient performance [29]. Factors such as variations in sensor placement, the introduction of new movements, and the effects of fatigue present discrepancies that need substantial amounts of data for training [30]. model-agnostic meta-learning (MAML) [31], [32] offers a significant advantage in terms of personalization over domain adaptation [33], transfer learning [17], and RL [34] methods, which often require pre-training on large, domain-specific datasets. This is due to MAML's flexibility and efficiency in quickly adapting to new tasks with minimal data. MAML's unique approach enables a model to learn a generalized representation from various tasks during training, which can then be rapidly fine-tuned with a small amount of data for a specific user or task [35].

In this research, we employed a model-agnostic algorithm for online adaptation to personalize an attention-based CNN-LSTM network. This network predicts the future angular positions of a 3-DoF soft upper-limb exoskeleton using EMG and IMU data. Our system's architecture features a three-tier hierarchy: at the top level, the meta learner; in the middle, the attention model; and at the base, a proportional-derivative (PD) controller coupled with gravity compensation. We demonstrate that leveraging MAML enables the model to rapidly adjust to repeated tasks performed with the exoskeleton, incorporate movements not present in the initial dataset, accommodate new users, and adapt to new users executing unfamiliar movements.

In contrast to current works, this study is the first to personalize and generalize a deep myoelectric decoder across users and motions while running in real time on a three-DoF pneumatic exoskeleton. We achieved (i) a reduction in system calibration time by 50% from 21 s dataset; (ii) a two-step fine-tuning schedule that preserves $< 10ms$ inference latency with 52% improved accuracy compared to conventional methods; (iii) extensive experiments across four challenging scenarios showing 38 % EMG reduction and 24 % torque gain after adaptation. These results demonstrate that meta-learning can bring practical autonomy to hardware where conventional one-time-trained models fail.

II. Methods

Our system was structured as a three-level hierarchical architecture as shown in Fig. 1. We utilized MAML, with an emphasis on K-shot learning in the top layer to optimize the initial parameters for the intention detection model in the middle- level. This optimization ensured that the model could achieve maximal performance on a diverse array of tasks, including new movements or users, by allowing for rapid adaptation after a minimal number of gradient updates. These updates are computed using a small dataset representative of the new task. The bottom level housed the robotic hardware controller coupled with a gravity compensation mechanism that delivers assistance aligned with the user's intent.

A. Position Prediction using the Attention Mechanism

We use an early-fusion strategy: all 21 input channels—nine EMG envelopes, nine IMU gyroscope traces, and three joint angles—are concatenated before any learnable layer. This allows the network to capture cross-modal dependencies (e.g., deltoid EMG bursts aligned with shoulder flexion) by operating on a unified $N \times 21$ input. Unlike late-fusion methods, this setup enables the first convolutional layer to learn compact kernels spanning both muscle activity and kinematics. To ensure alignment, EMG (1.9 kHz), IMU (148 Hz), and encoder (500 Hz) signals are filtered (20–450 Hz for EMG; 5 Hz low-pass for IMU), upsampled to EMG rate, and Z-score normalized. The synchronized stream fills a circular buffer of length $N = 250ms$, from which overlapping windows are sampled every 30 samples ($16ms$). This enables low-latency inference ($< 10ms$) while maintaining dense temporal coverage.

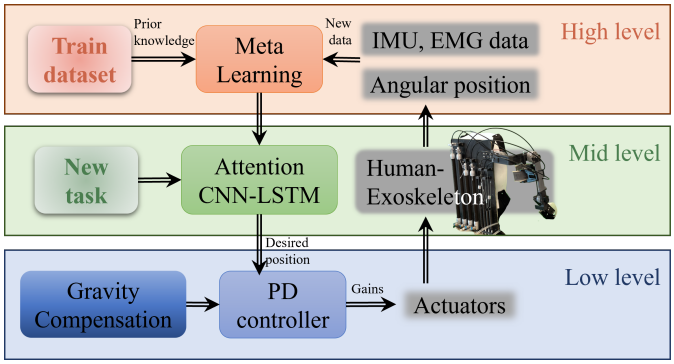


Fig. 1. A three-level hierarchy demonstrating the learning diagram of the model. As new data is introduced through the attention model, meta learner performs a gradient update, while updating its model parameters based on prior knowledge. Then the controller transfers the intended future position commands to the actuators.

The attention mechanism within our model served a dual purpose. Firstly, it improved the specificity of the model's predictions by allowing the decoder to selectively focus on certain features of the input by selecting a subset of all the feature vectors. The feature selection was particularly valuable in the context of minimal training data. Secondly, it ensured that the adaptation process leverages the most relevant information, thereby enhancing the efficiency of the MAML's rapid adjustment capabilities. We proposed a sequence-to-sequence model that mapped the time series of 9 channels of EMG, 9 channels of IMU, and 3 channels of angular position data to the prediction of angular position with a dimension D of DoF (Eqn. 1) for T samples into the future.

$$y = \{\mathbf{y}_1, \ldots, \mathbf{y}_T\}, \mathbf{y}_i \in \mathbb{R}^D \tag{1}$$

The model incorporated an encoder with CNN layers shown in Fig. 2 that were designed to explore temporal correlations within and between channels, thereby reducing the dimensionality of the feature map (Eqn. 2) and the overall complexity of the model.

$$a = \{\mathbf{a}_1, \ldots, \mathbf{a}_K\}, \mathbf{a}_j \in \mathbb{R}^X \tag{2}$$

The decoder processes the feature map from the encoder and the prior hidden and cell states to estimate the future angular position. It receives an input, denoted as $\mathbf{y}_{i'-1}$, which could either be the model's target trajectory or the trajectory

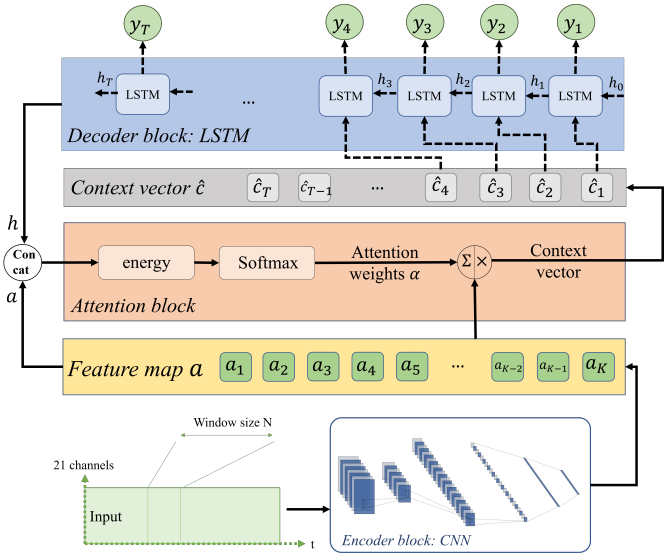


Fig. 2. The architecture of the attention-based CNN-LSTM network is designed to handle time series inputs within a $21 \times N$ window, aiming to estimate angular positions with an output dimension of $3 \times T$. Initially, the input is processed through a three-layer CNN, resulting in a feature map. This feature map is passed to an energy neural network function to compute the attention scores. The softmax function is applied to these scores to derive the attention weights. These weights are then multiplied by the feature map to generate the context vector. Finally, this context vector serves as the input to the LSTM decoder.

previously generated, based on the teacher-forcing ratio. The output of the LSTM cells is calculated using Eqn. 3. The last LSTM cell's hidden state with a dimension of 1×3 is then passed to the controller as the desired position: $y_T = h_T$.

$$\mathbf{h}_i = f\left(\mathbf{h}_{i-1}, \mathbf{y}_{i'-1}, \hat{\mathbf{c}}_i\right) \tag{3}$$

The decoder's input undergoes a linear embedding to match the feature vectors' dimensions. The LSTM updates its hidden state $\mathbf{h}_i$ at each timestep i using a context vector $\hat{\mathbf{c}}_i$ (Eqn. 4) which is the weighted sum of the input features and encapsulates the most relevant information about the multi-dimensional input.

$$\hat{\mathbf{c}}_i = \sum_{j=1}^{K} \alpha_{ij} \mathbf{a}_j \tag{4}$$

The attention mechanism assigns weights α_{ij} to each feature vector based on its relevance for predicting the next embedding of y_i, with $\hat{\mathbf{c}}_i$ representing the weighted expected energy. The *softmax* function calculates these weights, ensuring they are normalized and non-negative, effectively turning the energy scores e_{ij} into probabilities in Eqn. 5.

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{K} \exp(e_{ik})} \tag{5}$$

The similarity between the inputs around position j and the output at position i is scored by the energy function e_{ij} in Eqn. 6, utilizing the rectified linear unit (ReLU) of a feedforward neural network, f_{att}.

$$e_{ij} = f_{\text{att}}\left(\mathbf{h}_{i-1}, \mathbf{a}_j\right) \tag{6}$$

While the inner-loop attention model constitutes a core component of our system, it's essential to recognize that maintaining static parameters for this model is not suitable for different users or across variations in movements. To accommodate a wide range of tasks and user-specific requirements, it's imperative that the model's parameters θ are dynamically updated. This necessity forms the basis for integrating meta-learning on top of the attention model.

B. Model-Agnostic Adaptation

EMG signals exhibit unique variations among individuals, due to the distinct muscle characteristics of each person. Despite these differences, a level of similarity exists within the distribution of tasks $p(\mathcal{T})$ across users or within a single user across movement variations. This characteristic similarity renders meta-learning a suitable approach. The objective of employing few-shot MAML in our study was to enable the model to adapt to new tasks with minimal data from previously unseen tasks. Consequently, our focus extended beyond a regression problem, aiming instead for a broader generalization across diverse tasks.

The dataset used in this study was split into a meta-training set $\mathcal{D}^{tr}$ and a meta-testing set $\mathcal{D}^{ts}$. In the context of N-way K-shot learning demonstrated in Algorithm 1, both the meta train and test datasets were divided into two sets: The support set $D_i^s = \{(X_i, Y_i)\}_{i=1}^{K \times N}$ with N indicating the number of tasks, K the number of examples for each selected task, and X and Y the source and the target respectively. And the query set $D_j^q = \{(X_j, Y_j)\}_{j=1}^{M}$ with M being the remaining samples from the same set of selected tasks.

Algorithm 1 MAML: K-Shot Learning

Require: Distribution $\rho(\mathcal{T})$ over tasks
Require: ψ, β: learning rate hyperparameters
Require: Number of sampled tasks N, dataset $\mathcal{D}$
randomly initialize θ
while not done **do**
  Sample batch of tasks $\mathcal{T}_i$ from the $\rho(\mathcal{T})$
  for all $\mathcal{T}_i$ **do**
    Randomly select N tasks
    Take K examples of each task for the support set $\mathcal{D}^s$ and the rest for $\mathcal{D}^q$
    Evaluate $\mathcal{L}_i \leftarrow \mathcal{L}(\mathcal{D}^s, \theta)$ in Eqn. (8)
    Update parameters with gradient descent: $\phi_i \leftarrow \theta - \psi \nabla_\theta \mathcal{L}_i(\theta, \mathcal{D}^s)$
    Evaluate $\mathcal{L}_i \leftarrow \mathcal{L}(\mathcal{D}^q, \phi_i)$ in Eqn. (8)
  end for
  Update parameters with gradient descent $\theta \leftarrow \theta - \beta \nabla_{\phi_i} \sum \mathcal{T}_n \sim \mathcal{D}^q \mathcal{L}_i(\phi_i, \mathcal{D}^q)$
end while
return model θ

We represented the attention-based CNN-LSTM model by f_θ with the set of parameters θ. By taking one gradient step on

the support task $\mathcal{T}_i$, the model parameters were updated, where ψ was the learning rate for the inner-loop attention model.

$$\phi_i = \theta - \psi \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta) \tag{7}$$

After adapting the model parameters to ϕ_i using the support set, the model's performance was evaluated on the query set, and the meta-objective was defined by the loss on the query set. The loss function in this regression example was a mean-squared error (MSE). For regression tasks using MSE, the loss took the form below.

$$\mathcal{L}_{\mathcal{T}_i}(f_\phi) = \sum_{\mathbf{X}^{(j)}, \mathbf{Y}^{(j)} \sim \mathcal{T}_i} \left\| f_\phi\left(\mathbf{X}^{(j)}\right) - \mathbf{Y}^{(j)} \right\|_2^2 \tag{8}$$

During meta-learning, a single gradient step on each of the N tasks (N-way) (Eq.(7)) is followed by an outer-loop update that minimizes the summed loss on the corresponding query set $\mathcal{D}_i^{\mathrm{q}}$ with updated parameters in (Eqn. 7).

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_\phi) \tag{9}$$

The meta-objective is the expected query-set MSE after one inner-loop update, and the meta-training process continually adjusts θ to minimize that error across tasks.

In practice, we interleave this process with brief fine-tuning phases: after every 100 outer-loop epochs, the current meta-parameters θ are adapted on a small support set (one shot per new task) to emulate online personalization. Before the first inner-loop step all tasks share identical parameters, whereas after adaptation, each task obtains its own ϕ_i (using learning rate β). This intermittent schedule converged faster than a continuous inner-loop update while remaining fully compatible with the standard MAML objective.

The prediction of the desired position for each DoF was performed using the updated parameters of the attention model, while concurrently, the parameters of the meta-learner were refined. Simultaneously, this computed desired position was sent to the position controller for tracking.

C. Robot-assisted Control

For the trajectory tracking controller, we aimed to follow the predicted position, taking into consideration challenges such as delays and internal friction inherent to the pneumatic cable-driven upper extremity exoskeleton. Using a PD controller (Eqn. 10) with proportional K_P and derivative K_D gains, the robotic exoskeleton accurately followed the desired joint angles q_{di} at each time step. The controller was complemented by feedforward gravity compensation τ_{GCi} that leveraged the system's known kinematics for enhanced control performance. The tracking control law τ_{ri} was then sent to the pneumatic actuator of each joint i.

$$\tau_{ri} = K_P(q_{di} - q_i) - K_D\dot{q}_i + \tau_{GCi} \tag{10}$$

The proportional and derivative gains were selected with a pragmatic, human-in-the-loop procedure that is common in soft-robot literature [6], [10]. We first bounded the search range analytically from a linearized single-joint model (mass = 2.3 kg, tubing compliance 0.32 N mm^{-1}) and then performed small-step manual tuning while the exoskeleton was gravity-compensated and unloaded, increasing K_P until a slight overshoot appeared and adding K_D until the response became critically damped (¡ 5 % overshoot, ¡ 300 ms settling). The final gains ($K_P = 32Nmrad^{-1}$, $K_D = 4.5Nmsrad^{-1}$) produced stable motion across all eight subjects and were kept constant for every experiment, thereby isolating the effect of the meta-learned intention decoder.

III. Materials and Experiments

A. Hardware setup

For the exoskeleton setup, we utilized the Trigno™ Research+ system (Delsys Incorporated, MA, USA) which includes an embedded IMU in the EMG signal acquisition setup. This setup also leveraged the Trigno Software Development Kit (SDK), facilitating data transfer from the Trigno System to Python 3.10. The Trigno Base Station, connected to the PC, streamed data to Python via TCP/IP, with the EMG and IMU sampling rates set at 1925.125 Hz and 148.148 Hz, respectively.

The acquired data were transferred to the Python environment using the Pyserial library on an Arduino at a 115200 baud rate, ensuring a communication delay below 10 ms. Encoder data were down-sampled in order to be synchronized with the EMG signals. Data were processed within a 15 ms window using a buffer mechanism that continually updated the attention model input.

In this system, two Arduino boards interfaced with the exoskeleton: an Arduino Uno 3.0 for controlling pneumatic actuators and an Arduino Mega for reading encoder data at a high interrupt frequency of 50 kHz. Both boards were connected to the local PC via USB 2.0. The exoskeleton with 3D-printed parts was powered by fluidic muscles (DMSP-20-RM-CM, Festo Corp., Esslingen, Germany) and used electro-pneumatic transducers (EP211-X120-10V, Omega Engineering Inc., USA) for actuator pressure regulation. Position measurement employed quadrature optical encoders (HEDM-5500 B12, Broadcom Inc., US) at the shoulder and elbow joints.

B. Data collection

Eight able-bodied individuals (including 2 females), aged between 24 and 30 years and with varying heights (156-186 cm) and weights (48-92 kg), participated in the study. Participants were selected to represent a broad spectrum of body build. Prior to the experiment, all participants provided their consent after being briefed about the project's aims and instructions. All experiments were conducted in accordance with protocols approved by the University of Alberta Research Ethics Boards. Nine key muscles were targeted for EMG and IMU sensor placement [36], [37]: *i)* biceps brachii (EMG + IMU), *ii)* deltoideus medius (EMG + IMU), *iii)* deltoideus anterior (EMG + IMU), *iv)* brachialis, *v)* brachioradialis, *vi)* pectoralis major (clavicular head), *vii)* triceps brachii, *viii)* deltoideus posterior, *ix)* trapezius descendens. A visual

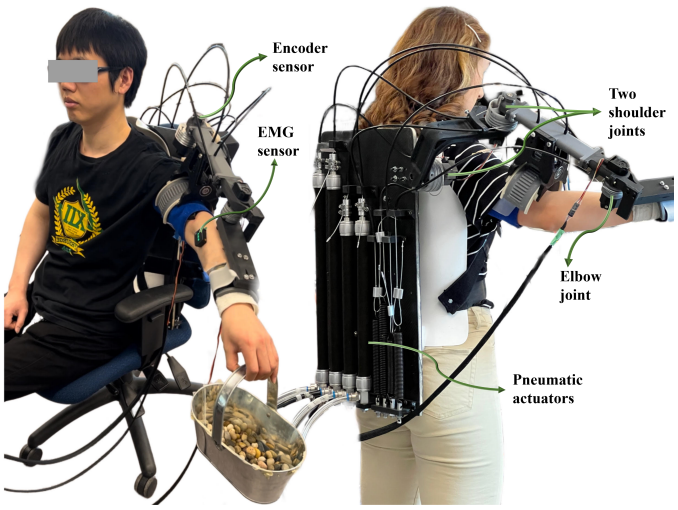


Fig. 3. The experimental configuration features a soft pneumatic cable-driven upper-limb exoskeleton, with nine EMG sensors positioned on the upper arms of two participants. On the left, a user is executing a movement while bearing a 1kg load in a seated posture, whereas the female participant on the right demonstrates a different movement, without any load, in a standing stance. During these activities, EMG, IMU, and angular position data are collected as the robot operates under gravity compensation.

representation of part of the data collection is shown in Fig. 3.

The first three sensors had an activated embedded IMU with an adjustable range, focusing on the gyroscope with three axes (+/- 250 dps). These primary sensors were linked to each main DoF. A brief calibration for two minutes was conducted to assess the sensor signal-to-noise ratio and the robot's operational state. The EMG and IMU signals were validated through the Delsys software (EMGworks®4.4).

The experiment involved 14 distinct movements while wearing the upper-limb exoskeleton. Half of these movements were conducted while carrying a 1 kg load, while the others were performed without any load. The movements ranged from activating a single joint to simultaneous activation of two or all three joint movements.

1) Elbow flexion without load
2) Shoulder flexion without load
3) Shoulder abduction without load
4) Elbow flexion + shoulder abduction without load
5) Elbow flexion + shoulder flexion without load
6) Shoulder abduction + shoulder flexion without load
7) All three joint movements without load
8) Elbow flexion with 1 kg load
9) Shoulder flexion with a 1 kg load
10) Shoulder abduction with a 1 kg load
11) Elbow flexion + shoulder abduction with 1 kg load
12) Elbow flexion + Shoulder flexion with 1 kg load
13) Shoulder abduction + shoulder flexion with 1 kg load
14) All three joint movements with 1 kg load

Participants had the autonomy to choose the pace of each movement and whether to sit or stand while performing the movements. For movements involving multiple joint activations, they could opt to rotate the joints simultaneously or in any preferred sequence. This flexibility in the experiment was intentional to collect more personalized data, aiming to tailor predictions to each user's specific ergonomic requirements.

Each participant repeated each movement five times, and the data collected included nine EMG signals, three gyroscope readings (each with three axes), and data from three angular encoders. Throughout the experiment, the exoskeleton was maintained in a gravity-compensated state. Each participant attended four sessions, each taking about 30 minutes on different days. Between sessions, the participant took breaks as needed. This would result in fatigue in any random movement that is performed last. Also, conducting the experiments on different days resulted in potential variations in EMG electrode placements ($displacement < 10mm$). The collected data were later used as input to the attention model.

By integrating variations across 14 movements and 8 users, we generated distinct *tasks* for meta-training purposes. This strategy yielded a total of $8 \times 14 = 112$ unique tasks. Given that each movement was performed 5 times, we obtained 5 examples (*shots*) per task, thereby limiting the combined size of the support and query sets to a maximum of 5 examples. The training was executed across various configurations of support and query set sizes. These examples were later used as input for the meta model.

C. Experimental Protocol

We assessed the performance and adaptability of the meta-attention model through a three-stage approach. Initially, we conducted a concise performance analysis, compared our model to other methodologies, and identified the most effective parameters for the few-shot learning. Subsequently, we evaluated the model's ability to adapt to four different scenarios explained later utilizing the configurations from the previous stage. Finally, we demonstrated the application of the adapted model in real-world scenarios by inferencing it with a robotic exoskeleton and conducting a series of experiments to validate its effectiveness and efficiency.

For the initial stage of our experiments, we utilized five examples per task, assigning between 1 and 4 examples to K for the support set and the remaining examples to the query set. Additionally, the model was trained using subsets of 5, 8, and 10 randomly chosen tasks to evaluate its performance across varying task numbers. Besides the internal attention mechanism, we also benchmarked our dataset against several other models. Specifically, we explored the efficacy of four deep learning architectures: Bi-LSTM, CNN-Bi-LSTM, CNN-LSTM, and an attention-based CNN-LSTM.

The Bi-LSTM model enhanced the capabilities of the uni-directional LSTM by simultaneously learning from both forward and backward contextual relationships in the input sEMG signals. Ma *et al.* [38] utilized a Bi-LSTM network for predicting upper limb joint angles, utilizing data from three EMG sensors. However, a combined framework of CNN and Bi-LSTM learned not only bi-directional temporal relationships but also spatial correlations. Karnam *et al.* [39] adopted this combined approach in their work on hand gesture recognition, employing EMG signal classification. The intention-based

predictive assistance (IBPA) model was a paralleled CNN-LSTM model before the integration of attention detailed in our previous work [22].

In the subsequent stage of the experiments, we assessed the model's adaptability and compared it with the standalone attention model across four distinct scenarios: *i)* encountering new repetitions of previously seen tasks, *ii)* experiencing new movements from previously observed users, *iii)* introducing new users to previously seen movements, and *iv)* observing new users executing new movements.

For the meta model we used a few-shot learning scheme where we conducted a fine- tuning step at every 100th epoch, and for the attention model, we evaluated the loss on the test set at every 100th epoch. During this phase, we reset the parameters of the meta and attention models to random initialization for each scenario. The meta model was trained 5 times with random initialization and random dataset splits for each scenario to ensure consistency across runs.

In the first scenario, we divided the training and testing datasets based on the five repetitions (examples), resulting in a dataset split ratio of 4:1 for train and test. In the second scenario, we split the dataset based on the type of movements, allocating the first half of movements (performed without a load) to the training set and the next half of movements (performed with a load) to the testing set. Additionally, we conducted an alternative version of the second scenario where tasks were randomly selected with a train-test ratio of 11:3 and repeated the training 5 times.

For the third scenario, we isolated the data of two users for testing purposes which led to a split of 6:2 ratio. In the final scenario, we concurrently divided the dataset based on users and movements. This entailed reserving the data from two random users and exclusively using movements performed with a load for the test set, while the remaining six users' movements without a load were allocated to the training set.

In the last stage of the experiments, we initially trained the meta model using data from six randomly selected participants, incorporating all 14 movements, both with and without the 1kg load. Subsequently, this trained model was employed to initialize the meta model for inference purposes in contrast with the previous stage where we restarted with random initializations. The two remaining participants were tasked with executing two new tasks (shown in solid black line in Fig. 6) while new, continuous data were gathered. Both participants started with the same pre-trained model. For each task, participants were guided to follow a displayed trajectory, along with the real-time position of the robot's end effector demonstrated on a computer screen. They were required to maintain a position within a 2cm radius of the indicated path, with deviations beyond this boundary considered as task failures.

The participants each repeated the first task 5 times. The first repetition of the first task was used in the support set for the fine-tuning step of the meta-learner. Two gradient steps were executed to refine and optimize the attention model's weights for these new tasks. Each repetition/shot of the first task, on average, took 21 seconds to complete for each participant. We ensured that the new task would remain in the support data set of the fine-tuning step after the 5 repetitions. The second task involved lifting a 2kg load to place it on a shoulder-level surface. Both tasks were executed with a speed of 0.13 m/s.

Overall, each stage of the experiment was directly aligned with our core objectives of demonstrating adaptability and user-specific customization of the myoelectric control system. By integrating a diverse range of movements and user interactions, we validated the system's real-world applicability and performance.

D. Meta-Training

A GPU with 32 GB of RAM was used for training the model, spanning over 4000 epochs for each stage of the experiment. The training duration ranged from 8 to 15 hours for each phase of the experiments. An optimal learning rate of 0.01 was established for the meta-learner, whereas a rate of 3×10^{-4} proved most effective for the inner loop optimizations. Experimentation with the number of gradient steps, ranging from 1 to 5, revealed that 2 steps provided the best balance between computational efficiency and model performance, coupled with a manageable batch size of 12. The model employed the Adam optimizer for both meta-learning and fine-tuning phases for its proven efficacy in similar deep learning tasks. Additionally, at every 100-epoch interval, the model entered a fine-tuning stage, where adjustments were made to enhance its specificity and performance on new tasks.

IV. Results

A. Performance Analysis

As illustrated in Table I, we evaluated the performance of four types of deep learning models in training the meta-learner, varying both the number of tasks selected (ways) and the number of examples per task (1 to 4) for the support set. The loss for each model was computed using Equation 8, normalized by the length of the evaluation set's dataloader. Instances marked as N/A in the table indicate attempts to train the model with those specific configurations; however, due to computational constraints and limited GPU resources, training could not be completed successfully.

In conducting a one-way ANOVA test to evaluate the impact of increasing the number of examples per task on model loss, we observed a statistically significant effect ($F(3, 44) = 10.95, p = 0.042$), the post hoc test indicates that models trained with a greater number of examples exhibited lower loss values. This result confirmed our hypothesis that additional data per task enhances model accuracy by providing a more comprehensive basis for fine-tuning.

Conversely, a post hoc test on the number of ways revealed a significant decrease in accuracy with an increase in the number of tasks ($F(2, 45) = 20.85, p = 0.028$). This suggests that as the model is required to generalize across a broader set of tasks, its ability to accurately predict outcomes diminishes.

Conducting a one-way ANOVA test to evaluate the impact of four types of deep learning models on the loss of the system revealed a statistically significant difference ($F(3, 44) = 36.13,\ p < 0.001$). The analysis was followed by a post hoc Tukey HSD test which confirmed our hypothesis that the

TABLE I
THE FINAL EVALUATION LOSS OF THE META-LEARNING MODEL APPLIED ON FOUR INTERNAL DEEP LEARNING MODELS WITH 5, 8, AND 10-WAY REGRESSION ACROSS THE INTRODUCTION OF 1 TO 4 TASK EXAMPLES.

	5-way				8-way				10-way			
Internal models	**1-shot**	**2-shot**	**3-shot**	**4-shot**	**1-shot**	**2-shot**	**3-shot**	**4-shot**	**1-shot**	**2-shot**	**3-shot**	**4-shot**
Bi-LSTM [38]	2.59	2.07	1.87	1.33	2.48	2.35	2.39	2.02	2.57	2.53	2.47	1.80
CNN Bi-LSTM [39]	1.94	1.80	1.68	1.54	2.03	1.93	1.72	1.55	2.54	2.19	1.98	N/A
IBPA (parallel CNN-LSTM) [22]	1.88	1.78	1.79	1.63	2.03	1.84	1.59	N/A	2.56	2.24	N/A	N/A
Attention-based CNN-LSTM	0.79	0.65	0.62	0.58	1.10	0.98	0.84	0.64	1.52	1.34	1.08	N/A

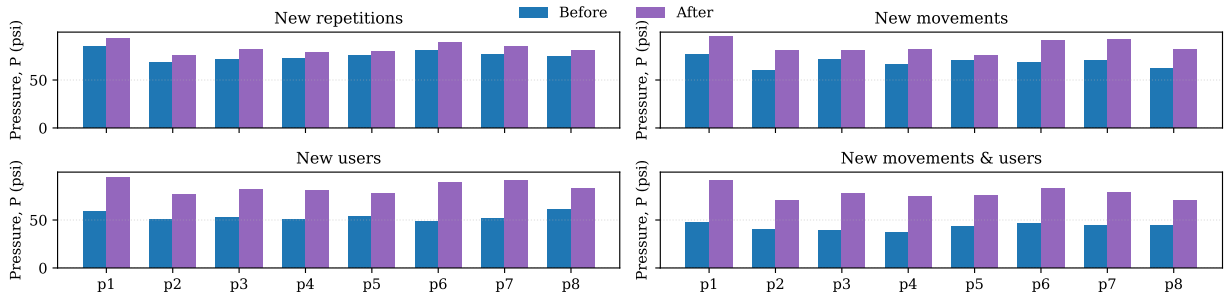


Fig. 4. The sum of all joint torques converted to psi, root-mean-squared over all actuators was averaged across all trainings. After a single fine-tuning session with the meta-learned controller, all four training scenarios exhibited a systematic increase in exoskeleton torque accompanied by a concomitant drop in EMG amplitude (not shown). In each subplot, blue bars depict performance of the standalone attention model before adaptation, whereas purple bars show the after-adaptation performance achieved with MAML.

attention model has a significantly lower loss compared with other counterparts ($p < 0.001$) with a 0.98 mean difference with the IBPA.

B. Intersubject Effort Evaluation

Despite the large biomechanical diversity of the eight volunteers, baseline torque production differed markedly across participants (e.g., new reps = 68.3 – 85.0 PSI; new moves + users = 37.1 – 47.3 PSI). A two-tailed paired t-test ($\alpha = 0.05$, Bonferroni-corrected) confirmed that torque gains were significant in every condition, with very-large within-subject effect sizes (Cohen's $d \geq 2.7$). Complementary analyses on rectified-and-RMS-filtered EMG showed an average $38 \pm 9\%$ reduction across the same four conditions (all $p < 0.005$), indicating that participants achieved the higher torques with substantially less muscular effort (see Fig. 4).

C. Task Adaptation

The adaptation process involves dynamically updating the model's parameters based on the specificity of the new tasks by employing a fine-tuning approach. To evaluate the adaptability of the meta model, we subjected it to four distinct scenarios, introducing new, unseen data in each. In the first scenario, where the model was trained on repeated instances of the same tasks, the plots on the far left column of Fig. 5 revealed that both the attention and MAML models adapted after 2.3k and 1.2k training iterations, respectively. Notably, the meta-learner achieved a lower loss after 4k training iterations ($\text{loss} = 0.51, \text{SD} = 0.09$) compared to the attention model ($\text{loss} = 1.3, \text{SD} = 0.16$).

For the second and third scenarios, the model underwent five training sessions on randomly partitioned dataset splits, as detailed in III-C. Despite the attention model's loss converging to zero during training, it underperformed in testing phases, especially when it was introduced to new users ($\text{attention loss} = 1.75, \text{SD} = 0.14$) & ($\text{Meta loss} = 0.7, \text{SD} = 0.1$). Within our application framework, a loss exceeding 2 is considered a failure.

In the final scenario, testing the model on new users executing new movements with a training dataset of 42×5 shots and testing dataset of 14×5 shots, the standalone attention model was unsuccessful ($\text{loss} = 2.5, \text{SD} = 0.25$) using a combined training data of 84 minutes. Conversely, employing the meta-learner enabled the model to adjust rapidly by fine-tuning on 1 shot of new data ($\text{loss} = 0.84, \text{SD} = 0.19$) accumulating to 16 minutes of data after 40 iterations, showcasing the meta-learner's capability to rapidly adapt to entirely new conditions.

Finally, in evaluating the effect of incorporating meta-learning into the attention model on the loss in all cases, a paired samples t-test was performed on the loss values before and after applying the meta-learning. The analysis revealed a statistically significant decrease in loss with the meta-learning application, with mean loss decreasing from ($M_{before} = 1.82$, $SD_{before} = 0.16$) to ($M_{after} = 0.86$, $SD_{after} = 0.09$), $t(23) = 5.62$, $p < 0.001$. This substantial reduction signifies the effectiveness of meta-learning in refining the attention model's predictive accuracy.

D. Online learning

In our final phase of experiments, designed to assess the model's real-time performance, we collected a new dataset while participants wore the exoskeleton. The initial task, depicted in Fig. 6, involved no load and was intended to capture muscle activity data from new users to update the attention

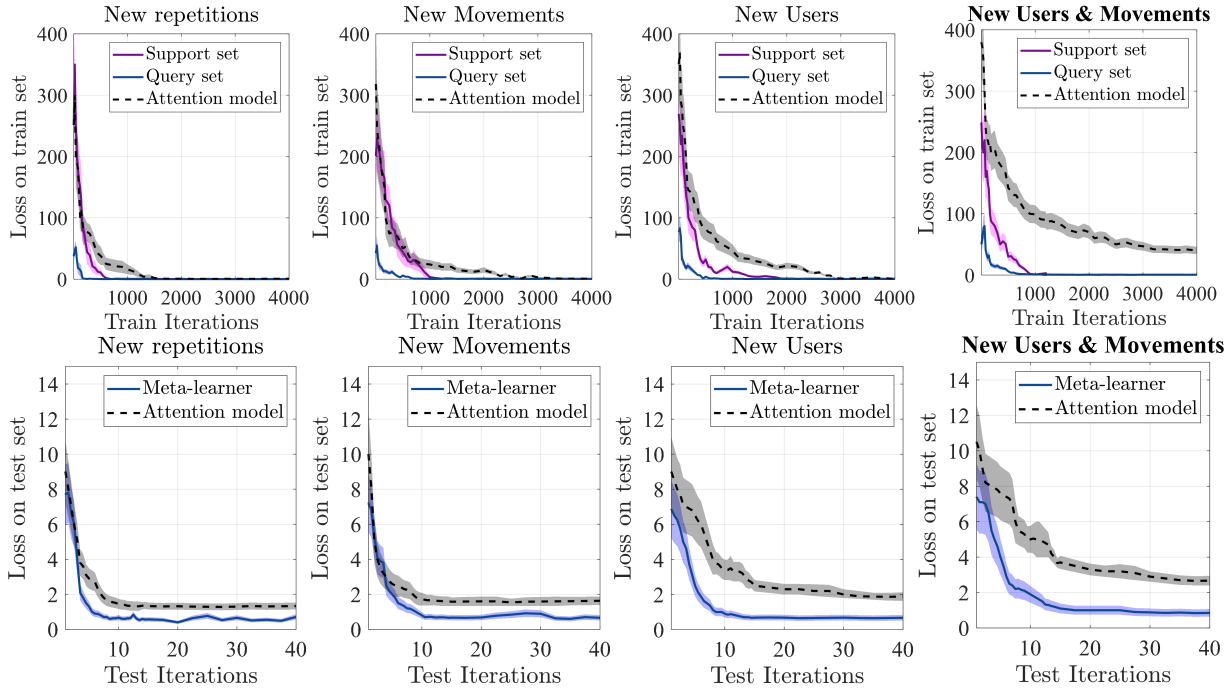


Fig. 5. Comparative Training Outcomes Across Four Distinct Scenarios: MAML vs. Attention Model (dashed line in black). The upper row displays training iterations conducted on the training set, while the lower row illustrates performance on the test set. The MSE loss is averaged over 5 trainings, which were randomly initialized and demonstrated with shading.

model's parameters. This was followed by 5 repetitions of the second task, aimed at evaluating the model's gradual learning ability. By updating the model after the first task, and including it in the support set of the fine-tuning step, we ensured that the second task would be learned more rapidly. Remarkably, the model demonstrated rapid adaptation to the second task with an average duration of 26 seconds per repetition per user.

On average, a notable reduction of 13% in user effort from task 1 to task 2 was observed, as measured by root mean square (RMS) EMG values. Additionally, a 24% increase in the exoskeleton's torque was noted, indicating enhanced assistance during the performance of the second task. Furthermore, participants were able to follow the reference trajectory more accurately, achieving an average MSE of 0.91.

V. Discussion

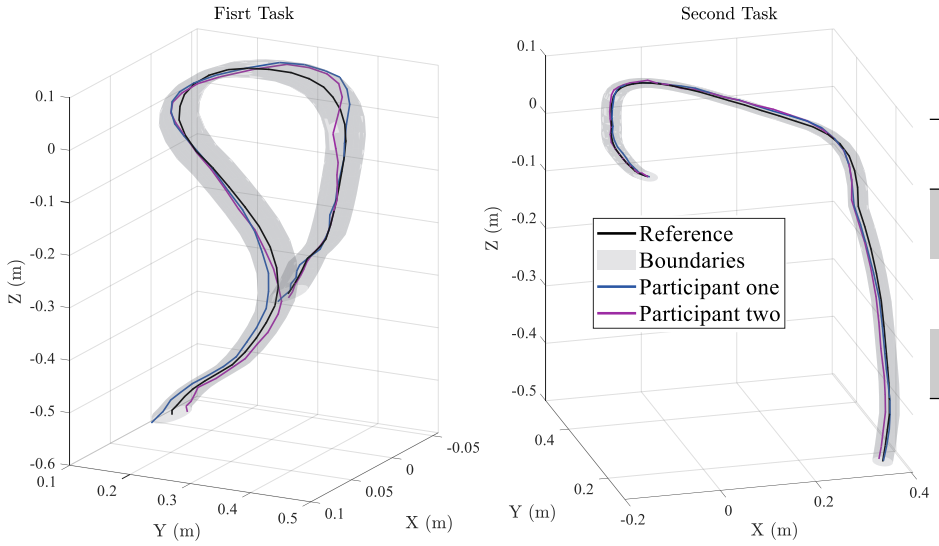
We previously conducted a comprehensive ablation study, which established that attention mechanisms outperform other models in terms of effectiveness [27]. Additionally, we highlighted the flexibility of attention mechanisms to accommodate variations in input length, a critical feature for handling time-series data using a CNN-LSTM network [22]. Our findings further confirmed that attention mechanisms contribute positively to modularity; that is, they maintain satisfactory performance even when one or two sensors are compromised or disabled. Given that the meta-learning approach is model-agnostic, it can be applied to various internal models beyond the CNN-LSTM configuration. This versatility significantly enhances the adaptability of our framework, making it suitable for a broader range of applications such as rehabilitation [40].

In this study, we leveraged the MAML framework via few-shot learning to customize the attention model for individual users. Our task adaptability experiments demonstrated the system's effective performance, even under user fatigue, and its robustness against sensor placement variations and EMG signal noise.

Our findings indicate that an increase in the number of examples per task during training, -equating to approximately 20 seconds of data on average and an additional 5 seconds of training time— enhanced model accuracy. This improvement is attributed to the expansion of the support set, which provides a more substantial data basis for fine-tuning the model. Conversely, increasing the number of ways (number of task subsets) in the K-shot learning scheme tends to diminish accuracy across all four deep learning models. This suggests a dilution of focus, as the model must generalize across a broader set of tasks.

While expanding the support set enhanced model accuracy, it introduced a trade-off in terms of training duration. Increasing the number of examples per task (shots) increased the training time from 25.4 seconds to 124.4 seconds, on average. Our analysis revealed that the discrepancy in loss between utilizing one shot and four shots for the attention model was minimal, prompting our decision to proceed with a one-shot approach henceforth. This would also reduced the computational complexity.

Nonetheless, among the evaluated models, the attention-based CNN-LSTM model exhibited the lowest loss, validating our hypothesis that an attention mechanism enhances predictive capability. This model's superior performance underscores the effectiveness of attention mechanisms in capturing relevant



Task	1	2
MSE	1.53	0.91±0.11
EMG	1.20	1.04±0.29
Torque	58	72 ± 6

Fig. 6. Two predefined trajectories, depicted as solid black lines, were executed by two participants. The initial task served to update the meta model, followed by a test involving the second task, during which participants carried a 2 kg load to a shoulder-level surface. Adjacent to the plots, a table compares the outcomes in terms of Root Mean Square (RMS) EMG and RMS torque (psi), as well as the average MSE of the trajectory traversed by the participants over 10 repetitions relative to the reference trajectory.

features from the data, thereby improving the accuracy of predictions. In our comparative analysis, attention mechanisms demonstrated a 32% higher accuracy in task performance (Table I), outperforming traditional models by a significant margin.

During the task adaptation experiments, we compared the adaptability of the attention model before and after the integration of the meta-learning framework. For the attention model to be effective on new unseen movements or users, a combined data of 200 minutes was required along with additional training iterations. When introduced to new users performing new movements, the attention model failed even with 200 minutes of combined data. Conversely, with the incorporation of the meta-learning approach, the system demonstrated a remarkable ability to adapt to movements and users utilizing less than 168 minutes of pre-trained data complemented by only 8 minutes of newly acquired data. In the most demanding scenario where the attention model failed, alongside 84 minutes of the pre-trained dataset, a supplementary 16 minutes of new data proved enough for the model to achieve satisfactory performance levels. This showed an average reduction of 50% in training data.

The intersubject findings support three key claims: (i) inter-subject variability—driven by anthropometry and sensor placement—creates a wide spread in baseline effort; (ii) meta-adaptation reliably lowers user effort while boosting mechanical output, independent of the new task or user identity; (iii) when motion variability rises (new users + new movements), a standalone attention-based CNN-LSTM fails to maintain intention-decoding accuracy (29 % drop versus baseline), whereas the proposed MAML-initialised network preserves performance through rapid on-line fine-tuning.

In the third part of the experiments, our findings highlighted the model's ability to learn new tasks with just a single example during real-time data collection. The repeatability of this experiment was further validated by having another participant repeat the same tasks. The first encounter of the model with the "new user performing a new movement" was the most strict condition in our experiment. However, not only did we observe a satisfactory performance during this task, but also it was observed that updating the parameters using only one shot of this new task (21 seconds of data and 4 seconds of training), resulted in a better performance during the second task with a loss reduction of 40%. This also led to a notable reduction in human effort during load-bearing tasks compared to non-load-bearing tasks. In return, the exoskeleton's torque output increased, all while achieving higher precision, as demonstrated in Figure 6. This demonstrated the model's escalating adaptability while providing assistance, improving progressively with each introduction of minimal data.

Given that the model had previously been trained with data from six individuals, it had already acquired a robust ability to adjust to new tasks. This foundational learning ensures that even when presented with data from a new user during the first task, the model achieved satisfactory results with just a single instance (21 seconds). This outcome highlights its distinct advantage over the conventional methods. It's important to note that by placing the first new task in the support set, we effectively treated the second task as a "new movement" scenario with updated parameters from the first task. After five repetitions of the second task, the model's performance improved, benefiting from the additional data exposures.

Beyond addressing cross-subject variations, the model rapidly adjusted to new loads, enhancing user comfort during subsequent tasks despite increased weight and longer trajectories. Also, we demonstrated that the model was robust to sensor placement variations introduced in the dataset.

Looking ahead, we plan to introduce multiple new tasks and move beyond simply using a pre-trained model for initialization. Instead, we aim to incorporate tasks from new users

directly into the training dataset for subsequent participants. This approach will allow us to demonstrate incremental learning across several stages, reinforcing our theory of progressive learning capability. Additionally, this method will enable us to conduct comparisons between users. We could also perform more repetitions of the first task to further see the gradual improvement of the model.

Although online PD or impedance control [13], [14] can improve adaptability, our system's constant PD gains were sufficient due to the pneumatic exoskeleton's inherent low bandwidth and slow dynamics, coupled with effective gravity compensation that minimized unmodeled disturbances. In future work, we plan to integrate an adaptive impedance layer to fine-tune PD gains in real-time, bridging toward the dynamic control strategies shown.

We plan to undertake a comprehensive statistical analysis using methods such as paired t-tests and regression models to quantitatively assess the improvements brought about by our model in various operational scenarios. Future work will also explore the model's capabilities during online operations by expanding the variety of new tasks and more diverse user scenarios such as those with disabilities during inference. Additionally, we aim to further refine the efficiency of our system, reducing adaptation times and exploring the potential for incorporating additional sensory inputs to enrich user interaction with the exoskeleton.

VI. Conclusion

In this study, we represented a meta-learning framework that is model agnostic for addressing the critical need for adaptable and personalized myoelectric control systems. We utilized an attention-based CNN-LSTM model that could decode muscle activity and map it into future positions. Our goal was to detect user intention when interacting with an upper-limb exoskeleton to better assist the user with shared control and load-bearing tasks to prevent injuries in industrial settings. In our previous study, we showed the benefits of using attention in our CNN-LSTM model. However, the most predominant limitation in our work was the lack of adaptability and personalization when introduced to new movements or new users unseen by the model. The main contribution of this work was that by using the MAML approach, we were able to predict future angular positions with only a small amount of new data. In addition, we mitigated the need for new training data for new users or new environments. We also showed that our model can adapt to a new task in real-time.

References

[1] A. Golabchi *et al.*, "A framework for evaluation and adoption of industrial exoskeletons," *Applied Ergonomics*, vol. 113, p. 104103, 2023.

[2] X. Li *et al.*, "A passive upper limb assistive exoskeleton for overhead assembly tasks," in *2022 IEEE 17th Conference on Industrial Electronics and Applications (ICIEA)*, 2022, pp. 319–324.

[3] J. S. Khan *et al.*, "A review on the design of assistive cable-driven upper-limb exoskeletons and their experimental evaluation," in *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2022, pp. 59–64.

[4] L. Grazi *et al.*, "Design and experimental evaluation of a semi-passive upper-limb exoskeleton for workers with motorized tuning of assistance," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 10, pp. 2276–2285, 2020.

[5] Y. Guo *et al.*, "Task performance-based adaptive velocity assist-as-needed control for an upper limb exoskeleton," *Biomedical Signal Processing and Control*, vol. 73, p. 103474, 2022.

[6] M. Lee *et al.*, "Self-healing soft bio-electronics enable long-term upper-limb exosuits," *npj Flexible Electronics*, vol. 8, no. 27, pp. 1–12, 2024.

[7] J. J. Huaroto *et al.*, "A soft pneumatic actuator as a haptic wearable device for upper limb amputees: Toward a soft robotic liner," *IEEE Robotics and Automation Letters*, vol. 4, no. 1, pp. 17–24, 2019.

[8] N. Li *et al.*, "Multi-sensor fusion-based mirror adaptive assist-as-needed control strategy of a soft exoskeleton for upper limb rehabilitation," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 1, pp. 475–487, 2024.

[9] Y. Wang *et al.*, "Extracting human-exoskeleton interaction torque for cable-driven upper-limb exoskeleton equipped with torque sensors," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, pp. 4269–4280, 2022.

[10] E. Trigili *et al.*, "Detection of movement onset using emg signals for upper-limb exoskeletons in reaching tasks," *Journal of NeuroEngineering and Rehabilitation*, vol. 16, 2019.

[11] W. Ye *et al.*, "Motion detection enhanced control of an upper limb exoskeleton robot for rehabilitation training," *Int. J. Humanoid Robotics*, vol. 14, 2017.

[12] Y.-X. Liu *et al.*, "A muscle synergy-inspired method of detecting human movement intentions based on wearable sensor fusion," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 1089–1098, 2021.

[13] J. Xu *et al.*, "Mirror adaptive impedance control of multi-mode soft exoskeleton with reinforcement learning," *IEEE Transactions on Automation Science and Engineering*, vol. PP, pp. 1–13, 01 2024.

[14] X. Xiong *et al.*, "Learning-based multifunctional elbow exoskeleton control," *IEEE Transactions on Industrial Electronics*, vol. PP, pp. 1–1, 10 2021.

[15] J. Fu *et al.*, "Myoelectric Control Systems for Upper Limb Wearable Robotic Exoskeletons and Exosuits—A Systematic Review," *Sensors (Basel, Switzerland)*, vol. 22, no. 21, p. 8134, 2022.

[16] Z. Tang *et al.*, "An upper-limb power-assist exoskeleton using proportional myoelectric control," *Sensors*, vol. 14, no. 4, pp. 6677–6694, 2014.

[17] G. Hajian *et al.*, "Generalizing upper limb force modeling with transfer learning: A multimodal approach using emg and imu for new users and conditions," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 32, pp. 391–400, 2024.

[18] E. Bardi *et al.*, "Upper limb soft robotic wearable devices: a systematic review," *Journal of NeuroEngineering and Rehabilitation*, vol. 19, no. 1, p. 87, Aug 2022.

[19] L. Wang *et al.*, "Hand gesture recognition using smooth wavelet packet transformation and hybrid cnn based on surface emg and accelerometer signal," *Biomedical Signal Processing and Control*, vol. 86, p. 105141, 2023.

[20] H. Ashraf *et al.*, "Optimizing the performance of convolutional neural network for enhanced gesture recognition using semg," *Scientific Reports*, vol. 14, no. 1, p. 2020, 2024.

[21] D. Tam *et al.*, "Deep-metric meta-learning for subject-independent emg gesture recognition," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 32, pp. 777–788, 2024.

[22] P. Sedighi *et al.*, "Emg-based intention detection using deep learning for shared control in upper-limb assistive exoskeletons," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 41–48, 2024.

[23] T. Bao *et al.*, "A cnn-lstm hybrid model for wrist kinematics estimation using surface electromyography," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021.

[24] A. Vaswani *et al.*, "Attention Is All You Need," *arXiv*, 2017.

[25] K. Xu *et al.*, "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention," *arXiv*, 2015.

[26] A. Zhang *et al.*, "Upper limb movement decoding scheme based on surface electromyography using attention-based kalman filter scheme," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 31, pp. 1878–1887, 2023.

[27] P. Sedighi *et al.*, "A hybrid cnn-lstm network with attention mechanism for myoelectric control in upper limb exoskeletons," in *2024 21st International Conference on Ubiquitous Robots (UR)*, 2024, pp. 238–244.

[28] E. Rahimian *et al.*, "FS-HGR: Few-Shot Learning for Hand Gesture Recognition via Electromyography," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 1004–1015, 2021.

[29] C. Zhu *et al.*, "An Attention-Based CNN-LSTM Model with Limb Synergy for Joint Angles Prediction*," *2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, vol. 00, pp. 747–752, 2021.
[30] X. Fan *et al.*, "CSAC-Net: Fast Adaptive sEMG Recognition through Attention Convolution Network and Model-Agnostic Meta-Learning," *Sensors*, vol. 22, no. 10, p. 3661, 2022.
[31] C. Finn *et al.*, "Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks," *arXiv*, 2017.
[32] M. Shabanpour *et al.*, "Moemba: Memory-based meta-learning for continual robot motor adaptation," *IEEE Robotics and Automation Letters*, vol. 10, no. 2, pp. 1234–1241, 2025.
[33] K. Wang *et al.*, "Iterative self-training based domain adaptation for cross-user semg gesture recognition," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 31, pp. 2974–2987, 2023.
[34] M. Oghogho *et al.*, "Deep reinforcement learning for emg-based control of assistance level in upper-limb exoskeletons," in *2022 International Symposium on Medical Robotics (ISMR)*, 2022, pp. 1–7.
[35] N. Li *et al.*, "Model-Agnostic Personalized Knowledge Adaptation for Soft Exoskeleton Robot," *IEEE Transactions on Medical Robotics and Bionics*, vol. 5, no. 2, pp. 353–362, 2023.
[36] Y. A. C. Campos *et al.*, "Different shoulder exercises affect the activation of deltoid portions in resistance-trained individuals," *Journal of Human Kinetics*, vol. 75, pp. 5–14, Oct. 2020.
[37] C. G *et al.*, "An electromyographic analysis of lateral raise variations and frontal raise in competitive bodybuilders," *Int J Environ Res Public Health*, vol. 17, Aug. 2020.
[38] C. Ma *et al.*, "A bi-directional lstm network for estimating continuous upper limb movement from surface electromyography," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7217–7224, 2021.
[39] N. K. Karnam *et al.*, "Emghandnet: A hybrid cnn and bi-lstm architecture for hand activity classification using surface emg signals," *Biocybernetics and Biomedical Engineering*, vol. 42, no. 1, pp. 325–340, 2022.
[40] H. P. J. Dutta *et al.*, "Efficient hand segmentation for rehabilitation tasks using a convolution neural network with attention," *Expert Systems with Applications*, vol. 234, p. 121046, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417423015488