

Decomposing Task-Relevant Information From Surface Electromyogram for User-Generic Dexterous Finger Force Decoding

Jiahao Fan^{1b} and Xiaogang Hu^{1b}

Abstract—Existing electromyographic (EMG) based motor intent detection algorithms are typically user-specific, and a generic model that can quickly adapt to new users is highly desirable. However, establishing such a model remains a challenge due to high inter-person variability and external interference with EMG signals. In this study, we present a feature disentanglement approach, implemented by an autoencoder-like architecture, designed to decompose user-invariant, motor-task-sensitive high-level representations from user-sensitive, task-irrelevant representations in EMG amplitude features. Our method is user-generic and can be applied to unseen users for continuous multi-finger force predictions. We evaluated our approach on eight subjects, predicting the force of three fingers (index, middle, and ring-pinky) concurrently. We assessed the decoder's performance through a rigorous leave-one-subject-out validation. Our developed approach consistently outperformed both the conventional EMG amplitude method and a commonly used feature projection approach, principal component analysis (PCA), with a lower force prediction error (RMSE: $6.91 \pm 0.45\%$ MVC; R^2 : 0.835 ± 0.026) and a higher finger classification accuracy ($83.0 \pm 4.5\%$). The comparison with the state-of-the-art neural networks further demonstrated the superior performance of our method in user-generic force predictions. Overall, our methods provide novel insights into the development of user-generic and accurate neural decoding for myoelectric control of assistive robotic hands.

Index Terms—Surface electromyogram, feature decomposition, finger force prediction, neural interface.

Manuscript received 3 November 2023; revised 16 February 2024; accepted 25 March 2024. Date of publication 3 April 2024; date of current version 3 July 2024. This work was supported in part by the National Science Foundation under Grant CBET-2246162, Grant IIS-2330862, and Grant IIS-2319139, and in part by the Department of Defense under Grant W81XWH2110185. (Corresponding author: Xiaogang Hu.)

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Institutional Review Board of the Pennsylvania State University in 2022 under IRB approval No. STUDY00021035, and performed in line with the Declaration of Helsinki.

Jiahao Fan is with the Department of Mechanical Engineering, Pennsylvania State University, University Park, PA 16802 USA.

Xiaogang Hu is with the Departments of Mechanical Engineering, Kinesiology, Physical Medicine and Rehabilitation, Huck Institutes of Life Sciences, and the Center for Neural Engineering, Pennsylvania State University, University Park, PA 16802 USA (e-mail: xxh120@psu.edu).

This article has supplementary downloadable material available at <https://doi.org/10.1109/JBHI.2024.3383598>, provided by the authors.

Digital Object Identifier 10.1109/JBHI.2024.3383598

I. INTRODUCTION

THE human digits are capable of performing precise and coordinated movements with little conscious effort, empowering us to execute a variety of daily tasks ranging from simple grasping to complex object manipulations. However, neuromuscular injuries can severely impair hand functions, which imposes challenges to the quality of life of these individuals. In recent years, assistive devices like prosthetic hands and exoskeletons, which were designed to mimic the intricate movements of a biological human hand, have progressed significantly [1], [2], [3], [4], [5]. However, naturally controlling these advanced robotic devices during real-life activities has been a long-standing challenge [6], [7]. One primary reason is the absence of a robust neural-machine interface capable of accurately decoding the user's motion intentions at individual digit levels [8], [9].

Surface electromyogram (EMG) has emerged as a source for decoding human motion intentions. For upper limb control, extensive studies have explored EMG-based pattern recognition, demonstrating its ability to identify a vast array of hand/wrist gestures [10], [11], [12]. However, pattern recognition operates on a discrete finite state machine, which does not fully capture the nuances of human hand continuous control of movements. In contrast, an alternative approach, termed proportional direct control [13], [14], allows for continuous control of individual fingers and is highly preferable for achieving dexterous control of advanced robotic hands. Although promising, such an approach poses additional challenges, because it requires an accurate decoding of not only the intended motion of individual fingers but also the continuous variations of finger joint kinematics or fingertip forces.

Generally, continuous control of robotic fingers could be achieved through various regression models that maps between EMG amplitude and motor output. However, most previous studies focus on the estimation of one finger force at a time, which has limited applicability, since dexterous multi-finger motions are generally involved in most daily manipulation tasks. It remains a substantial challenge to concurrently decode the motor output of multiple fingers in a continuous manner, largely due to the anatomical organization of finger muscles that are closely located to each other. Recently, high-density EMG (HD-EMG) has been utilized in proportional EMG-force modeling, revealing distinct localized activation patterns on EMG feature maps

during different finger motions, which enables the decoding of the forces of separate fingers in a concurrent manner [15], [16], [17]. With HD-EMG, previous studies managed to establish the finger-specific mappings between the HD-EMG energy map and finger force, demonstrating the possibility of finger-specific control. However, the accuracy is still far from satisfactory as the intricate interplay across multiple fingers could not be fully captured by linear models. In contrast, deep neural networks, with high model complexity, have shown some potential with high decoding accuracy [18], [19]. A primary limitation, though, is that most existing models are user-specific and require extensive training data for each user, which imposes a cumbersome burden for data acquisition. From a practical perspective, a generic model that can efficiently adapt to new users without extensive calibration procedures, while maintaining a high level of accuracy, is highly desirable.

A primary challenge in developing a user-generic model is the user-specific peculiarities embedded in global HD-EMG features. For example, the HD-EMG energy map during specific finger motion can differ between people, which is attributed to person-specific variations in anatomical structures, neuromuscular control strategies, or muscle co-activation patterns. Such variations are often considered task-irrelevant and may lead to notable interference during modeling. Despite these individual differences, the dominant patterns in HD-EMG during the same finger motion should still exhibit consistent information across different individuals, given the shared fundamental physiological and mechanical constructs [20]. For example, a significant overlap can be observed in the HD-EMG energy maps associated with the same finger motion across different people.

In light of these challenges, rather than attempting to directly establish a generic mapping between EMG and force, we seek to develop a feature projection technique that learns the representation of original features in a new latent space that can disentangle the user-invariant and task-sensitive representations from user-sensitive and task-irrelevant representations. Specifically, the latent space can maximize the most prominent information relevant to finger motor output while the user-sensitive information is minimized. We hypothesize that a user-generic model would benefit from mitigating the influence of the user-sensitive (task-irrelevant) components while retaining the task-sensitive components that encapsulate the information of involved fingers and their corresponding forces.

Accordingly, we developed a novel framework designed to explicitly extract the task-sensitive high-level representations from the original HD-EMG features. Our implementation leveraged an autoencoder (AE) with specifically tailored loss functions. The AE consists of two branches that can separately capture task-sensitive information and user-sensitive information, respectively. By extracting the task-sensitive information across users, we were able to apply the trained model directly to new users for both finger force estimation and finger classification. We evaluated the method, termed Disentangle Autoencoder (DisAE), on eight participants for concurrent prediction of finger forces during multiple-finger (index, middle, and ring-pinky finger) force production tasks. Our results showed that this novel approach can achieve a significantly better prediction

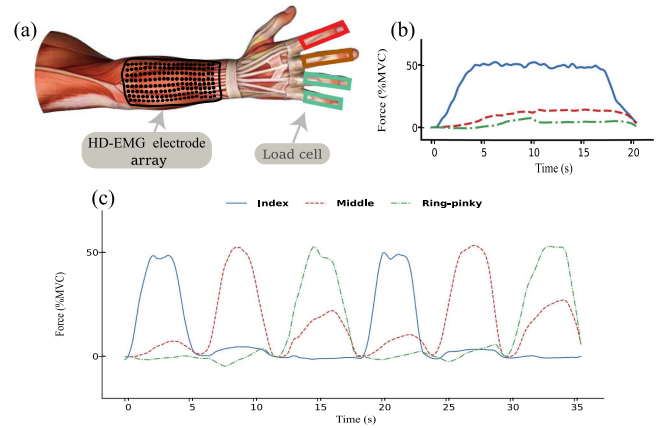


Fig. 1. Experimental setting. (a) An 8×20 EMG electrode array was placed on the extensor digitorum communis of the forearm (b) An example of the single-finger trial (c) An example of the multi-finger trial.

performance over the conventional EMG amplitude method and the state-of-the-art feature projection approach. The comparisons with the state-of-the-art neural networks further demonstrated the superior performance of the DisAE in cross-user force prediction. Overall, the current study has several novel contributions. First, we introduced a novel feature projection approach for HD-EMG, yielding more compact and informative representations that enhanced performance in both finger classification and force prediction. Second, our approach offered a new user-generic neural decoding approach for efficient and dexterous control of assistive robots.

II. MATERIALS

A. Participants

Eight participants (one female and seven males, aged between 21 to 34) without any known neural or muscular disorders were enrolled in the experiments. All participants provided informed consent with protocols reviewed and approved by the Institutional Review Board of the Pennsylvania State University in 2022. (IRB approval number: STUDY00021035).

B. Experiment Setup

During the experiment, participants were comfortably seated in a height-adjustable chair. Their forearms were positioned in a neutral posture and rested on a desk, cushioned by soft foam. To mitigate potential force contamination, both the palm and dorsal aspects of the hand were stabilized. As depicted in Fig. 1, an 8×20 electrode array (with a 3-mm electrode diameter and 10-mm inter-electrode distance) was placed over the extensor digitorum communis (EDC) of the forearm. The placement of this electrode array was determined by palpating the EDC muscle as the participants extended their fingers, with the HD-EMG array approximately centered at the midway between the olecranon processes and the styloid process. Four miniature load cells (SM-200 N, Interface) were attached to the index, middle, ring, and pinky fingers to measure the forces.

Notably, due to the significant enslaving effect between the ring and pinky fingers [21], [22], these fingers were consistently directed to extend simultaneously during the experiment. The forces produced by these two fingers were always summed up during subsequent data processing.

First, the maximum voluntary contraction (MVC) forces for each finger were measured for each participant. Next, the participants were instructed to perform two types of tasks. In the first task, designated as the “single-finger task”, the participants were directed to extend a single finger (treating the ring and pinky finger as one finger, hereinafter referred to as ring-pinky) to follow a trapezoidal force target, peaking at 50% MVC over a duration of 21 seconds. The participants were asked to avoid co-activation of the non-instructed fingers during this task. In the second task, termed “multi-finger task”, the target force was shaped as a series of trapezoids, interspersed with 1-second rest intervals. During the task, the participants were instructed to extend a minimum of two fingers to follow the presented force trapezoids, chosen at random before each trial. The multi-finger task duration varied based on the number of fingers involved: 36 seconds for three fingers and 12 seconds for two fingers). The peak contraction level remained at 50% MVC. Given that the multi-finger task aimed to simulate everyday finger motions, co-contractions of non-instructed fingers were not intentionally controlled. Each participant was required to perform fifteen single-finger trials and twenty-eight multi-finger trials. After each trial, a quality check was conducted. Some participants had difficulty completing the single-finger trials without co-activation. Therefore, we ensured that a minimum number of single-finger trials were completed after multiple attempts. During the multi-finger trials, if the researcher was uncertain about the correctness of a trial, participants were asked to repeat it. As a result, some participants completed more trials than others.

The finger force signals were obtained at a sampling rate of 1000 Hz and presented to the participants during the experimental trials. The EMG signals were acquired by the EMG-USB2+ (OT Bioelettronica) with a gain of 1000 and a sampling frequency of 2048 Hz, filtered with a 10–900 Hz bandpass filter.

III. METHOD

A. Data Preprocessing

We performed the data analysis using MATLAB (The MathWorks, Inc.), scikit-learn,¹ and Pytorch,² running on a computer equipped with an Intel i7-12700 k CPU and an Nvidia RTX 3070 Ti GPU. The motion artifacts removal was subsequently applied using the method detailed in [23]. The acquired force signals were normalized by the MVC value of each finger.

B. Feature Extraction

In our research, we initially explored four HD-EMG features frequently employed in myoelectric control studies [24]. These

features included: 1) Root Mean Square (RMS), 2) Wave Length (WL), 3) Zero Crossing (ZC), and 4) Slope Sign Change (SSC). Our results indicated that the best performance was achieved using only the RMS feature (the feature evaluation results are described in the supplementary material). To maintain computational efficiency, we only presented results using RMS. Specifically, the RMS was calculated for each channel using a moving window of 0.5 seconds with a step size of 50 ms. The normalized force was also smoothed employing the same window and step size. In subsequent analyses, the extracted RMS value functioned as the input $\mathbf{x} \in \mathbb{R}^{8 \times 20}$ for the decoder, while the preprocessed force $\mathbf{y} \in \mathbb{R}^3$ was used as the ground truth for prediction evaluations. For samples derived from the single-finger trials, an additional attribute p was assigned. This attribute could be one of the following: {index, middle, ring-pinky, rest}, indicating the currently activated finger. The “rest” state was determined by comparing the recorded forces with the 10% MVC threshold of the corresponding finger.

C. Autoencoder for Feature Disentanglement

Our framework is illustrated in Fig. 2. For an EMG data sample $\mathbf{x}$ at a given time period, it could be labeled by three attributes: the identity $s \in \{1, N_s\}$, the involved finger $p \in \{1, N_p\}$, and the produced force, where $N_s = 8$ and $N_p = 4$ are the number of enrolled subjects and involved fingers, respectively. $\mathbf{y}$ is a 3-dimensional vector where each dimension represents the force of one of the three fingers, and its values are continuous. Considering that HD-EMG features exhibit high consistency at different muscle contraction levels when the same finger is activated [15], we optimized our approach by merging the force attribute $\mathbf{y}$ into the finger p during feature learning. While this could result in a loss of information, we found that it streamlined the learning process for obtaining user-invariant and task-sensitive representations. Once these representations are extracted, the finger force information can be obtained readily using the linear regression function.

To this end, let $\mathbf{x}_i^j$ represents the input sample associated with identity s_i and finger p_j . Our goal was to progressively learn two types of information from $\mathbf{x}_i^j$: 1) a static component representing the inherent characteristics of the subjects and 2) a dynamic component encoding the finger. To achieve this, we implemented a two-branch autoencoder, comprising two encoders, E_p and E_s . The E_p was designed to capture task-sensitive finger information, while E_s was tailored to learn the static, user-sensitive characteristics. A decoder D , was subsequently utilized to reconstruct the original feature from these decomposed components.

To train the network, we constructed a loss function consisting of three components: the reconstruction loss and two triplet losses. For the reconstruction loss, we used the mean squared error (MSE) between the input and the output of the AE.

$$\mathcal{L}_{recon} = \mathbb{E}[\|D(E_p(\mathbf{x}_i^j), E_s(\mathbf{x}_i^j)) - \mathbf{x}_i^j\|] \quad (1)$$

During decomposition, we expected that the extracted latent vector $E_p(\mathbf{x}^j)$ with the same finger should be tightly clustered on the latent space. Specifically, the distance between vectors of different fingers should be larger than the distance between that

¹[Online]. Available: <https://scikit-learn.org/stable/>

²[Online]. Available: <https://pytorch.org/>

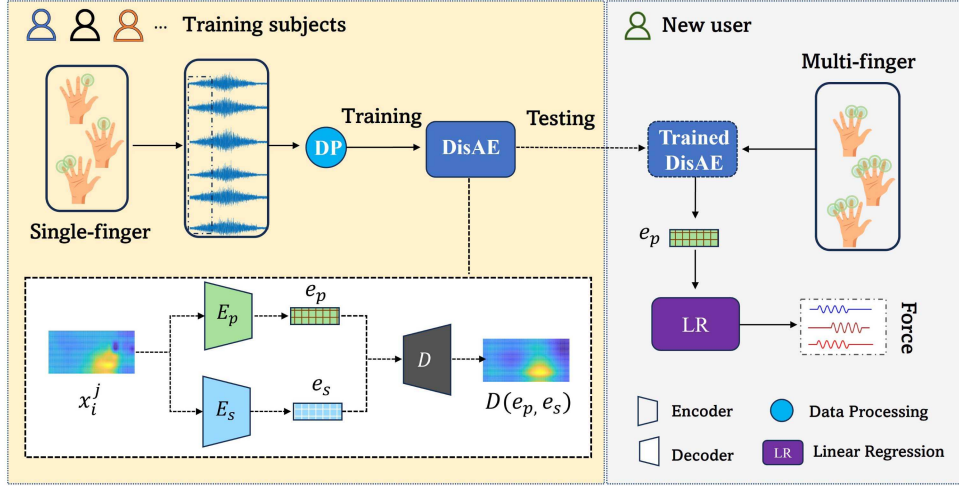


Fig. 2. Conceptual framework of the DisAE.

of the same finger. Thus, For E_p , we applied triplet loss on the finger latent space:

$$\mathcal{L}_p = \mathbb{E}[\|E_p(\mathbf{x}_i^j) - E_p(\mathbf{x}_i^k)\| - \|E_p(\mathbf{x}_i^j) - E_p(\mathbf{x}_m^k)\| + \alpha]_+ \quad (2)$$

where $\mathbf{x}_i^j$ ($l \neq i$) represents a sample shared the same finger with $\mathbf{x}_i^j$ but from a different subject. $\mathbf{x}(k \neq j)$ represents a sample that have a different p attribute with $\mathbf{x}_i^j$. α is the margin.

$$\mathcal{L}_s = \mathbb{E}[\|E_s(\mathbf{x}_i^j) - E_s(\mathbf{x}_i^l)\| - \|E_s(\mathbf{x}_i^j) - E_s(\mathbf{x}_m^k)\| + \alpha]_+ \quad (3)$$

where $\mathbf{x}_i^l$ ($l \neq j$) represents a sample that shares the same identity with $\mathbf{x}_i^j$ but represents a different finger. $\mathbf{x}_m^k$ ($k \neq i$) denotes a sample that have a different s attribute with $\mathbf{x}_i^j$.

Combining the three loss components gives the total loss:

$$\mathcal{L} = \mathcal{L}_{recon} + \lambda_1 \mathcal{L}_p + \lambda_2 \mathcal{L}_s + \lambda_3 \sum_i |w_i| \quad (4)$$

where λ_1 and λ_2 are the balancing weights, which were set as 0.5 in this study. The last term is the l_1 regularization to provide sparsity and avoid overfitting. The balancing weight λ_3 was set as 10^{-4} based on a grid search among $\{10^{-3}, 10^{-4}, 10^{-5}\}$

D. Force Estimation

By training the user-generic AE, we leveraged the E_p branch to decompose the original features into the finger latent space $\mathbf{e} = E_p(\mathbf{x})$. The next step involves using the disentangled latent vector to estimate the actual force via linear regression:

$$y^{(i)} = \mathbf{w}_i \mathbf{e} + C_i \quad (5)$$

where $y^{(i)}$ signifies the force of the i -th finger, $\mathbf{e}$ is the learned latent vector, $\mathbf{w}_i$ is the regression coefficient vector, and C_i is the intercept. To further remove feature redundancy, an l_1 regularization with a weight of 10^{-4} was incorporated during

the regression. The regression function was trained using data from the new subject to establish the relation between the latent vector and the force of individual fingers.

E. Validation Protocol

As illustrated in Fig. 2, we utilized a leave-one-subject-out validation to assess the efficacy of the developed DisAE. Specifically, we used data from one subject for testing and one subject for validation, with the remaining 6 subjects for training. During regression on the new user (testing subject), we selected two of the three-finger trials to train the linear regression function. To guarantee a thorough assessment, we adopted a leave-two-trial-out approach for all three-finger trials, ensuring every trial had a presence in both the regression and testing stages. The remaining multi-finger trials were tested on the regression function to evaluate the accuracy of the force prediction.

F. Implementation Details

The network structure of the DisAE method is detailed in Table I. All the encoders and the decoder were implemented using 2-D Convolutional Neural Networks (CNNs). Both encoders utilized convolution operations across the spatial dimension of the RMS map and incorporated down-sampling to obtain the latent vectors. In contrast, the CNN blocks in the decoder employed up-sampling processes to reconstruct the input samples. Training optimization was achieved using the Adam optimizer (Kingma and Ba, 2014) initialized at a learning rate of 0.002 and gradually decayed with a factor of 0.5 every 200 iterations. The maximum number of training epochs was set at 500, with a training batch size of 2000. The dropout probability (if applied) was set at 0.5. Validation was performed every 200 iterations on the validation data by evaluating the regression performance between the learned latent vector and the force signals. The model that yielded the highest coefficient of determination was selected for subsequent analyses.

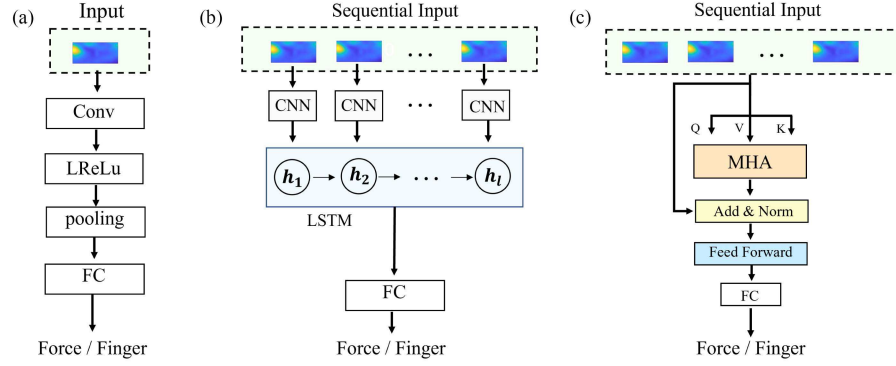


Fig. 3. Neural networks utilized for comparison include (a) CNN, (b) CNN+LSTM, and (c) Transformer. ‘Conv’ represents the convolutional layer with a filter size of 3 and a stride of 1. ‘LReLU’ stands for Leaky Rectified Linear Unit with a negative slope of 0.1. ‘Pooling’ refers to max pooling with a filter size of 2. ‘FC’ denotes a fully connected layer. ‘MHA’ indicates multi-head attention. ‘Add & Norm’ consists of the residual connection and layer normalization. When the output is ‘Force’, the networks were trained for force prediction, using continuous force values as labels. Conversely, when the output is ‘Finger’, the networks were trained for finger classification.

TABLE I
ARCHITECTURE OF THE NETWORK

Module	Layers	k	s	In/Out
Encoder E_p	Conv+ IN + LReLU	3	1	1/4
	Conv+ IN+ LReLU	3	2	4/8
	Conv+ IN+ LReLU	3	2	8/16
Encoder E_s	Conv+ IN+ LReLU	3	1	1/4
	Conv+ IN+ LReLU	3	2	4/8
	Conv+ IN+ LReLU	3	2	8/16
Decoder D	US+ Conv+ DO+ LReLU	3	1	32/16
	US+ Conv+ DO+ LReLU	3	1	16/8
	Conv $\times$ 2	3	1	8/1

In this depiction, ‘Conv’, ‘IN’, ‘LReLU’, ‘UpS’, and ‘DO’ stand for convolution, instance normalization, leaky ReLU, upsampling, and dropout layers, respectively. All convolution layers employ zero padding. The terms ‘k’ and ‘s’ represent kernel width and stride, respectively. The ‘In/Out’ column on the far right denotes the channel numbers for input and output.

We evaluated our DisAE method against two standard benchmarks: the personalized amplitude method (EMG-amp) and the Principal Components Analysis (PCA) [25], [26]. Notably, both benchmarks transform the original features into different representations akin to our DisAE approach. For the EMG-amp method, we incorporated a channel selection technique [27], [28] to the original features. This strategy allowed us to diminish finger co-activation, thereby improving multi-finger force prediction accuracy. Specifically, we began by choosing 60 channels (denoted as the channel pool C_i for the i -th finger) that had the highest average RMS values across the single-finger trials (note that motion artifacts have been removed from the EMG signals). Subsequently, for the channel pool C_i of each finger, we examined the correlation between the RMS of a channel in the pool and the forces of all fingers using the training multi-finger trials. If the highest coefficient of determination on a channel was linked to the i -th finger, it was retained in C_i . If not, the channel was excluded. Finally, the refined channel pool for each finger was employed to estimate the force via a linear regression

model. This strategy can be considered as a transformation to a sparse representation of the original feature.

For PCA, our objective was to investigate if the principal components extracted from the population could be generalized to the new user. Therefore, we used the same leave-one-subject-out validation protocol to determine the projection matrix. Only the principal components that accounted for 95% variance were kept, which were subsequently utilized to predict the forces of individual fingers.

In addition to the feature projection benchmark, we have compared DisAE with the state-of-the-art neural networks, which have proven effective in both sEMG-based pattern recognition and continuous kinematic/force predictions. These neural networks included CNN [19], CNN+LSTM [18] and Transformer [29]. The architectures of these neural networks are illustrated in Fig. 3, and the detailed parameters are presented in Table S.3 of the Supplementary Material. For CNN+LSTM and Transformer, sequential input was used, with a sequence length $L = 5$ (equivalent to an additional 200 ms delay compared to the one-to-one CNN network), selected to ensure the delay remained within the acceptable scope. All neural networks were trained separately under two tasks: finger classification and force prediction. For classification, the label was the ‘fingers’ as that in DisAE. The network weights were optimized using cross-entropy loss. For force prediction, the label was the continuous force exerted by the three fingers, and the network weights were optimized using the mean squared error (MSE) between the predicted and actual forces. The networks were trained using the same leave-one-subject-out validation protocol as DisAE, with the weights being updated using the Adam optimizer, set at a learning rate of 10^{-3} . The training process was capped at a maximum of 200 epochs, incorporating an early stopping strategy to prevent overfitting.

G. Performance Metrics

The accuracy of finger force prediction was evaluated using the root mean square error (RMSE) and the coefficient of determination (r^2). The ability of finger separation by different

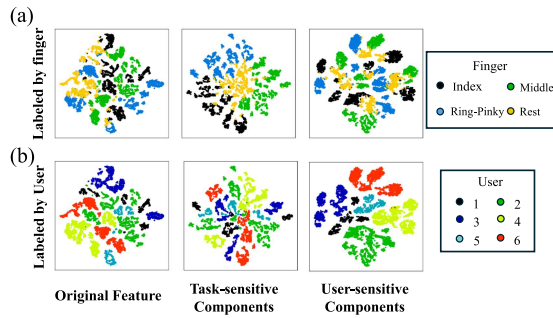


Fig. 4. Visualization of the t-SNE embeddings for (a) the original feature, (b) the disentangled task-sensitive component, and (c) the user-sensitive component. Both the top and bottom depict the same figure, but they are labeled by fingers and users, respectively.

methods was also investigated. The finger classification accuracy on the learned features by the k-nearest neighbor (KNN) was reported on the single-finger trials of the testing subjects. The false-active-rate and false-rest-rate were calculated at different threshold levels on multi-finger trials. Specifically, a finger was deemed “active” if the measured force surpassed a preset MVC threshold (5%, 10%, or 15%) during a given period. Conversely, a force below this threshold indicated a “rest” state of the finger. The false-rest-rate measures the percentage of “rest” samples labeled as “active”, while the false-active-rate denotes the percentage of ‘active’ samples incorrectly identified as ‘rest’.

The normality of the obtained metrics was tested using the Shapiro-Wilk test. For non-normally distributed variables, the Box-Cox transformation was used to fit the normality assumption. The repeated measures analysis of variance (ANOVA) was then used to analyze the influence of specific variables on the measured performance metrics. *Post hoc* pairwise comparisons with Bonferroni-Holm corrections were conducted when necessary. A significant level of $p < 0.05$ was used.

IV. RESULTS

A. Finger Classification

In Fig. 4, we used t-Distributed Stochastic Neighbor Embedding (t-SNE) to visualize the task-sensitive and user-sensitive components extracted from single-finger trials. It can be observed that clear decision boundaries were hard to discern in the original features when samples were labeled by either finger or user. In contrast, the extracted task-sensitive components displayed clear clustering when labeled by fingers. Similarly, the extracted user-sensitive component exhibited tighter clustering compared to the original features when labeled by users. Fig. 5 shows the RMS maps corresponding to different finger forces during a representative trial compared with their reconstructions via PCA and DisAE. Both PCA and DisAE appeared to preserve the intrinsic spatial activation patterns in the original RMS map for each finger. Interestingly, the map reconstructed from the task-sensitive latent vector via DisAE seemed to exhibit a more refined and localized spatial activation pattern, which was particularly evident for the reconstructed map corresponding to the index and middle finger tasks.

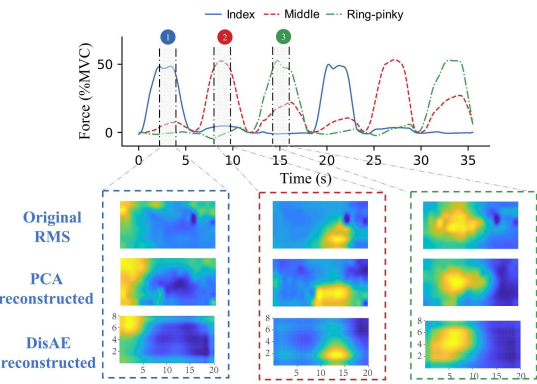


Fig. 5. RMS map, and the map reconstructed by PCA and DisAE during the index, middle, and ring-pinky finger extension in a representative trial.

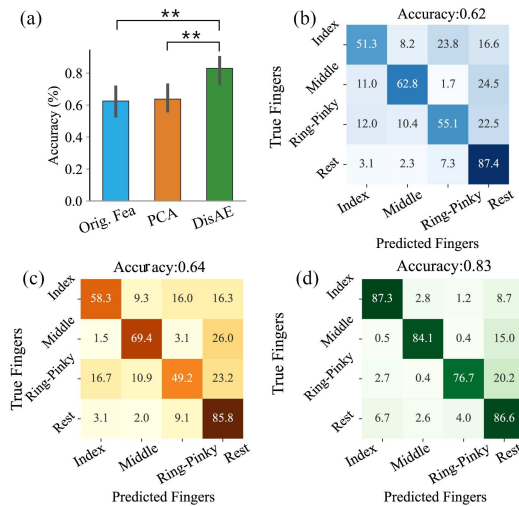


Fig. 6. (a) The average finger classification results across subjects by KNN using the original feature (Orig. Fea) and the feature projected by PCA and DisAE. (b) to (d): The confusion matrix obtained by DisAE, PCA, and the original feature. *: $p < 0.05$. **: $p < 0.01$.

KNN classifiers ($k = 1$) were employed on the original feature, the PCA-transformed feature, and the task-sensitive component extracted by DisAE. As depicted in Fig. 6, there was a substantial enhancement in classification accuracy in DisAE with an accuracy of 83% across subjects, marking a significant improvement over the original feature and PCA (overall accuracy at $p < 0.01$). The confusion matrix underscored the effectiveness of the task-sensitive information extracted by DisAE, particularly in the accurate classification of individual fingers (finger-wise F-1 score: $p < 0.05$ for three fingers). However, no significant difference was found in the detection accuracy in the “rest” state (F-1 score: $F(2, 21) = 0.18, p > 0.05$).

B. Finger Force Predictions

Fig. 7 displays the accuracy of the force prediction using the three approaches. With feature projection, DisAE exhibited superior performance to the PCA and EMG-amp methods, achieving RMSE values of 6.91 ± 0.45 (%MVC, mean

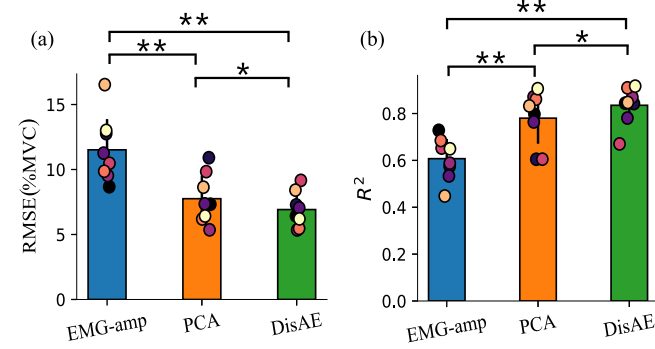


Fig. 7. Averaged force prediction results on target fingers. The results were obtained under LOSO validation and averaged across testing subjects, trials, and fingers. (a) RMSE; (b) R^2 . The results were obtained under LOSO validation and averaged across testing subjects, trials, and fingers. *: $p < 0.05$. **: $p < 0.01$.

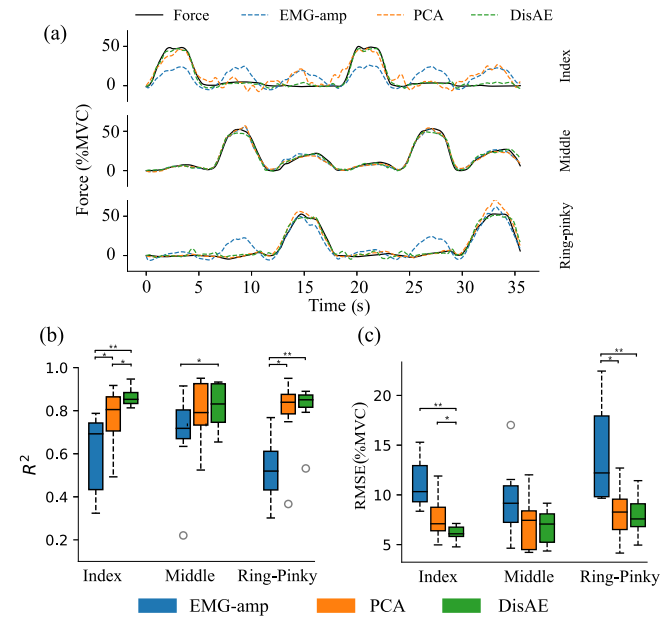


Fig. 8. (a) An exemplar trial of multiple finger extension. The RMSE (b) and R^2 (c) of the force estimation for individual fingers when the corresponding finger was instructed to move. *: $p < 0.05$, **: $p < 0.01$.

$\pm$ standard error), and R^2 values of 0.835 ± 0.026 . The PCA approach showed RMSE values of 7.72 ± 0.63 and R^2 values of 0.772 ± 0.039 . In contrast, the EMG-amplitude method demonstrated the worst performance, with RMSE values of 11.51 ± 0.83 and R^2 of 0.607 ± 0.029 , which is comparable with our previous studies [27]. The accuracy was significantly different among the three methods as indicated by one-way repeated measures of ANOVA (RMSE: $F(2, 21) = 30.90, p < 0.001$; R^2 : $F(2, 21) = 51.20, p < 0.001$). Further *post hoc* comparison confirmed the significant differences between DisAE and the other two methods ($p < 0.05$).

Fig. 8(a) shows a representative trial of force estimation for the index, middle, and ring-pinky fingers. While all methods aligned well with the recorded force of the middle finger, the EMG-amp method tended to overestimate both the index and

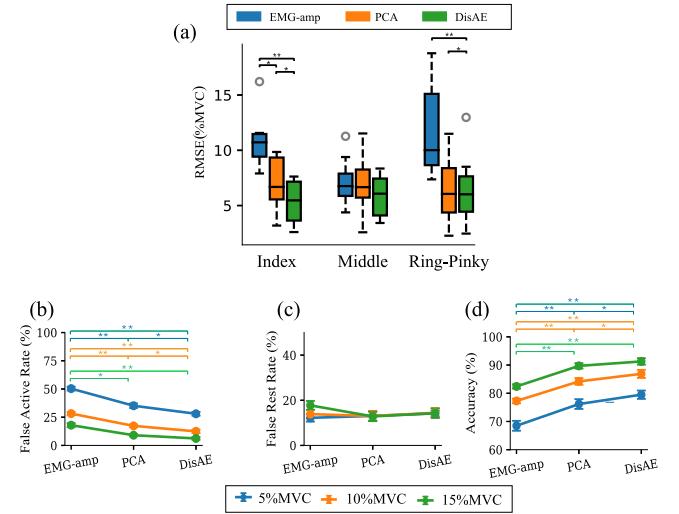


Fig. 9. (a) The average RMSE across all subjects on non-target fingers. (b) to (d): The false active rate, the false rest rate, as well as the accuracy for the active v.s. rest classification at 5%, 10%, and 15%MVC. *: $p < 0.05$, **: $p < 0.01$.

ring-pinky finger forces. Furthermore, The PCA force prediction of the index finger appeared to be influenced by the activities of other fingers. In contrast, our DisAE force estimation closely mirrored the actual forces of all fingers, maintaining a high level of accuracy even when multiple fingers were concurrently activated.

Fig. 8(b) and (c) offered a detailed examination of the force prediction performance of individual fingers. The DisAE method consistently showed the lowest prediction error, with the RMSE of the index (6.02 ± 0.31), middle (6.7 ± 0.63), and ring-pinky fingers (8.0 ± 0.84 %MVC). The two-way (method (EMG-amp vs. PCA vs. DisAE) $\times$ finger (index vs. middle vs. ring)) ANOVA revealed a significant effect of the method on the RMSE ($F(2, 14) = 22.70, p < 0.001$ and $R^2(F(2, 14)) = 16.89, p < 0.001$) with no significant effect on the finger factors and no interaction effect. Subsequent *post hoc* analysis revealed that the DisAE method significantly outperformed the EMG-amp in almost all conditions, except when compared to PCA, a significant difference was observed only in the index finger.

We also quantified the prediction errors of the three methods on non-targeted fingers (Fig. 9). Interestingly, the RMSE values of these non-target fingers were close to those of the target fingers. A two-way ANOVA (method $\times$ finger) indicated a significant effect of the method on RMSE ($F(2, 14) = 13.48, p < 0.001$). However, there was no significant effect on fingers, and no interaction effect between the method and finger factors was observed. Further *post hoc* analysis revealed that the DisAE method consistently registered significantly lower prediction errors for non-target fingers compared with EMG-amp, particularly for the index and ring-pinky fingers. However, no significant difference was found for the middle finger across any of the method comparisons. When comparing DisAE with PCA, a significant difference was only observed in the index finger.

TABLE II
PERFORMANCE COMPARISON FOR FINGER CLASSIFICATION

Method	Accuracy (%)	Macro-F1 (%)	Training Time (second / epoch)
CNN	80.49 ± 4.60	80.00 ± 4.70	0.249 ± 1.38
CNN+LSTM	80.99 ± 5.28	80.30 ± 5.18	1.801 ± 0.13
Transformer	80.88 ± 5.71	79.05 ± 5.91	1.596 ± 0.11
DisAE+KNN	83.0 ± 4.5	83.44 ± 3.20	1.601 ± 1.91

To assess the finger separation capabilities of the three methods in multi-finger tasks, we evaluated the active vs. rest detection accuracy. As depicted in Fig. 9(b), the DisAE method consistently exhibited the lowest false-active-rate across all thresholds (5%, 10%, and 15% MVC). In contrast, the EMG-amp method showed a notably higher false-active-rate, which was especially evident at the 5% MVC threshold. A two-way repeated measures ANOVA, factors of method and threshold, revealed a significant interaction effect ($F(4, 28) = 9.36, p < 0.01$). Subsequent *post hoc* analyses indicated that compared to EMG-amp, the DisAE method achieved a significantly lower false-active-rate across all thresholds. When compared with PCA, significant differences emerged at the 5% and 10% MVC thresholds. Observations from Fig. 9(c) demonstrate no significant differences among either methods or thresholds in the false-rest-rates. Regarding detection accuracy (Fig. 9(d)), DisAE outperformed its counterparts. The two-way ANOVA confirmed the significant effect of both factors (method and threshold) without any interaction effect (method: $F(2, 14) = 39.7, p < 0.001$; threshold: $F(2, 14) = 53.56, p < 0.001$). *Post hoc* comparisons underscored that the DisAE detection accuracy notably surpassed EMG-amp across all thresholds and surpassed PCA at the 10% and 15% MVC thresholds.

Lastly, we evaluated the efficiency of the developed approach based on both training and testing times. The specific training durations for the three methods are detailed in Table S.2 of the supplementary materials. Training with both EMG-amp and PCA took under 100 ms. In contrast, training the user-generic encoder took an average of 469.37 seconds, while training the personalized regression model only took 0.106 seconds. All approaches managed to process a 500-ms EMG signal within the 50-ms signal update interval. Remarkably, our method was able to generate latent features and predict EMG force in 1.6 ms.

C. Comparison With Other Deep Learning Methods

The performance of SOTA neural networks for finger classification was evaluated under a leave-one-subject-out validation protocol. The results were presented in Table II. All tested approaches achieved average classification accuracies above 80% on unseen subjects, with our DisAE yielding the highest performance. However, a one-way repeated measures ANOVA revealed no significant differences among the methods in terms of both accuracy ($F(3,21)=0.938, p > 0.05$) and the F1 score ($F(3,21)=0.951, p > 0.05$). There are significant differences in the training time cost per epoch among these four approaches

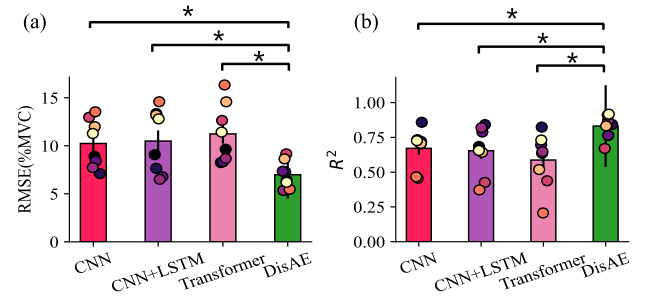


Fig. 10. Force prediction performance on target fingers. The results were obtained under leave-one-subject-out validation and averaged across testing subjects, trials, and fingers. *: $p < 0.05$. **: $p < 0.01$.

TABLE III
CLASSIFICATION AND REGRESSION PERFORMANCE ON ABLATION STUDIES

Configuration	Classification		Force Prediction	
	Acc	F1	RMSE	R^2
w/o two-branch encoder	76.86 ± 6.01	75.55 ± 6.45	8.57 ± 0.84	0.790 ± 0.030
w/o l_1 regularization	80.23 ± 3.87	79.12 ± 4.13	7.74 ± 0.87	0.819 ± 0.036
full implementation	83.0 ± 4.5	82.44 ± 4.0	6.91 ± 0.45	0.835 ± 0.026

($F(3, 21) = 48.98, p < 0.01$). Further *post-hoc* comparison indicated that CNN emerged as the most efficient approach ($p < 0.01$). In contrast, the CNN+LSTM model exhibited the longest training time relative to all other methods ($p < 0.01$). Compared to the Transformer, the DisAE showed an average increase in training duration of 5 ms, with this difference being statistically significant ($p < 0.01$).

Fig. 10 depicted the force prediction performance when applying the trained neural networks to the unseen subjects with the network weights fixed. All the end-to-end neural networks, when predicting data from unseen subjects, exhibited a high prediction error with RMSE values of 10.24 ± 1.19 , 10.50 ± 1.36 , and 11.24 ± 1.33 , and R^2 values of 0.671 ± 0.07 , 0.654 ± 0.08 , and 0.587 ± 0.101 for CNN, CNN+LSTM, and Transformer, respectively. The prediction accuracy was significantly lower than that obtained by DisAE (RMSE: $F(3, 21) = 10.81, p < 0.01$ and $R^2: F(3, 21) = 8.90, p < 0.01$). Further *post-hoc* comparison confirmed that DisAE had significantly more accurate force prediction outcomes than CNN, CNN+LSTM, and Transformer ($p < 0.05$).

D. Ablation Studies

Table III presents the results of the ablation studies, where we evaluated the effectiveness of the two-branch encoder and l_1 regularization. Without the two-branch encoder, essential for feature disentanglement, the autoencoder showed significantly lower classification accuracy compared with the full DisAE implementation ($p < 0.05$ for both accuracy and F1 score, as indicated by a paired t-test). Omitting l_1 regularization during training also resulted in a notable drop in classification performance ($p < 0.05$ for both accuracy and RMSE). Similarly, force prediction accuracy decreased significantly without the two-branch encoder, as evidenced by the higher RMSE and lower R^2 values ($p < 0.05$). A decline in regression performance

was also observed without the l_1 regularization, although these differences were not statistically significant. Overall, this ablation study highlights the critical role of each component in enhancing the DisAE's performance.

V. DISCUSSION

The current study exploited a feature disentanglement approach to enable a robust user-generic EMG-based neural decoder for finger extension force decoding. Our DisAE method, through an autoencoder network structure, can explicitly disentangle the task-sensitive features from the user-sensitive features in HD-EMG signals. Our results suggested that the learned task-sensitive features offered a more generic representation across users. Intriguingly, when we applied these latent features to both finger classification and individual finger force prediction, the DisAE-learned task-sensitive features demonstrated superior performance compared with the conventional EMG amplitude method and the commonly used feature projection PCA method. Our developed approach offers a novel decoding framework for user-generic decoding models that allow efficient clinical implementations. The outcomes pave the way for further development of robust neural-machine interfaces feasible for dexterous control of assistive robotic hands.

Previous studies have characterized the spatial activation of finger extensor muscles, highlighting finger-specific localized patterns evident across users using HD-EMG grids [15]. However, due to individual variations in anatomical and physiological factors, the muscle activation patterns for different persons often exhibit variations. These variations are also influenced by external factors, including inconsistencies in electrode placement, changes in electrode-skin impedance, or background noise [30]. As a result, it remains a challenge to establish a generic relation between muscle activation patterns and the corresponding motor outputs across users. In the context of individual finger force prediction, the HD-EMG grid covers substantial skin regions with minimum muscle activation, since only localized muscle compartments of the extensor muscle are active. Consequently, conventional EMG amplitude methods can be significantly improved by removing non-active channels for each finger movement through a “channel pool refinement” strategy [31], [32]. However, during multi-finger movements, the co-activation effects between fingers emerge, thereby introducing complex activation patterns. These complexities may not be adequately captured by the “channel mask” learned from single-finger tasks.

PCA is a widely accepted feature projection tool in HD-EMG analysis and more broad neural decoding. PCA has demonstrated efficacy in both pattern recognition and proportional control [25], [33], [34], [35]. In our study, this substantial reduction in dimensionality potentially eliminated interference in the original features, thereby greatly simplifying the force regression process and leading to improved results. However, a limitation of PCA is that data corresponding to tasks involving different fingers might not be distinctly clustered in the reduced feature space, as evidenced by the poor finger classification outcomes.

DisAE considers class separation during feature projection. This feature disentanglement paradigm has been used in various fields, such as gait recognition [36], motion retargeting [37], and voice conversion [38]. In line with these studies, our results revealed that the representations learned by DisAE form compact clusters in the respective latent spaces, namely the task space and the user space. Additionally, the DisAE representations displayed more distinct decision boundaries for classifying various finger tasks compared to the original features. This indicated a substantial improvement in classification performance, which is evident even in the cross-user validation protocol.

For force predictions across three fingers (index, middle, and ring-pinky), the DisAE representation clearly achieved the highest correlation with the recorded forces, delivering accurate force predictions even with concurrent activation of multiple fingers. No significant differences were detected across fingers in prediction error, especially for the middle finger. This could be attributed to the activation region of the middle finger being relatively distinct (most proximal), compared to the other two fingers [15]. Consequently, the middle finger force prediction was minimally influenced by co-activation from the other two fingers.

The active- vs. rest-state classification offers insights into the decoder capability in finger separation, a critical aspect of robotic hand control. By directly comparing the recorded forces with the prediction outcomes, we showed that DisAE consistently yielded a lower false-active-rate than the other two methods, which may enhance the reliability of prosthetic hand control in practice by addressing the overestimation issues. This improvement could be attributed to the attenuation of user-sensitive information in the original features. However, the problem of underestimation remained unaddressed with all methods exhibiting a high false-rest-rate. Future efforts should aim to identify and retain information that contributed to nuanced force variations.

In comparison with the state-of-the-art neural networks such as CNN, CNN+LSTM, and Transformer, we observed that these networks demonstrated a high level of generalizability in finger classification when tested on unseen subjects, comparable to our DisAE approach. However, for force prediction, the neural network models' performance significantly declined when applied directly to unseen subjects without adjusting the network weights. This observation is consistent with findings in other fields, such as computer vision, which suggest that achieving good cross-domain regression performance is more challenging than classification [29]. In contrast, our DisAE approach provides new insights into cross-domain regression by extracting task-sensitive information that is relevant to the variability between domains.

Our method is different from previous studies that sought to enhance the generalizability of neural decoders. Transfer learning has been the key approach to tackle the generalization issue, either in supervised settings [37], [39] or unsupervised settings [40], [41]. In supervised settings, once the classifier is trained, it is adjusted using limited labeled data from the new user. Unsupervised learning, on the other hand, seeks to find a feature space that minimizes domain shift. This is done

by using data from both the source domain (i.e., multiple-user cohort) and the new user to tune the model parameters. Though many studies target pattern recognition, these concepts have also been applied in proportional control. Jiang et al. [42] employed correlation-based data weighting to align data distribution across multiple users in elbow force estimation. A BERT-based network was introduced for continuous hand kinematics estimation, drawing on testing data from various sources [43]. Yet, the training of the neural networks in this study was not completely user-independent.

In contrast, our approach aligns more with domain generalization [44], aiming to model generic representations of all users. We address the adaptation challenge by extracting task-sensitive components from HD-EMG features. Our results showed that our feature projection establishes a more generalized relation between the HD-EMG feature and executed motor tasks. When applied to a new user, only a simple linear regression between the latent feature and motor output needs to be trained, without the need to adjust the weights of the DisAE. This process is highly efficient, requiring an average of only 0.106 seconds for regression learning.

Fundamentally, our DisAE model falls into the category of standard autoencoder frameworks, focusing on learning efficient representations of input EMG features and encouraging feature disentanglement through specific mechanisms (e.g. two-branch encoder structure and the triplet losses). Indeed, there are various autoencoder variants using the concept of variational autoencoders (VAE), such as β -VAE [45], The FactorVAE [46], and Ladder-AVE [47]. These VAEs are designed to explicitly learn the data distribution, which is beneficial to encourage the learning of statistically independent features in the latent space. Additionally, VAEs can function as generative models, capable of generating samples from the learned EMG feature distribution. This capability could be worth exploring for various applications, such as data augmentation and zero-shot machine learning in myoelectric control contexts. The optimization of our DisAE model in conjunction with other VAEs will be explored in future research.

Although the results were promising in the prediction of individual finger forces, several aspects merit further investigation in future work. First, the training of DisAE currently depends on recorded force to determine motor task labels. This presents a challenge when considering amputees, among whom individual-digit motor output may not be readily available. Consequently, it may be of importance to explore unsupervised methods that do not rely on data label input. Additionally, our current methodology mainly focuses on finger extensions. Clearly, further research is needed to incorporate both finger flexion and extension for effective control of advanced robotic hands. In this context, the interactions between fingers could be more intricate than those observed in the current study. Consequently, the encoder will need further adjustment to adapt to these new scenarios.

VI. CONCLUSION

Utilizing an autoencoder-based feature projection approach, we decomposed the HD-EMG features into user-invariant and

task-sensitive representations from user-sensitive and task-irrelevant representations. Our findings suggest that the learned representations provide a better distinction in performed motor tasks across different users. Moreover, the features obtained using our approach demonstrated a significantly higher force prediction accuracy compared with other approaches. In summary, our method contributed to a deeper understanding of the relation between HD-EMG and associated motor output, paving the way for advancements in user-generic neural interface techniques for assistive robot control.

REFERENCES

- [1] C. Castellini, "Upper limb active prosthetic systems—overview," in *Wearable Robotics*. Amsterdam, The Netherlands: Elsevier, 2020, pp. 365–376.
- [2] M. S. Johannes et al., "The modular prosthetic limb," in *Wearable Robotics*. Amsterdam, The Netherlands: Elsevier, 2020, pp. 393–444.
- [3] T. Worsnopp, M. Peshkin, J. Colgate, and D. Kamper, "An actuated finger exoskeleton for hand rehabilitation following stroke," in *Proc. IEEE 10th Int. Conf. Rehabil. Robot.*, 2007, pp. 896–901.
- [4] K. Tong et al., "An intention driven hand functions task training robotic system," in *Proc. IEEE Annu. Int. Conf. Eng. Med. Biol.*, 2010, pp. 3406–3409.
- [5] T. Bützer, O. Lambercy, J. Arata, and R. Gassert, "Fully wearable actuated soft exoskeleton for grasping assistance in everyday activities," *Soft Robot.*, vol. 8, no. 2, pp. 128–143, 2021.
- [6] A. T. Nguyen et al., "A bioelectric neural interface towards intuitive prosthetic control for amputees," *J. Neural Eng.*, vol. 17, no. 6, 2020, Art. no. 066001.
- [7] A. E. Schultz and T. A. Kuiken, "Neural interfaces for control of upper limb prostheses: The state of the art and future possibilities," *PM&R*, vol. 3, no. 1, pp. 55–67, 2011.
- [8] K. Volkova, M. A. Lebedev, A. Kaplan, and A. Ossadtchi, "Decoding movement from electrocorticographic activity: A review," *Front. Neuroinform.*, vol. 13, 2019, Art. no. 74.
- [9] M. Simao, N. Mendes, O. Gibaru, and P. Neto, "A review on electromyography decoding and pattern recognition for human-machine interaction," *IEEE Access*, vol. 7, pp. 39564–39582, 2019.
- [10] E. N. Kamavuako, E. J. Scheme, and K. B. Englehart, "Determination of optimum threshold values for EMG time domain features; a multi-dataset investigation," *J. Neural Eng.*, vol. 13, no. 4, 2016, Art. no. 046011.
- [11] L. Quitadamo et al., "Support vector machines to detect physiological patterns for eeg and emg-based human-computer interaction: A review," *J. Neural Eng.*, vol. 14, no. 1, 2017, Art. no. 011001.
- [12] Z. Lu, A. Stampas, G. E. Francisco, and P. Zhou, "Offline and online myoelectric pattern recognition analysis and real-time control of a robotic hand after spinal cord injury," *J. Neural Eng.*, vol. 16, no. 3, 2019, Art. no. 036018.
- [13] M. Jordanić, M. Rojas-Martinez, M. A. Mananas, and J. F. Alonso, "Prediction of isometric motor tasks and effort levels based on high-density emg in patients with incomplete spinal cord injury," *J. Neural Eng.*, vol. 13, no. 4, 2016, Art. no. 046002.
- [14] I. J. R. Martinez, A. Mannini, F. Clemente, A. M. Sabatini, and C. Cipriani, "Grasp force estimation from the transient EMG using high-density surface recordings," *J. Neural Eng.*, vol. 17, no. 1, 2020, Art. no. 016052.
- [15] X. Hu, N. L. Suresh, C. Xue, and W. Z. Rymer, "Extracting extensor digitorum communis activation patterns using high-density surface electromyography," *Front. Physiol.*, vol. 6, 2015, Art. no. 279.
- [16] N. v. Beek, D. F. Stegeman, J. C. v. d. Noort, D. H. Veeger, and H. Maas, "Activity patterns of extrinsic finger flexors and extensors during movements of instructed and non-instructed fingers," *J. Electromyogr. Kinesiol.*, vol. 38, pp. 187–196, 2018.
- [17] C. Dai and X. Hu, "Extracting and classifying spatial muscle activation patterns in forearm flexor muscles using high-density electromyogram recordings," *Int. J. Neural Syst.*, vol. 29, no. 01, 2019, Art. no. 1850025.
- [18] Y. Chen, C. Dai, and W. Chen, "Cross-comparison of EMG-to-force methods for multi-DoF finger force prediction using one-DoF training," *IEEE Access*, vol. 8, pp. 13958–13968, 2020.
- [19] G. Hajian, A. Etemad, and E. Morin, "Generalized EMG-based isometric contact force estimation using a deep learning approach," *Biomed. Signal Process. Control*, vol. 70, 2021, Art. no. 103012.
- [20] R. Merletti and P. J. Parker, *Electromyography: Physiology, Engineering, and Non-Invasive Applications*, vol. 11. Hoboken, NJ, USA: Wiley, 2004.

- [21] W. S. Yu, H. v. Duinen, and S. C. Gandevia, "Limits to the control of the human thumb and fingers in flexion and extension," *J. Neurophysiol.*, vol. 103, no. 1, pp. 278–289, 2010.
- [22] C. E. Lang and M. H. Schieber, "Human finger independence: Limitations due to passive mechanical coupling versus active neuromuscular control," *J. Neurophysiol.*, vol. 92, no. 5, pp. 2802–2810, 2004.
- [23] Y. Zheng and X. Hu, "Interference removal from electromyography based on independent component analysis," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 5, pp. 887–894, May 2019.
- [24] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE Trans. Biomed. Eng.*, vol. 50, no. 7, pp. 848–854, Jul. 2003.
- [25] D. Staudenmann, I. Kingma, A. Daffertshofer, D. F. Stegeman, and J. H. v. Dieën, "Improving EMG-based muscle force estimation by using a high-density EMG grid and principal component analysis," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 4, pp. 712–719, Apr. 2006.
- [26] G. R. Naik, S. E. Selvan, M. Gobbo, A. Acharyya, and H. T. Nguyen, "Principal component analysis applied to surface electromyography: A comprehensive review," *IEEE Access*, vol. 4, pp. 4025–4037, 2016.
- [27] Y. Zheng and X. Hu, "Concurrent prediction of finger forces based on source separation and classification of neuron discharge information," *Int. J. Neural Syst.*, vol. 31, no. 06, 2021, Art. no. 2150010.
- [28] Y. Zheng and X. Hu, "Concurrent estimation of finger flexion and extension forces using motoneuron discharge information," *IEEE Trans. Biomed. Eng.*, vol. 68, no. 5, pp. 1638–1645, May 2021.
- [29] X. Chen, S. Wang, J. Wang, and M. Long, "Representation subspace distance for domain adaptation regression," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 1749–1759.
- [30] J. W. Sensinger, B. A. Lock, and T. A. Kuiken, "Adaptive pattern recognition of myoelectric signals: Exploration of conceptual framework and practical algorithms," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 17, no. 3, pp. 270–278, Jun. 2009.
- [31] D. Yang and H. Liu, "An EMG-based deep learning approach for multi-DoF wrist movement decoding," *IEEE Trans. Ind. Electron.*, vol. 69, no. 7, pp. 7099–7108, Jul. 2022.
- [32] R. Roy, Y. Zheng, D. G. Kamper, and X. Hu, "Concurrent and continuous prediction of finger kinetics and kinematics via motoneuron activities," *IEEE Trans. Biomed. Eng.*, vol. 70, no. 6, pp. 1911–1920, Jun. 2023.
- [33] Y. Du, W. Hu, and L. Shyu, "The effect of data reduction by independent component analysis and principal component analysis in hand motion identification," in *Proc. IEEE 26th Annu. Int. Conf. Eng. Med. Biol. Soc.*, 2004, pp. 84–86.
- [34] J.-U. Chu, I. Moon, and M.-S. Mun, "A real-time EMG pattern recognition system based on linear-nonlinear feature projection for a multifunction myoelectric hand," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 11, pp. 2232–2239, Nov. 2006.
- [35] Y. Matsumura, M. Fukumi, and Y. Mitsukura, "Hybrid EMG recognition system by MDA and PCA," in *Proc. IEEE Int. Joint Conf. Neural Netw. Proc.*, 2006, pp. 5294–5300.
- [36] X. Gu, Y. Guo, F. Deligianni, B. Lo, and G.-Z. Yang, "Cross-subject and cross-modal transfer for generalized abnormal gait pattern recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 2, pp. 546–560, Feb. 2021.
- [37] K. Aberman, R. Wu, D. Lischinski, B. Chen, and D. Cohen-Or, "Learning character-agnostic motion for motion retargeting in 2D," *ACM Trans. Graph.*, vol. 38, no. 4, Jul. 2019, doi: [10.1145/3306346.3322999](https://doi.org/10.1145/3306346.3322999).
- [38] B. Sisman, J. Yamagishi, S. King, and H. Li, "An overview of voice conversion and its challenges: From statistical modeling to deep learning," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 132–157, 2021.
- [39] X. Chen, Y. Li, R. Hu, X. Zhang, and X. Chen, "Hand gesture recognition based on surface electromyography using convolutional neural network with transfer learning method," *IEEE J. Biomed. Health Inform.*, vol. 25, no. 4, pp. 1292–1304, Apr. 2021.
- [40] U. Côté-Allard, E. Campbell, A. Phinyomark, F. Lavolette, B. Gosselin, and E. Scheme, "Interpreting deep learning features for myoelectric control: A comparison with handcrafted features," *Front. Bioeng. Biotechnol.*, vol. 8, 2020, Art. no. 158.
- [41] E. Campbell, A. Phinyomark, and E. Scheme, "Deep cross-user models reduce the training burden in myoelectric control," *Front. Neurosci.*, vol. 15, 2021, Art. no. 657958.
- [42] 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.
- [43] C. Lin, X. Chen, W. Guo, N. Jiang, D. Farina, and J. Su, "A BERT based method for continuous estimation of cross-subject hand kinematics from surface electromyographic signals," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 87–96, 2023.
- [44] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain generalization: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 4396–4415, Apr. 2023.
- [45] I. Higgins et al., "beta-VAE: Learning basic visual concepts with a constrained variational framework," in *Int. Conf. Learn. Representations*, vol. 3, 2016.
- [46] S. Zhao, J. Song, and S. Ermon, "InfoVAE: Information maximizing variational autoencoders," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, no. 1, 2019, pp. 5885–5892.
- [47] C. K. Sønderby, T. Raiko, L. Maaløe, S. K. Sønderby, and O. Winther, "Ladder variational autoencoders," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 3745–3753.