Diffusion-based Learning for Cross Day Hand Gesture Recognition using HD-sEMG Signals

Kejia Su[a,b,c], Bo Wan[a,b,c,*], Jiayang Huang[a,b,c], Zhi-Qiang Zhang[d,*], Pengfei Yang[a,b,c] and Quan Wang[a,b,c]

[a] *School of Computer Science and Technology, Xidian University, Xi'an, 710126, China*
[b] *The Key Laboratory of Smart Human-Computer Interaction and Wearable Technology of Shaanxi Province, Xi'an, 710126, China*
[c] *The Shaanxi Engineering Research Center of High-confidence embedded computing, Xi'an, 710126, China*
[d] *School of Electronic and Electrical Engineering, Institute of Robotics, Autonomous Systems and Sensing, University of Leeds, Leeds, U.K*

ARTICLE INFO

Keywords:
high-density surface electromyography
cross-day gesture recognition
diffusion models
transfer learning

ABSTRACT

Gesture recognition using high-density surface electromyography (HD-sEMG) signals has attracted significant attention in myoelectric control. While recent studies report high intraday performance, interday accuracy often drops due to poor generalizability, limiting real-world deployment. To improve robustness, we propose a Diffusion-based Hand Gesture Recognition framework (DiffHGR) that integrates diffusion-based data augmentation with autoencoder representation learning. During training, a Diffusion (Diff) component corrupts HD-sEMG signals through a forward Gaussian diffusion process and employs a U-Net–based denoiser to reconstruct high-fidelity signals, which are used to augment the training set with diverse samples. Meanwhile, an Autoencoder (AE) component learns discriminative latent representations for gesture classification, enhanced via skip connections from the Diff encoder to reuse multi-scale denoising features. To address cross-day distribution shifts, we further introduce a lightweight few-shot calibration protocol. During calibration, the Diff is kept frozen and is used only as a generator to synthesize additional samples that augment the limited target-day data, while the AE encoder and classifier are updated for fast adaptation. During online inference, prediction is performed solely by the calibrated AE encoder and classifier, with the Diff generator inactive in the inference path, enabling low-latency deployment. Extensive experiments demonstrate that DiffHGR consistently outperforms other benchmark models. Real-time validation further confirms its robustness and practical applicability. These results highlight the effectiveness of combining diffusion-driven data augmentation and autoencoder-regularized representation learning for robust HD-sEMG-based gesture recognition.

1. Introduction

Hand gesture recognition (HGR) provides an intuitive, convenient, and natural human-computer interaction way. It has been applied in a wide range of applications, such as prosthesis control [1], interaction systems [2; 3] and virtual reality game [4]. The common HGR technologies mainly involve three types of sensors, i.e., data gloves [5], vision-based sensors [6] and surface electromyography (sEMG) [7]. Among these, high-density surface electromyography data (HD-sEMG) can capture detailed muscle activities, it thus has been widely used for HGR in the past decades [8–10].

The most common approaches to decode HD-sEMG signals into hand gestures are machine learning-based classifiers, such as linear discriminant analysis (LDA) [11], support vector machine (SVM) [12], random forest (RF) [13], and artificial neural network (ANN) [14]. However, these methods are hindered by cumbersome hand-crafted feature extraction [15], and the optimal combinations of hand-crafted features varied with different conditions [16]. Deep learning (DL) methods have recently proven to be a powerful model to extract complex hidden features automatically, thus they can learn the more robust and invariant representations of EMG signals than machine learning methods. For example, Yang et al. proposed a multi-stream residual network (MResLSTM) for dynamic hand movement recognition [17]. Karnam et al. introduced a hybrid CNN and Bi-LSTM architecture for hand activity classification [18]. Zhang et al. proposed a convolutional neural network with multi-attention for hand gesture recognition [19]. Montazerin et al. introduced a Vision Transformer (ViT) based method to recognize hand gestures [20]. However, these methods are generally trained and evaluated using sEMG signals collected on the same day. The performance of these methods may be seriously degraded when a trained model is tested with data collected on a different day. This degradation arises from the neglect of factors such as sensor misplacement, sensor displacement, and variations in human neurophysiology and skin conductivity across different days [8]. Thus, it is essential to improve the generalizability of these HGR methods by ensuring high reliability on different days [21].

To address the issue of cross-day variation and improve model generalization, transfer learning and adversarial learning have been widely adopted [4; 22; 23]. The fine-tuning technique is mainly one of the transfer learning methods. For instance, Côté-Allard et al. proposed three ConvNet architectures combined with a transfer learning

[*] Corresponding author.

kjsu@stu.xidian.edu.cn (K. Su); wanbo@xidian.edu.cn (B. Wan); huangjiayang@xidian.edu.cn (J. Huang); z.zhang3@leeds.ac.uk (Z. Zhang); pfyang@xidian.edu.cn (P. Yang); qwang@xidian.edu.cn (Q. Wang)
ORCID(s):

strategy for sEMG-based hand gesture recognition, demonstrating improved performance after transfer learning [22]. Chen et al. introduced an effective CNN+LSTM network and a finetuning framework for gesture recognition tasks [24]. Wang et al. proposed a CNN-AM model using an attention mechanism and transfer learning for sEMG-based hand gesture estimation [25]. Although these existing approaches have achieved high classification accuracy by using transfer learning, several challenges remain, which can be summarized as follows: (1) reliance on a large amount of labeled data, (2) limited generalization to new gestures, and (3) dependence on larger window sizes ranging from 150 ms to 300 ms. In response to these challenges, Hu et al. proposed the ViT-MDHGR method, which demonstrates the effectiveness of using short time windows and minimal calibration for multi-day hand gesture recognition [8]. While their work proves the feasibility of using small window sizes and few calibration trials to address cross-day variability, it does not explore the generalization to a larger set of gestures, which is critical for practical applications that require a broader gesture repertoire. Adversarial learning is another promising method to improve model generalization by generating diverse training data, addressing the requirement of a large amount of labeled data. For example, Chen et al. introduced a deep convolutional generative adversarial network (DCGAN) to enhance multiple-channel EMG data, with results showing that the synthetic data could increase the diversity of the original dataset [26]. Shi et al. developed a low-shot adversarial network incorporating physics-based information to estimate muscle and joint kinematics from sEMG signals [27]. Lee et al. proposed a recursive domain adversarial neural network with data synthesis, which updates the EMG classifier to a target day in a semi-supervised manner for robust cross-day HGR [28]. Lin et al. developed a robust framework named RoHDE based on GAN, and their findings indicated that the proposed RoHDE can generate synthetic HD-sEMG signals to simulate recording conditions affected by disturbances [29]. Despite their promising results, GAN-based methods face challenges in terms of training stability and sample diversity.

To overcome these limitations, diffusion models, an emerging class of deep generative models, have gained attention due to their stable training processes and ability to generate high-quality synthetic data [30; 31]. In this study, we propose a Diffusion-based Hand Gesture Recognition (DiffHGR) framework DiffHGR consists of two key components: the Diff component, which performs diffusion-based signal reconstruction and is used to generate high-fidelity synthetic HD-sEMG samples for data augmentation, and the AE component, which learns discriminative latent representations for gesture classification. The key contributions are summarized as follows:

1. Diff Component: Utilizes a U-Net architecture to perform the forward and reverse diffusion processes, capturing both low-level and high-level features from HD-sEMG signals. The model is trained to minimize the reconstruction loss between the generated synthetic signals and the original data, effectively enhancing signal reconstruction and augmenting training data.

2. AE Component: Extracts rich latent representations of HD-sEMG signals essential for accurate gesture classification. The AE benefits from skip connections from the Diff component, allowing it to correct for loss of information incurred during the denoising process and refine feature extraction.

3. Joint training between Diff and AE Components: The two components are trained in a unified framework under a composite objective, where the Diff component is optimized by its diffusion reconstruction loss, while the AE is optimized by the reconstruction-gap and classification losses. The denoising-aware features from the Diff encoder are skip-connected to assist AE decoding, thereby combining diffusion-driven augmentation with discriminative representation learning and improving cross-day/cross-subject generalization.

4. Few-Shot Calibration: DiffHGR achieves robust cross-day performance with minimal calibration. The framework demonstrates excellent performance with few-shot augmented data from a new day, achieving an average accuracy of 90.27% across 20 subjects, surpassing benchmark methods such as CNNAM (83.38%), ViT-MDHGR (58.15%), DANN_CRC (84.87%), and DCGAN (84.33%).

The remainder of this paper is organized as follows: Section 2 describes the proposed method DiffHGR in detail. Section 3 provides the experimental results and findings. Section 4 discusses the results and outlines potential directions for future work. Lastly, Section 5 concludes the findings of this work.

2. Methodology

This section first presents three public HD-sEMG datasets selected in our experiments and introduces preprocessing methods applied in each dataset. Secondly, we introduce the main framework of the proposed DiffHGR method, which includes a Diff component and an AE component. Finally, we describe the hyperparameter setting, benchmark methods, and evaluation metrics.

2.1. Datasets

2.1.1. Hyser PR dataset

The Hyser PR dataset [32], consisting of HD-sEMG data from twenty subjects (12 male, 8 female, 21-34 years old). The goal and the experimental protocol were explained to each participant. This dataset includes 34 gestures. HD-sEMG signals were acquired with a sampling rate of 2048 Hz, using four 8×8 array electrodes (256 channels in total). Two were placed on each of the extensor and flexor muscles. Each gesture is executed with 6 trials, each lasting one-second duration. Data was collected from two distinct days, with intervals ranging from 3 to 25 days, and are referred

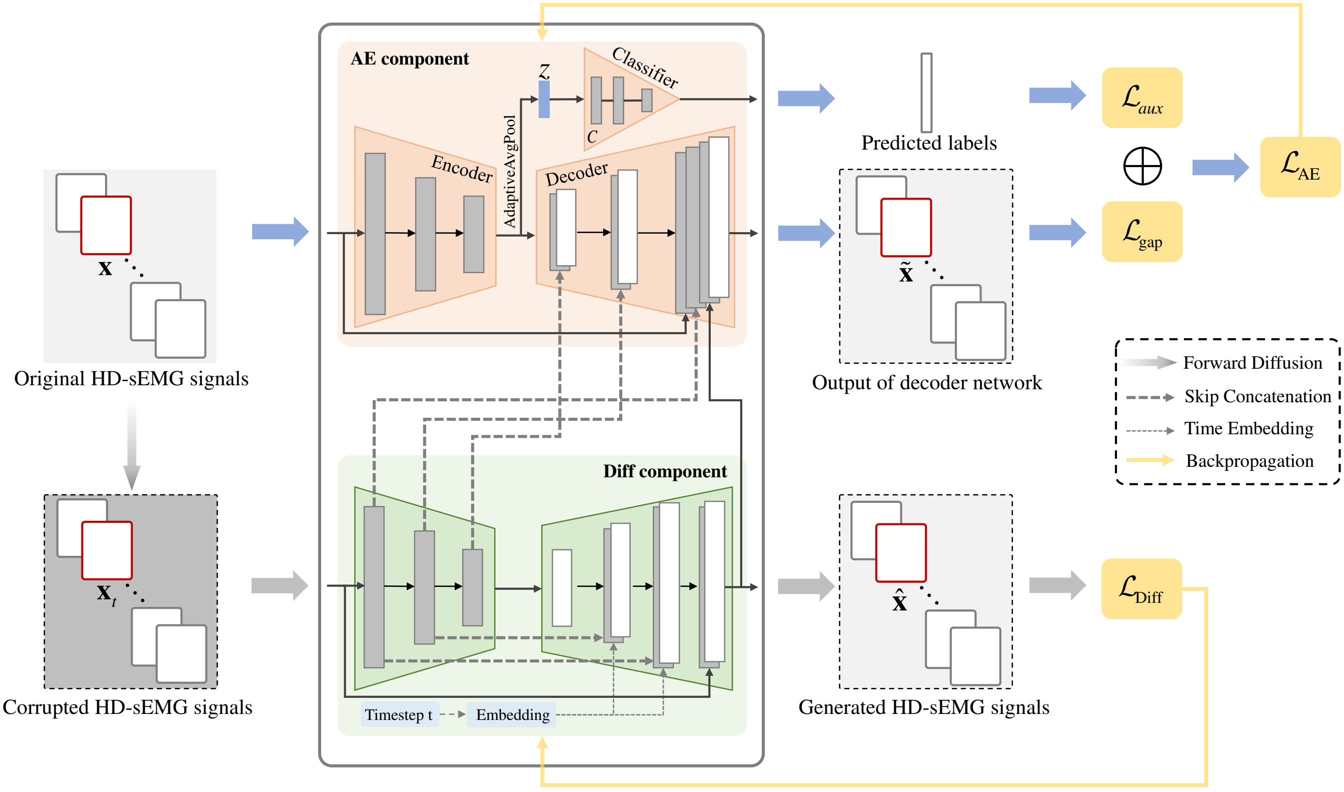


Figure 1: Detailed illustration of Stage A in DiffHGR. The Diff component reconstructs high-fidelity signals from corrupted inputs via a reverse diffusion process, while the AE component extracts discriminative features and enhances classification with skip connections from the Diff encoder. Both components are jointly optimized using reconstruction and classification losses.

to as session 1 and session 2, respectively. The acquired HD-sEMG signals are filtered by applying an eight-order, high-pass with a cutoff frequency of 10 Hz and an eight-order, low-pass with a cutoff frequency of 500 Hz Butterworth filters. A notch filter combination is then applied to reduce power line interference at 50 Hz and its harmonic components up to 400 Hz. Finally, we split the data using a 62.5 ms sliding window with a 10 ms sliding step.

2.1.2. *CapgMyo DB-a dataset*

The CapgMyo DB-a dataset [33], consisting of HD-sEMG data from 18 subjects. This dataset includes 8 distinct finger gestures, with each gesture being performed for a duration ranging from 3 to 10 seconds, followed by a 7-second rest period. HD-sEMG signals were acquired with a sampling rate of 1000 Hz, using 8 × 16 electrodes (128 channels in total). Each subject executed each gesture 10 repetitions. The power-line interference is removed and the acquired HD-sEMG signals are filtered by applying a second-order Butterworth band-stop filter (44-55 HZ). For each gesture and trial, the middle one-second window of data is used (1000 sample points). Finally, we split the data using a 128 ms sliding window with a 50 ms sliding step.

2.1.3. *CSL-HDEMG dataset*

The CSL-HDEMG dataset [34], consisting of HD-sEMG data from 5 subjects. This dataset includes 27 gestures. HD-sEMG signals were collected with a sampling rate of 2048 Hz, using 192 electrodes. Data of each subject was collected over 5 sessions and each gesture was performed 10 trials in each session. The powerline noise and cable motion artifacts are removed, and the acquired HD-sEMG signals are filtered by applying a fourth-order Butterworth band-pass filter (20-400 HZ). For each gesture and trial, the middle one-second window of data is used. Additionally, [34] points out that every eighth channel does not contain meaningful data. This paper ignored these channels, and a total of 168 channels of usable data were used.

2.2. Framework Overview

The main framework of the DiffHGR consists of three training stages. 1) **Stage A: Joint training of Diff and AE components**. As shown in Figure 1, Stage A jointly trains two core components: the Diff component for signal generation and the AE component for feature extraction and classification. The Diff component adopts a denoising diffusion probabilistic model (DDPM), where the original HD-sEMG signal $\mathbf{x}$ from the training set $\mathcal{D}_{\text{train}}$ is progressively noised into $\mathbf{x}_t$ and then reconstructed to $\hat{\mathbf{x}}$ using a U-Net-based reverse diffusion process. The reconstruction loss $\mathcal{L}_{\text{Diff}}$ is computed between $\hat{\mathbf{x}}$ and $\mathbf{x}$ to update the Diff component. Simultaneously, the AE component, consisting of encoder E, decoder D, and classifier C, processes the original input $\mathbf{x}$ to extract latent features $\mathbf{z}$ and reconstruct $\tilde{\mathbf{x}}$. Skip connections from the Diff encoder enhance feature representations. The AE loss $\mathcal{L}_{\text{AE}}$ comprises a reconstruction gap loss $\mathcal{L}_{\text{gap}}$ and an auxiliary classification loss $\mathcal{L}_{\text{aux}}$, computed using the classifier's output. Both $\mathcal{L}_{\text{Diff}}$ and $\mathcal{L}_{\text{AE}}$ guide the end-to-end optimization of the full network. 2) **Stage B: Synthetic sample generation and mixing**. Once the Diff component is trained, we apply it to generate synthetic HD-sEMG samples $\hat{\mathbf{x}}$ for each input $\mathbf{x}$ from the training set $\mathcal{D}_{\text{train}}$. These samples, denoted as $(\hat{\mathbf{x}}, y)$, preserve the original gesture labels and are collected into a synthetic set $\mathcal{S}_{\text{syn}}$. A random subset of $\mathcal{S}_{\text{syn}}$ is selected with ratio p, and mixed with the original training set to build a hybrid dataset $\mathcal{D}_{\text{mix}}$ used for training the final classifier. This strategy enhances data diversity while maintaining class consistency. 3) **Stage C: Discriminative classifier training**. As shown in Figure 2, the mixed dataset is fed into the frozen AE encoder, and the extracted latent embeddings $\mathbf{z}$ are input to the final classifier C^*. The classifier is trained from scratch using the cross-entropy loss $\mathcal{L}_{\text{C}}$ between predictions $\hat{\mathbf{y}}^*$ and ground-truth labels $\mathbf{y}^*$. This stage ensures that the classifier generalizes well to both original and generated samples. The detailed three-stage training pipeline is shown in Algorithm 1.

For an interday or intersubject scenario, we perform a lightweight few-shot calibration using a small number of labeled trials from the target condition. During calibration, the Diff component is kept frozen and used only to synthesize augmentation samples. We then adapt the DiffHGR in two steps: 1) we first update the encoder E of the AE component while others are frozen, so that the latent representation $\mathbf{z}$ aligns to the target condition while preserving task-relevant structure, 2) we subsequently update the classifier C^* using the combination of real and synthetic samples to refine the decision boundary. After calibration, inference prediction is performed by executing only the updated AE encoder and classifier, while the diffusion generator remains inactive in the inference path.

2.3. The Diff component based on DDPM

As illustrated in Figure 3, DDPM corrupts training data by gradually adding Gaussian noise in the forward diffusion process. It then learns to recover the corrupted data during the reverse process [30]. Consequently, a trained DDPM model can generate fake data from arbitrary Gaussian noise. Specifically, both the forward and reverse processes are defined as parameterized Markov chains. In the forward diffusion process, the original HD-sEMG data $\mathbf{x}$ can be denoted as $\mathbf{x}_0$, and the corrupted data after t steps can be defined as:

$$p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \tag{1}$$

$$p(\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_t, \cdots, \mathbf{x}_T \mid \mathbf{x}_0) = \prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) \tag{2}$$

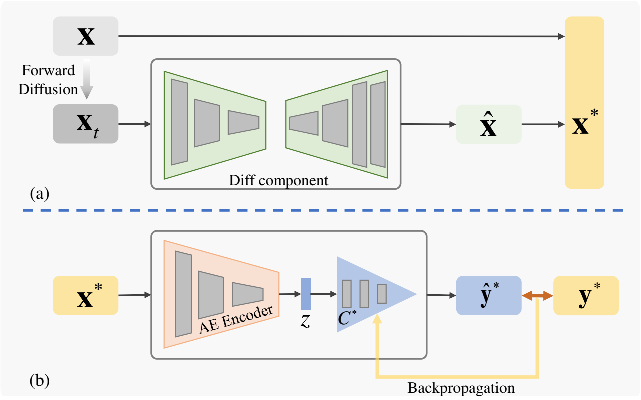


Figure 2: Illustration of Stage B and Stage C in DiffHGR. (a) The Diff component generates synthetic HD-sEMG samples $\hat{\mathbf{x}}$ from original signals. We augment training data with the generated HD-sEMG data. (b) We train a final classifier with the augmented HD-sEMG data.

Algorithm 1 DiffHGR Three-stage Training

```
Require: Training loader $\mathcal{D}_{\text{train}}$, test loader $\mathcal{D}_{\text{test}}$; epochs $E$; mix ratio $p$; weight $\alpha$
Ensure: Trained $\text{Diff}_\theta$, $\text{AE}_\phi$={$E_\phi, D_\phi, C_\phi$}, final classifier $C^*$
Initialize Diff (DDPM) $\text{Diff}_\theta$, AE ($E_\phi, D_\phi, C_\phi$), final classifier $C^*_\gamma$
Stage A: Joint training of Diff and AE
for epoch = $1 \dots E$ do
    for $(\mathbf{x}, \mathbf{y}) \in \mathcal{D}_{\text{train}}$ do
        $(\hat{\mathbf{x}}, \text{skips}, \_, \epsilon, t) \leftarrow \text{Diff}_\theta(\mathbf{x})$
        $\mathcal{L}_{\text{Diff}} \leftarrow \|\hat{\mathbf{x}} - \mathbf{x}\|_1$;   update $\theta$ on $\mathcal{L}_{\text{Diff}}$
        $\tilde{\mathbf{x}} \leftarrow D_\phi(\mathbf{x}, \hat{\mathbf{x}}, \text{skips}, t)$;  $\mathbf{z} \leftarrow E_\phi(\mathbf{x})$;  $\hat{\mathbf{y}} \leftarrow C_\phi(\mathbf{z})$
        $\mathcal{L}_{\text{gap}} \leftarrow \|\tilde{\mathbf{x}} - \text{stopgrad}(\mathcal{L}_{\text{Diff}})\|_1$
        $\mathcal{L}_{\text{aux}} \leftarrow \|\mathbf{y} - \hat{\mathbf{y}}\|_2$
        $\mathcal{L}_{\text{AE}} \leftarrow \mathcal{L}_{\text{gap}} + \alpha\,\mathcal{L}_{\text{aux}}$
        update $\phi$ on $\mathcal{L}_{\text{AE}}$
    end for
end for
Stage B: Synthesis and mixing (via Diff)
$\mathcal{S}_{\text{syn}} \leftarrow \emptyset$
for $(\mathbf{x}, \mathbf{y}) \in \mathcal{D}_{\text{train}}$ do
    $(\hat{\mathbf{x}}, \_, \_, \_, \_) \leftarrow \text{Diff}_\theta(\mathbf{x})$            ▷ reconstruction
    $\mathcal{S}_{\text{syn}} \leftarrow \mathcal{S}_{\text{syn}} \cup \{(\hat{\mathbf{x}}, \mathbf{y})\}$
end for
$\mathcal{S}_{\text{syn}} \leftarrow \text{RandomSubset}(\mathcal{S}_{\text{syn}}, p)$;   Build mixed loader $\mathcal{D}_{\text{mix}}$ from $\mathcal{D}_{\text{train}}$ and $\mathcal{S}_{\text{syn}}$
Stage C: Classifier training on encoder features of AE
for epoch = $1 \dots E$ do
    for $(\mathbf{x}^*, \mathbf{y}^*) \in \mathcal{D}_{\text{mix}}$ do
        $\mathbf{z} \leftarrow E_\phi(\mathbf{x}^*)$;  $\hat{y}^* \leftarrow C^*_\gamma(\mathbf{z})$
        $\mathcal{L}_{\text{C}} \leftarrow \text{CE}(\hat{\mathbf{y}}^*, \mathbf{y}^*)$;   update $\gamma$ on $\mathcal{L}_{\text{C}}$
    end for
    Evaluate on $\mathcal{D}_{\text{test}}$
end for
```

Table 1
The structure of the Diff component, which adopts a U-Net-inspired structure with three downsampling and three upsampling blocks, followed by a final convolutional layer. GroupNorm and PReLU are used throughout to stabilize training and enhance non-linearity.

Layer	Type	Input Shape	Output Shape
1	Conv1d + GroupNorm + PReLU + MaxPool1d	[batchsize, 256, 128]	[batchsize, 128, 64]
2	Conv1d + GroupNorm + PReLU + MaxPool1d	[batchsize, 128, 64]	[batchsize, 256, 32]
3	Conv1d + GroupNorm + PReLU + MaxPool1d	[batchsize, 256, 32]	[batchsize, 384, 16]
4	Upsample + Conv1d + GroupNorm + PReLU	[batchsize, 384, 16]	[batchsize, 128, 32]
5	Upsample + Conv1d + GroupNorm + PReLU	[batchsize, 128, 32]	[batchsize, 128, 64]
6	Upsample + Conv1d + GroupNorm + PReLU	[batchsize, 128, 64]	[batchsize, 256, 128]
7	Conv1d	[batchsize, 256, 128]	[batchsize, 256, 128]

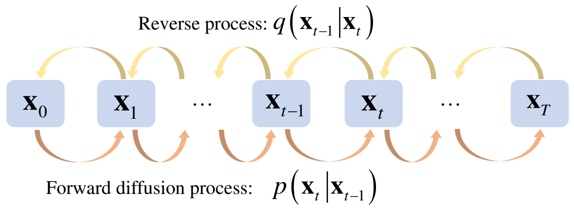

Reverse process: $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$
$\mathbf{x}_0$
$\mathbf{x}_1$
$\cdots$
$\mathbf{x}_{t-1}$
$\mathbf{x}_t$
$\cdots$
$\mathbf{x}_T$
Forward diffusion process: $p(\mathbf{x}_t|\mathbf{x}_{t-1})$

Figure 3: Denoising diffusion probabilistic model is shown. $\mathbf{x}_0$ represents the original HD-sEMG signals, and $\mathbf{x}_T$ represents the corrupted data after T steps transformation.

where T is the total number of diffusion steps, β_t is from a fixed variance schedule, and $\mathbf{x}_t = \sqrt{1-\beta_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\epsilon_t, \epsilon_t \sim \mathcal{N}(0,\mathbf{I})$ is latent variable of DDPM. According to the rule of the sum of normally distributed random variables, we can directly sample $\mathbf{x}_t$ from the original data $\mathbf{x}_0$ for arbitrary t with :

$$p(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}) \tag{3}$$

Here, $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \epsilon \sim \mathcal{N}(0,\mathbf{I}), \alpha_t = 1-\beta_t$, and $\bar{\alpha}_t = \prod_{i=1}^{t}$. Finally, the data $\mathbf{x}_0$ can be transformed into $\mathbf{x}_T \sim p(\mathbf{x}_T)$, where $p(\mathbf{x}_T) \sim \mathcal{N}(\mathbf{x}_T; 0, \mathbf{I})$.

The reverse diffusion process learns the reversal of the forward process, thereby recovering the original data distribution. Ho et al. [30] proposed training a neural network to predict the noise added during the forward process. It starts with standard Gaussian noise sampled from $q(\mathbf{x}_T) \sim \mathcal{N}(\mathbf{x}_T; 0, \mathbf{I})$. The reverse process is described as follows:

$$q(\mathbf{x}_{t-1} \mid \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)) \tag{4}$$

$$q(\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_t, \cdots, \mathbf{x}_T) = q(\mathbf{x}_T)\prod_{t=1}^{T} q(\mathbf{x}_{t-1} \mid \mathbf{x}_t) \tag{5}$$

where $\mu_\theta(\mathbf{x}_t, t)$ and $\Sigma_\theta(\mathbf{x}_t, t)$ represent the mean and the covariance obtained by training a network $\epsilon_\theta(\mathbf{x}_t, t)$. The training objective of this network is to ensure the predicted noise is consistent with the actual added one as follows:

$$\mathcal{L} = \mathbb{E}_{\mathbf{x}_0,\epsilon,t}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|^2\right] \tag{6}$$

In contrast, given the corrupted HD-sEMG data of each window $\mathbf{x}_t$, this study trains a Diff component to predict the uncorrupted signal, instead of estimating the added noise. During training, the forward diffusion step t is randomly sampled for each sample to expose the model to different noise levels, whereas the reverse process uses a single-step denoising without iterative reverse diffusion. As shown in Figure 1, the Diff component mainly comprises three downsampling convolution layers, three upsampling convolution layers, and an additional one-dimensional convolution layer. The detailed structure of the Diff component network is shown in Table 1. After forward propagation, the output of the Diff component $\hat{\mathbf{x}}$ is obtained. We employ $L1$ loss as the loss function to measure the absolute differences between the Diff component output and the original signal:

$$\mathcal{L}_{\mathrm{Diff}} = \|\hat{x} - x\|_1 \tag{7}$$

The objective function encourages the Diff component to produce output similar to the uncorrupted data. The parameters of this component can be updated as follows.

$$\theta \leftarrow \theta - \eta_1 \nabla_\theta \mathcal{L}_{\mathrm{Diff}}(\theta) \tag{8}$$

where η_1 is the learning rate.

2.4. The AE component

The forward process may introduce information loss, which the AE component aims to mitigate by identifying and rectifying these losses to extract meaningful representations for gesture classification tasks. The AE component is composed of an encoder E, a decoder D, and an auxiliary classifier C. The encoder processes the original HD-sEMG data $\mathbf{x}$ and maps it into a compressed latent space, while the decoder network reconstructs the signal from the latent representation. They are the same structure as the Diff component, excluding the last one-dimensional convolution (Conv1D) layer. The classifier is composed of three linear layers. The detailed structure is shown in Table 2. The encoder-decoder pair is trained jointly with the Diff component. Specifically, during joint training, the decoder D receives multi-scale skip connections from corresponding layers of the Diff encoder, allowing it to reuse denoising-aware intermediate features

Table 2
The structure of the AE component. The encoder reduces temporal dimensions to extract compact features, the decoder reconstructs signals from latent space, and the classifier predicts gesture labels from the latent embedding.

Layer		Type	Input Shape	Output Shape
Encoder	1	Conv1d + GroupNorm + PReLU + MaxPool1d	[batchsize, 256, 128]	[batchsize, 256, 64]
	2	Conv1d + GroupNorm + PReLU + MaxPool1d	[batchsize, 256, 64]	[batchsize, 256, 32]
	3	Conv1d + GroupNorm + PReLU + MaxPool1d	[batchsize, 256, 32]	[batchsize, 256, 16]
Decoder	1	Upsample + Conv1d + GroupNorm + PReLU	[batchsize, 256, 16]	[batchsize, 128, 32]
	2	Upsample + Conv1d + GroupNorm + PReLU	[batchsize, 128, 32]	[batchsize, 128, 64]
	3	Upsample + Conv1d	[batchsize, 128, 64]	[batchsize, 256, 128]
Classifier	1	Linear+ GroupNorm + PReLU	[batchsize, 256]	[batchsize, 512]
	2	Linear+ GroupNorm + PReLU	[batchsize, 512]	[batchsize, 512]
	3	Linear	[batchsize, 512]	[batchsize, 34]

and compensate for potential information loss introduced by the forward diffusion. In addition, both the original signal $\mathbf{x}$ and the denoised output of the Diff component $\hat{\mathbf{x}}$ are concatenated to the last decoding stage via skip connections, which further encourages structurally consistent reconstruction. Meanwhile, the classification objective imposed on C regularizes the AE to learn discriminative representations for robust gesture recognition. By integrating these connections, the decoder can leverage the structural information of the original signal along with the details reconstructed by the Diff component, learning more meaningful representations.

To improve the feature representation, the output of the encoder is passed through an adaptive average pooling layer, which aggregates the features into a fixed-size representation $\mathbf{z}$. This compressed representation $\mathbf{z}$ serves as the input to the classifier C, which is responsible for predicting the gesture class label. C is jointly trained with the encoder and decoder networks. The objective function for training the AE component is defined as:

$$\mathcal{L}_{\mathrm{AE}} = \|\widetilde{\mathbf{x}} - \mathrm{stopgrad}(\mathcal{L}_{\mathrm{Diff}})\|_1 + \alpha\|\mathbf{y} - \hat{\mathbf{y}}\|_2 \tag{9}$$

where $\mathbf{y}$ is the true labels, $\hat{\mathbf{y}}$ is the predicted labels, stopgrad$(\cdot)$ denotes the stop-gradient operation, and α is a hyperparameter. The parameters of this component can be updated as follows.

$$\phi \leftarrow \phi - \eta_2 \nabla_{\phi} \mathcal{L}_{\mathrm{AE}}(\phi) \tag{10}$$

where η_2 is the learning rate.

The Diff and AE components are trained jointly in a multi-objective optimization framework. The Diff component generates diverse synthetic data that captures the essential features of the original HD-sEMG signals. At the same time, the AE component extracts discriminative representations for gesture classification, strengthened by denoising-aware multi-scale features delivered through skip connections from the Diff encoder.

2.5. Data augmentation and gesture classification

After training Diff and AE, the trained Diff component can be used to generate high-quality synthetic HD-sEMG data. As shown in Figure 2, the original HD-sEMG data is augmented by adding an equal amount of synthetic HD-sEMG data, resulting in the augmented dataset denoted as $\mathbf{x}^*$, increasing the diversity of the training data. The augmented data $\mathbf{x}^*$ is then fed into the trained AE encoder network E, which maps the data into a latent representation by collapsing the time dimension into a single feature vector. This latent representation, which captures the underlying patterns in the HD-sEMG data, serves as the input to a classification network C^*. The classification network consists of three linear blocks. Each of the first two blocks contains a linear layer followed by group normalization and a PReLU activation function, which helps the network learn nonlinear representations. The final linear layer in C^* performs the gesture classification, outputting the predicted gesture labels. Finally, we calculate cross-entropy loss between the true labels and the predicted labels and backpropagate to update C^*.

2.6. Hyperparameter Setting

The training of the proposed DiffHGR framework involves three consecutive stages (as illustrated in Algorithm 1). In Stage A, we jointly train the Diff component and the AE component using two RMSProp optimizers and two cyclic learning rate schedulers. The base learning rate of the RMSProp optimizer is set to 9×10^{-5}, and the maximum learning rate is set to 1×10^{-3}. A batch size of 64 is used, and the training runs for 200 epochs. An adaptive learning rate reduction strategy is applied: the learning rate is reduced by a factor of 10 if the validation loss does not improve for 10 consecutive epochs, and training is terminated early after three such reductions. The loss weight α that balances the AE reconstruction and classification terms is empirically set to 0.1. In Stage B, we use the trained Diff component to generate synthetic samples and randomly select a portion (p is set to 0%, 25%, 50%, 75%, or 100%) to augment the original training set. In Stage C, the final classifier C^* is trained using the Adam optimizer with a learning rate of 1×10^{-5}. During this stage, only the classifier parameters are updated while the encoder is frozen. Each training stage is evaluated on the held-out test set to monitor generalization performance and ensure reproducibility.

2.7. Benchmark Methods

To evaluate the advantages of the proposed DiffHGR in cross-day hand gesture recognition, we compare our method with four state-of-the-art benchmark methods. These benchmarks were chosen based on their ability to handle cross-day variability and their effectiveness in gesture recognition tasks. CNNAM with transfer learning (CNNAM_TL) [25] comprises a CNN-based feature extractor composed of three two-dimensional convolution layers integrated with attention modules and a label classifier containing three fully connected layers. ViT-MDHGR [8] is a compact ViT-based network for multi-day dynamic hand gesture prediction, which captures crossday features by learning the relationships between HD-sEMG signals at any two timestamps within a window. DANN_CRC [28] is a recursive DANN structure with CRC data synthesis to augment the unlabeled EMG signals of the target day for robust cross-day HGR. DCGAN [26] consists of a generator and a discriminator. The generator and discriminator are adversarially trained. To further evaluate the ability to generate data, the training data is augmented with synthetic data generated by the trained generator. Then, the augmented data is fed into a MobileNet classifier for training.

2.8. Evaluation Metrics

To comprehensively evaluate the effectiveness of the proposed DiffHGR framework, we employ both classification metrics and generative quality metrics. For classification performance, we report standard metrics including classification accuracy, precision, recall, and F1 score. They are calculated by:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{11}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{12}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{13}$$

$$\text{F1 Score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{14}$$

where TP represents the number of true positives, FP represents the number of false positives, TN and FN represent the number of the true negatives and false negatives, respectively.

To further assess the quality and diversity of the synthetic HD-sEMG data generated by the Diff component, we adopt two widely used generative evaluation metrics: Inception Score (IS) and Fréchet Inception Distance (FID). The IS evaluates the diversity and semantic clarity of the generated data by measuring the KL divergence between the conditional label distribution and the marginal distribution. A higher IS score indicates that the generated samples are both diverse and confidently classifiable:

$$\text{IS}(G) = \exp\left(\mathbb{E}_{x \sim p_g}\left[\text{D}_{\text{KL}}\left(p(y|x) \parallel p(y)\right)\right]\right) \tag{15}$$

where $p(y|x)$ denotes the predicted label distribution for a generated sample x, and $p(y)$ is the marginal distribution across all generated data.

The FID score evaluates the similarity between the real and generated data distributions in feature space. A lower FID indicates that the generated data is more similar to the real data:

$$\text{FID} = \left\|\mu_r - \mu_g\right\|_2^2 + \text{Tr}\left(\Sigma_r + \Sigma_g - 2\left(\Sigma_r \Sigma_g\right)^{\frac{1}{2}}\right) \tag{16}$$

where μ_r, Σ_r and μ_g, Σ_g represent the means and covariances of features extracted from real and generated data, respectively.

These metrics, originally designed for image data, are adapted to our HD-sEMG classification scenario by replacing the standard Inception network with task-specific models trained on our datasets. Specifically, first, the trained Diff component is used to generate synthetic HD-sEMG signals. The generated data, along with an equal number of real data, is passed through the pretrained encoder of the AE component, which transforms each input into a compressed latent representation. These feature embeddings are used for downstream IS and FID calculations. The extracted features are input to the trained gesture classifier, producing softmax outputs $p(y|x)$ over gesture classes. These probabilities are used to compute the IS score.

3. Results

3.1. Evaluation on Hyser Dataset

This study primarily evaluates the proposed method on the Hyser dataset, which consists of two sessions collected on different days, providing a real-world cross-day evaluation. Specifically, we conduct leave-one-out cross-validation in intrasession experiments for selecting test and validation trials needed in intersession experiments. Furthermore, the intrasession performance serves as a baseline for comparison with intersession performance to assess the tolerable decrease in performance in cross-day hand gesture recognition tasks.

3.1.1. Results of intrasession

We evaluate the performance of the proposed DiffHGR method in session 1 (Day 1) and session 2 (Day 2), respectively. These results are intended to serve as a baseline for comparison with intersession performance. The intrasession experimental results assess the ability of DiffHGR to recognize gestures when trained and tested under consistent conditions. We evaluate intrasession performance using leave-one-trial-out cross-validation, with each subject conducting

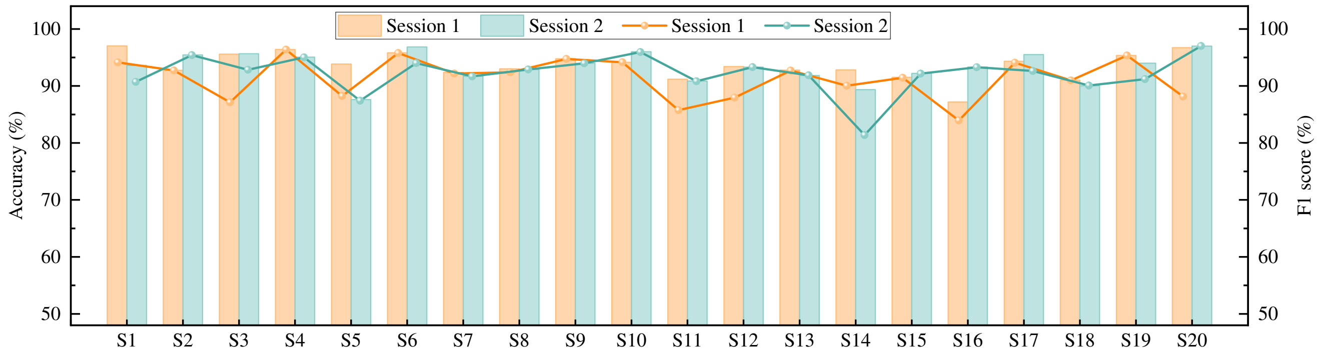


Figure 4: Intrasession classification performance across 20 subjects. Bar plots indicate the classification accuracy for each subject, while the overlaid lines show the corresponding F1 scores.

6 training and testing experiments. The results are shown in Figure 4 across 20 subjects. For each subject, we report the best classification accuracy among 6 experiments, along with the corresponding F1 score. The results demonstrate that the proposed DiffHGR method achieves high classification accuracies across all subjects. In session 1, the DiffHGR achieves the highest accuracy of 97.03% (S1) and the lowest of 87.19% (S16). Similarly, in session 2, the accuracy ranges from 87.6% (S5) to 97% (S20). The consistent performance across different subjects illustrates the stability and reliability of the DiffHGR method when applied within a single-day scenario.

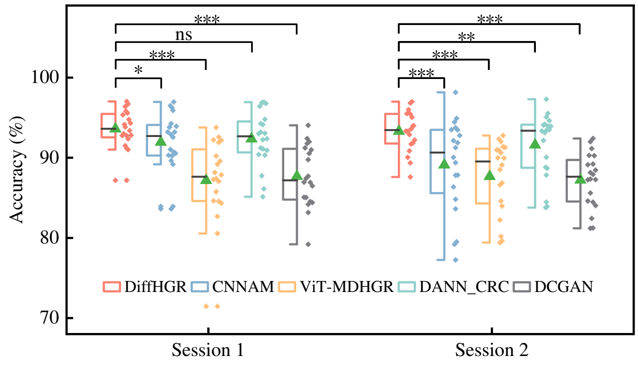


Figure 5: Intrasession performance comparison between the proposed DiffHGR and other benchmark methods, illustrated through box plots. Statistical significance was assessed via paired t-tests between DiffHGR and each baseline method ($*p < 0.05$, $**p < 0.01$, $***p < 0.001$, "ns" indicates no significant difference).

Additionally, a comparative analysis of the DiffHGR method with four benchmark methods is presented in Figure 5. These comparisons underline the effectiveness of the proposed approach relative to other methods in terms of classification accuracy and robustness within a session. As shown in Figure 5, DiffHGR significantly outperforms all benchmark methods in both sessions. It achieves the highest average accuracies of 93.6% for session 1 and 93.31% for session 2 across 20 subjects. We also note that the proposed method has the lowest variances among 20 subjects of 2.3% for session 1 and 2.48% for session 2. The low standard deviations indicate that the performance of DiffHGR is stable across different subjects within the same session. This stability is crucial for practical applications, as it suggests the proposed model can reliably recognize gestures without significant performance fluctuations. For example, the maximum accuracy difference between subjects in session 1 is 9.84%, compared to larger variations seen in CNNAM, ViT-MDHGR, DANN_CRC, and DCGAN, where the differences are 13.34%, 22.31%, 11.8%, and 14.85% between subjects. Additionally, the maximum accuracy difference between subjects in session 2 for CNNAM is above 20%, while the DiffHGR achieves consistency between different sessions. We also performed paired t-tests between DiffHGR and the other methods across all subjects, and the results show that DiffHGR achieves statistically significant improvements in accuracy over other baseline methods.

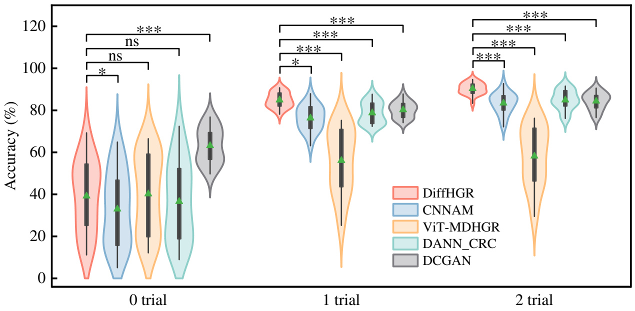


Figure 6: Comparison results of intersession performance between the proposed DiffHGR and other benchmark methods. Green triangles denote the mean values, while the black boxes within the violin plots represent the interquartile range, spanning from the 25th percentile to the 75th percentile.

3.1.2. Results of intersession (cross-day)

This study aims to achieve high and stable hand gesture recognition performance with a few-shot data for calibration. To investigate the minimum needed trials for calibration, we conduct intersession experiments by introducing transfer learning. The HD-sEMG data from session 1 is utilized for training and validation, while the data from session 2 is

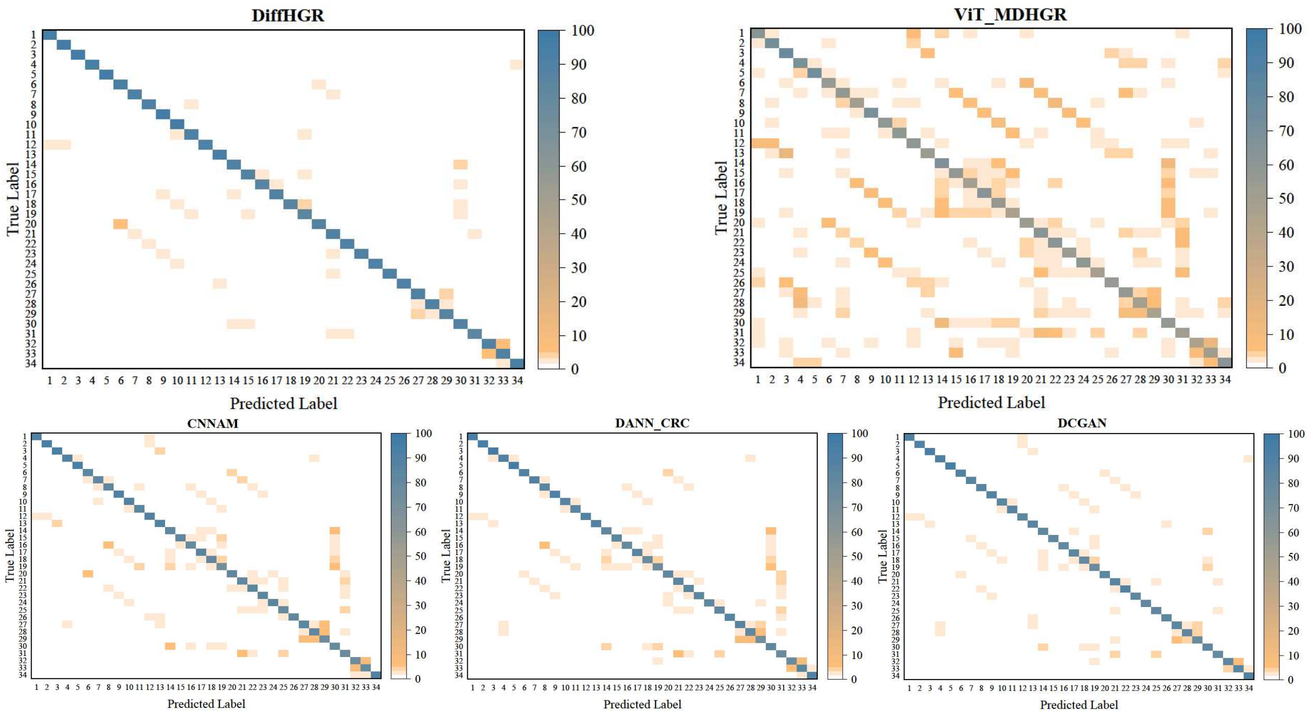


Figure 7: Confusion matrix comparison across five methods after two-trial calibration on the Hyser dataset. The horizontal axis indicates the predicted gesture label, and the vertical axis indicates the true label (34 gesture classes in total). Each heatmap cell shows the classification accuracy for a given (true, predicted) label pair.

used for calibration and testing. Specifically, the intersession experiments consist of pre-training and calibration stages. The intrasession experiments in session 1 are regarded as the pre-training stage. Empirically, we observe diminishing returns after two calibration trials, and 2-trial calibration yields performance close to the intrasession (see Section 3.1.3). Thus, the calibration phase involves three types of calibration experiments for each subject: 0-trial, 1-trial, and 2-trial calibration. In 0-trial calibration, the pre-trained model is directly tested with session 2 data. For 1-trial calibration, the model is first calibrated with one calibration trial and then tested on session 2 data. Similarly, 2-trial calibration uses two calibration trials before testing on session 2 data. All calibration experiments are implemented on individuals.

During 0 trial calibration experiments, pre-trained models achieve an average accuracy of 39.15% ± 16.82% across all subjects. As presented in Figure 6, though there are noticeable declines in performance compared to the intrasession results, the proposed method outperforms almost all the benchmark models. When calibrating the pre-trained models on 1 trial or 2 trials data of session 2, the proposed method achieves the average accuracies of 84.87% ± 3.53% and 90.27% ± 2.75% across 20 subjects, respectively, while CNNAM achieves 76.31% ± 6.03% and 83.38% ± 4.94%, ViT-MDHGR achieves 56.09% ± 15.13% and 58.15% ± 14.08%, DANN_CRC achieves 78.74% ± 4.96% and 84.87% ± 4.64%, and DCGAN achieves 80.22% ± 3.72% and 84.33% ± 4.03%.

Calibration with just one trial from session 2 leads to a significant improvement in accuracy, increasing by approximately 15.98% (S9) to 74.07% (S20) across subjects compared to the pre-trained model. The 2-trial calibration further enhances performance, which can almost match the intrasession performance, only with an average 3.04% accuracy gap. The results demonstrate that the model effectively adapts to session variability with few-shot additional data. Additionally, Figure 7 presents the confusion matrices for five different methods after two calibration trials. The confusion matrix provides a detailed view of how well each method distinguishes between the various gestures. As observed from Figure 7, the proposed DiffHGR method demonstrates a superior ability to accurately distinguish 34 gestures compared to the benchmark methods. The diagonal values, representing correct classifications, are consistently higher for DiffHGR, indicated by more intense blue blocks along the diagonal. In contrast, the off-diagonal areas, which indicate misclassifications, are notably lighter and contain fewer orange blocks in our method compared to the other methods. This reduction in orange blocks highlights a lower rate of misclassifications, meaning that our method is more effective in minimizing confusion between gestures.

3.1.3. *Minimum calibration trials for cross-day adaptation*

We further investigate the minimum calibration effort required for reliable cross-day performance by varying the number of labeled calibration trials collected from the target session (session 2) from 0 to 4 trials per gesture. As shown in

Figure 8, introducing only one calibration trial already yields a substantial improvement over the zero-shot setting, and increasing the budget to two trials further brings a significant performance gain ($p < 0.001$). Notably, the performance shows diminishing returns beyond two trials, as no statistically significant improvement is observed when increasing the calibration budget from two to three or four trials ("ns"). These results indicate diminishing returns beyond two trials and suggest that two short calibration trials are sufficient to effectively compensate for cross-day distribution drift while keeping the annotation burden low. Therefore, unless otherwise specified, we adopt the 2-trial calibration protocol in all subsequent experiments that involve target-day adaptation.

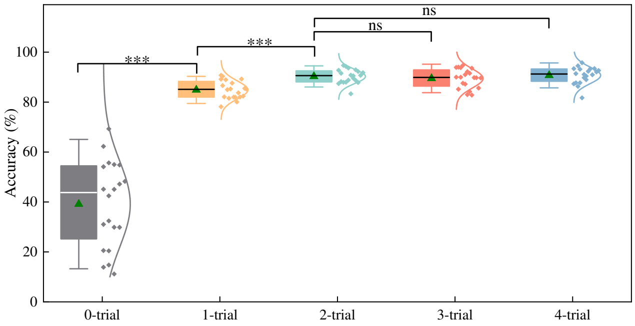


Figure 8: Cross-day accuracy under different calibration trials (from 0 to 4 trials per gesture). Statistical significance is assessed using paired t-tests across subjects ($*p < 0.05$, $**p < 0.01$, $***p < 0.001$, "ns" indicates no significant difference).

3.2. Cross-subject evaluation on Hyser dataset

We further investigate the generalization ability of our proposed DiffHGR framework in a cross-subject setting. Cross-subject evaluation presents a greater degree of variability due to individual differences in physiological structures, skin impedance, and electrode placement. Although our method is not explicitly optimized for inter-subject transfer, it is still important to examine whether the learned representations can adapt to unseen individuals with minimal calibration efforts. We conducted target evaluation experiments on six unseen subjects using a calibration setting of 0-trial, 1-trial, and 2-trial. As shown in Figure 9, in the 0-trial setting, the average classification accuracy across unseen subjects was significantly limited (e.g., only 15.19% for Subject 17). This highlights the considerable intersubject variability in sEMG signals. However, after incorporating just one labeled calibration trial, performance improved dramatically across all subjects, with accuracy exceeding 96% in all cases. A further gain was observed in the 2-trial scenario, achieving over 99% accuracy for most subjects. These results suggest that the proposed DiffHGR method enables the model to facilitate rapid adaptation to new subjects with limited supervision.

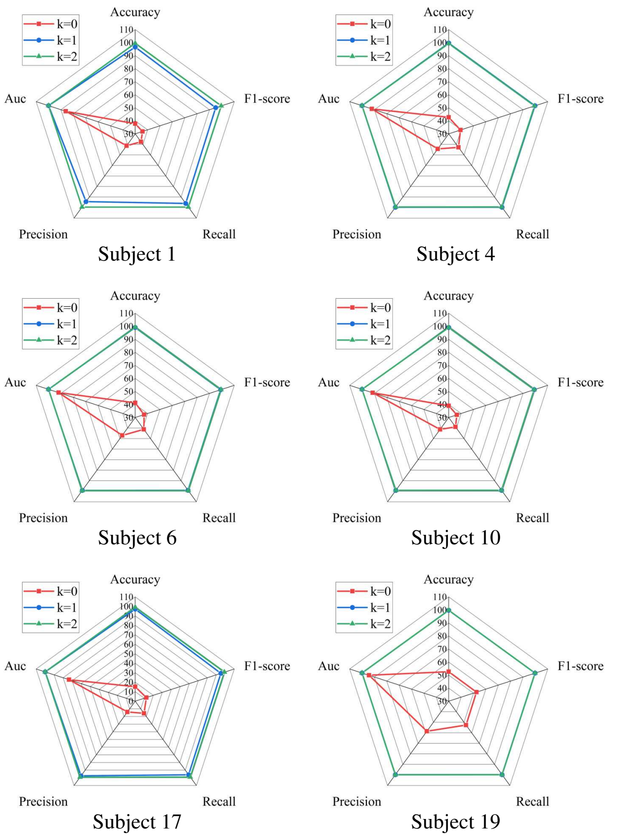


Figure 9: Radar charts of classification performance across six target subjects in the cross-subject experiments, evaluated with 0, 1, and 2 calibration trials (k = 0, 1, 2). Each radar chart summarizes five metrics: accuracy, F1-score, recall, precision, and AUC.

3.3. Evaluation on CapgMyo and CSL-HDEMG Datasets

In addition to the Hyser dataset, we further validate the proposed DiffHGR method on two other publicly available datasets, CapgMyo and CSL-HDEMG. Although they are not collected across different days, they still exhibit cross-trial variability. These datasets allow us to assess the performance of DiffHGR in various environments. CapgMyo dataset includes data from 18 subjects. For the evaluation, we use 7 trials for training and 3 trials for testing. As shown in Figure 10, DiffHGR consistently outperforms all other benchmark methods, achieving the highest average accuracy of approximately 96.13%. The data points in the plot represent the performance across different subjects, with DiffHGR showing the least variation, as evidenced by its narrow interquartile range (IQR). This indicates that DiffHGR is robust to trial-to-trial variability and performs stably across different subjects. The CSL-HDEMG dataset consists of data from 5 subjects, each with 5 sessions. For session 5 as an example, 7 trials are used for training and 3 trials for testing. As illustrated in Figure 10, DiffHGR again leads with an accuracy above 92%, outperforming the other methods in terms of both accuracy and consistency across

Table 3
Quantitative comparison of the IS and FID among different generative methods. Higher IS indicates better diversity and quality of the generated samples, while lower FID reflects closer alignment between the distribution of generated and real data.

	DANN_CRC	DCGAN	DiffHGR(0%)	DiffHGR(25%)	DiffHGR(50%)	DiffHGR(75%)	DiffHGR(100%)
IS↑	3.880	1.028	7.872	9.650	10.369	10.561	**10.974**
FID↓	95.796	3271.117	3.937	3.792	3.875	3.893	**3.982**

different subjects. DiffHGR's lower variation and tighter IQR suggest that it effectively handles cross-trial variability.

In both datasets, DiffHGR demonstrates superior performance, with a consistent accuracy range across subjects. Notably, the box plots show that DiffHGR yields fewer outliers and exhibits less variation in its performance compared to methods like DCGAN and ViT-MDHGR, which show more significant performance fluctuations. In addition, we conducted paired t-tests to statistically evaluate the differences between DiffHGR and the baseline methods. The results, annotated in Figure 10, indicate that DiffHGR significantly outperforms all other methods on the CapgMyo dataset and the CSL-HDEMG dataset. These results validate the robustness and effectiveness of the proposed DiffHGR method, not only in true cross-day scenarios (using the Hyser dataset) but also in other situations, such as those presented by CapgMyo and CSL-HDEMG. The ability of DiffHGR to achieve high and stable performance across these different datasets highlights its potential for generalizing to a wide range of hand gesture recognition tasks.

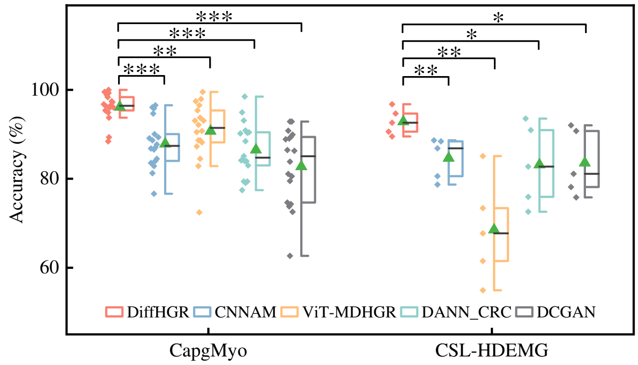
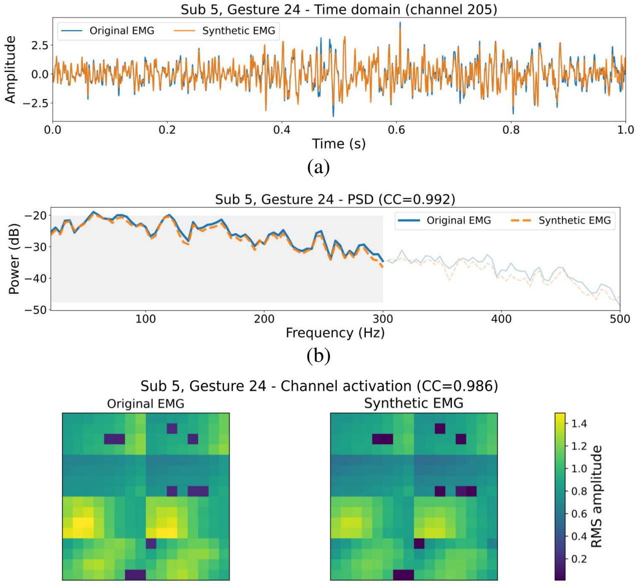


Figure 10: The evaluation comparison between DiffHGR and other benchmark methods on CapgMyo and CSL-HDEMG dataset. Statistical significance was assessed via paired t-tests between DiffHGR and each baseline method($*p < 0.05$, $**p < 0.01$, $***p < 0.001$).

3.4. Evaluation of the quality and diversity of the generated data

To further assess the quality and diversity of the generated data, we evaluate the Inception Score (IS) and Fréchet Inception Distance (FID). These metrics are used to evaluate how well the generated synthetic data represents the real HD-sEMG data in terms of its diversity and similarity to the true data distribution. We compare the DiffHGR method with two benchmark data augmentation methods: DANN_CRC and DCGAN. These methods use synthetic data generation as part of their training process to enhance model performance on gesture recognition tasks. In particular, DANN_CRC uses a domain-adversarial approach to augment training data for hand gesture recognition, while DCGAN generates synthetic data through adversarial training to enhance the variety. For a fair comparison, we evaluated different DiffHGR models using different ratios of synthetic data incorporation (0%, 25%, 50%, 75%, and 100%) for training classifiers. As shown in Table 3, DiffHGR consistently outperforms the benchmark methods in terms of IS. As the proportion of synthetic data used for training increases from 0% to 100%, the IS score improves from 7.872 to 10.974, which suggests that DiffHGR can progressively enrich the diversity of the training distribution without sacrificing sample quality. Regarding FID, DiffHGR attains consistently low values across all settings, which are orders of magnitude lower than those of DANN_CRC and DCGAN. In our setting, the same Diff component of DiffHGR is used to produce synthetic samples for all ratios. Consequently, the small numerical differences in FID across the 0-100% settings mainly reflect sampling randomness. The key observation is that DiffHGR maintains a FID of around 3.9 in all settings, indicating that the synthesized HD-sEMG signals remain very close and stably aligned to the real data distribution.

To further validate these quantitative metrics, we additionally visualize representative pairs of original and synthetic HD-sEMG samples generated by DiffHGR (see Figure 11). First, in the time domain, the synthetic waveform closely follows the envelope and fluctuation patterns of the original EMG for the same subject and gesture, while still exhibiting sample-wise variability rather than a simple copy. Second, the power spectral density (PSD) curves of real and synthetic signals almost overlap within the main 20–300 Hz band, yielding a high Pearson correlation coefficient (e.g., CC = 0.988 in the illustrated case), which indicates that the generator preserves the characteristic frequency content of muscle activity. Third, the channel-wise RMS activation maps show highly similar spatial activation patterns across the electrode grid (e.g., CC = 0.986 between real and synthetic maps), demonstrating that DiffHGR can reproduce the multi-channel spatial synergy structure of HD-sEMG signals. These time-domain, spectral, and spatial visualizations support that the synthesized signals are not artificial artifacts, but realistic variations that faithfully reflect the underlying real-data distribution.



(a)

(b)

(c)

Figure 11: Qualitative comparison between original and synthetic HD-sEMG signals generated by DiffHGR for Subject 5 and Gesture 24.

Furthermore, we perform qualitative visualization of the real and synthetic features using three popular dimensionality reduction techniques: Principal Component Analysis (PCA), t-distributed Stochastic Neighbor Embedding (t-SNE), and Uniform Manifold Approximation and Projection (UMAP). As shown in Figure 12, we present comparisons for two representative subjects from the Hyser dataset: Subject 1 (left column) and Subject 17 (right column). Across all three visualization methods, we observe a high degree of overlap between real and generated features. PCA captures global variance structures and reveals that the synthetic features are distributed in a manner similar to the real data, without mode collapse or excessive concentration. The t-SNE and UMAP projections, which emphasize local neighborhood structures and nonlinear manifold relationships, further demonstrate that the generated data are well interleaved with the real data in feature space. These results suggest that the proposed diffusion-based generative module successfully preserves both global structure and local similarity patterns of HD-sEMG signals. Such visual consistency provides qualitative support for the effectiveness of our generation process.

3.5. Ablation and sensitivity studies

3.5.1. Component-wise ablation of Diff and AE

To disentangle the contribution of each component in DiffHGR, we conducted a comprehensive ablation study under the intrasession experimental setting. Specifically, we evaluated four model configurations:

- **Baseline:** Both Diff and AE components are removed. A simple encoder followed by a linear classifier is trained on raw HD-sEMG data without any augmentation or reconstruction.

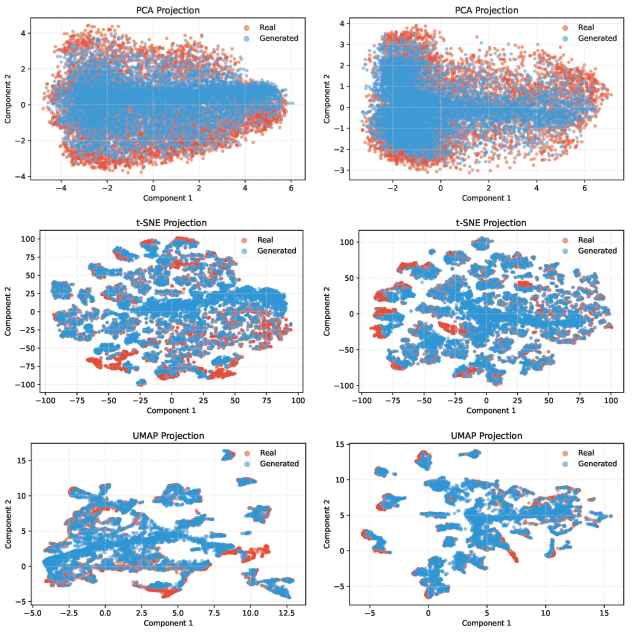


Figure 12: Feature visualizations of real and generated signals from the Hyser dataset using PCA, t-SNE, and UMAP. The left column corresponds to Subject 1, and the right column corresponds to Subject 17.

- **Diff-only:** The diffusion module is used to generate synthetic training data, but the AE component is excluded. Classification is performed by the same baseline encoder and classifier.

- **AE-only:** The model is trained only on real data, but the AE encoder–decoder structure is retained, including skip connections from the diffusion encoder.

- **DiffHGR:** Both Diff and AE components are used. The model is trained on a mixture of real and synthetic data, and the AE decoder benefits from skip connections of the Diff encoder to enhance representation learning.

The results, presented in Table 4, demonstrate that both components independently contribute to performance improvements. As shown, the AE component enhances the learning of discriminative representation through reconstruction and skip-connected features, leading to a substantial accuracy gain from 77.60% to 91.37% over the baseline. Meanwhile, the Diff component alone improves the generalization of the model by enhancing synthetic data, achieving 89.42% accuracy. When both modules are jointly integrated, the full DiffHGR model yields the best accuracy of 93.6%, confirming their complementary roles. In addition to the overall accuracy improvements, we also observe a clear reduction in performance variance when both components are enabled. Specifically, the standard deviation of accuracy drops from 21.92% in the Diff-only configuration and 6.95% in the AE-only configuration to only 2.3% in the full DiffHGR model. This suggests that DiffHGR not only improves accuracy but also enhances model stability across

Table 4
Ablation study evaluating the individual contributions of the Diff component and AE component in the intrasession experiments. We report the mean and standard deviation of four evaluation metrics (Accuracy, F1-score, Recall, and Precision) across all subjects.

Methods	Diff component	AE component	Accuracy	F1-score	Recall	Precision
Baseline	×	×	77.60% ± 7.38%	75.25% ± 8.23%	75.42% ± 7.52%	77.68% ± 7.63%
Diff-only	✓	×	89.42% ± 21.92%	87.97% ± 24.99%	88.11% ± 24.45%	89.49% ± 16.47%
AE-only	×	✓	91.37% ± 6.95%	90.20% ± 12.19%	90.00% ± 11.40%	90.40% ± 10.13%
DiffHGR	✓	✓	93.6% ± 2.3%	91.41% ± 3.45%	91.52% ± 3.4%	91.95% ± 3.26%

subjects. The higher variance in the Diff-only setting reflects that although synthetic data enhances generalization, the lack of reconstruction guidance from the AE module may lead to inconsistent representations. Conversely, the AE-only configuration offers more stable yet slightly less robust performance due to the absence of data augmentation. Their combination allows the model to benefit from both stable encoding and diverse training distributions, leading to the best trade-off between performance and consistency.

3.5.2. *Effect of diffusion-based data generation on cross-day robustness*

To further clarify the role of diffusion-based data augmentation, we perform an additional comparison between DiffHGR and the Baseline model defined in Section 3.5.1. The Baseline model is obtained by removing the Diff and AE components from DiffHGR, and retaining only the AE encoder together with the classifier used in Stage C. In other words, while DiffHGR trains an encoder–classifier using both real and synthetic HD-sEMG signals after jointly training the Diff and AE components, the Baseline employs the same encoder-classifier backbone but is trained on session 1 with real HD-sEMG data only, without any diffusion-based data generation and augmentation. We evaluate both models in a cross-day setting, where session 1 and session 2 are recorded on different days.

Figure 13 summarizes the cross-day accuracies of Baseline and DiffHGR under 0-trial, 1-trial, and 2-trial calibration, where the calibration trials are from session 2 recordings acquired on a different day than session 1. Bars indicate the mean and standard deviation across 20 subjects, and individual points show subject-wise results. Under the 0-trial condition, DiffHGR already achieves higher cross-day accuracy than Baseline, indicating that diffusion-based augmentation in session 1 improves the robustness of the learned representations rather than degrading them. With 1-trial and 2-trial calibration, both methods benefit from a small amount of labeled session 2 data. DiffHGR consistently outperforms Baseline under the same annotation cost (all $p < 0.001$). Moreover, the subject-wise data points and error bars in Figure 13 clearly show that DiffHGR not only improves the mean accuracy but also reduces inter-subject variability, leading to more stable cross-day performance. These findings suggest that diffusion-based generation is not intended to exhaustively model every future daily variation, but to regularize the source domain training, while a few-shot calibration phase efficiently aligns the model to the actual target-day distribution.

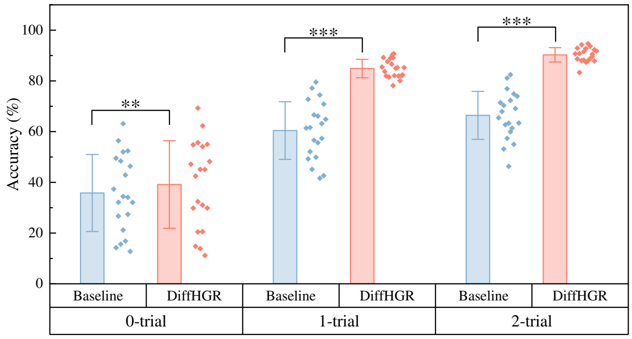


Figure 13: Comparison of cross-day recognition accuracies between the real-only baseline and the proposed DiffHGR under 0-trial, 1-trial, and 2-trial calibration settings. Statistical significance is assessed by paired t-tests ($*p < 0.05$, $**p < 0.01$, $***p < 0.001$).

3.5.3. *Performance comparison under different window size*

Recent studies confirmed that larger windows capture more detailed information. However, they introduce longer latency in myoelectric control. Given the requirement for myoelectric pattern recognition systems to operate with a response time of less than 300 ms for real-time application [35; 36], we aim to determine the shortest window size that can maintain high intersession HGR performance. This paper investigates the effect of window sizes on intersession performance. As shown in Figure 14, the 62.5 ms window consistently outperforms the 31.25 ms window across all calibration settings (0-trial, 1-trial, and 2-trial), with statistically significant differences ($p < 0.05$). Compared to the 125 ms window, the performance difference of 62.5 ms is not statistically significant, indicating comparable effectiveness. Although the 250 ms window achieves significantly better results than the 125 ms window in some settings, the latter incurs a much higher latency, which limits its suitability for real-time applications. Overall, the 62.5 ms window achieves a favorable trade-off between performance and latency, making it the most practical choice for our experiments.

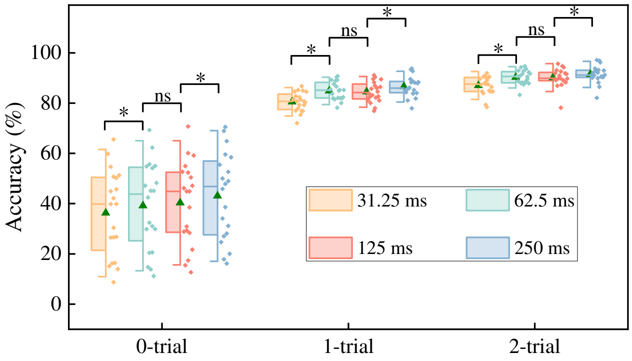


Figure 14: The impact of varying window sizes on intersession performance evaluated across 20 subjects. Statistical significance was assessed via paired t-tests between different window settings ($*p < 0.05$, $**p < 0.01$, $***p < 0.001$, "ns" indicates no significant difference).

3.6. Computational efficiency analysis

To quantitatively assess the computational cost of the proposed DiffHGR framework, we compare its training and calibration efficiency with two generative baselines, DANN_CRC and DCGAN. The results are summarized in Table 5. All experiments in this study were conducted on a workstation equipped with an Intel Xeon Gold 6226R, 256 GB RAM, and two NVIDIA GeForce RTX 3090 GPUs (each with 24 GB VRAM), using PyTorch 2.4.1 and CUDA 12.4.

3.6.1. Offline training cost

During offline training, DiffHGR exhibits a favourable trade-off between wall-clock time, model size, and arithmetic complexity. As shown in Table 5, the total training time of DiffHGR is 55.8 ± 7.1 minutes, which is only slightly longer than DCGAN, but substantially shorter than DANN_CRC. In addition, DiffHGR maintains the smallest model size (1.46 M trainable parameters) among the three methods, compared with 2.57 M for DANN_CRC and 15.04 M for DCGAN. In terms of computational complexity, DiffHGR requires 42.82 M MACs per sample during training, which is lower than DANN_CRC and more than one order of magnitude lower than DCGAN (1891.8 M). The much higher MACs of DCGAN mainly stem from the adversarial training pipeline. In each training iteration, both the generator and the discriminator are executed and updated, and they are implemented as several deep convolutional and transposed-convolution blocks operating on high-resolution HD-sEMG maps. Even though a lightweight MobileNet backbone is adopted for classification, the cost of repeatedly forwarding real and synthetic samples through the generator–discriminator pair dominates the overall complexity, leading to MACs that are approximately 40-fold higher than that of DiffHGR. By contrast, the reverse diffusion process in DiffHGR operates each sample once through the Diff component in a single-step denoising manner, leading to much lower training MACs. These results indicate that the diffusion-based framework introduces only a moderate increase in offline training time relative to DCGAN, while achieving a much more compact model and significantly reduced training MACs compared with GAN-based augmentation.

3.6.2. Calibration cost and inference times

In the cross-day or cross-subject settings, all methods perform a few-shot calibration before use. For both DANN_CRC and DCGAN, we follow a conservative calibration protocol and only fine-tune the final linear classification layer, while keeping all feature extractors fixed. This choice keeps the number of trainable parameters during calibration very small and favors these baselines in terms of calibration time. As for DiffHGR, we first update the encoder of the AE component while the Diff component is frozen and used only to synthesize generated samples for data augmentation. Then, the classifier is fine-tuned using a mix of real and generated synthetic data. This enables the latent representation and the decision boundary to better align with the target-day distribution under the same few-shot budget. As summarized in Table 5, on Hyser dataset, the full DiffHGR model contains 1.46 M parameters, while only 0.25 M parameters are trainable in calibration, and the average calibration time is 6.66 ± 0.95 min. On XDHDEMG dataset, reducing the HD-sEMG channel count from 256 to 64 further lowers the total parameters to 0.09 M, with only 0.016 M trainable during calibration. Moreover, after calibration the per-sample inference latency is 0.451 ± 0.023 ms and 0.373 ± 0.034 ms, which is well within the sliding step of 10 ms. Finally, we further corroborate the practical deployability of DiffHGR via real-time validation on XDHDEMG (Section 3.7). Under the same data preprocessing setup as offline, the end-to-end latency from window acquisition to prediction is approximately 132 ms, and the online accuracy remains consistent with the corresponding offline evaluation, supporting the feasibility of DiffHGR in real-world HGR applications.

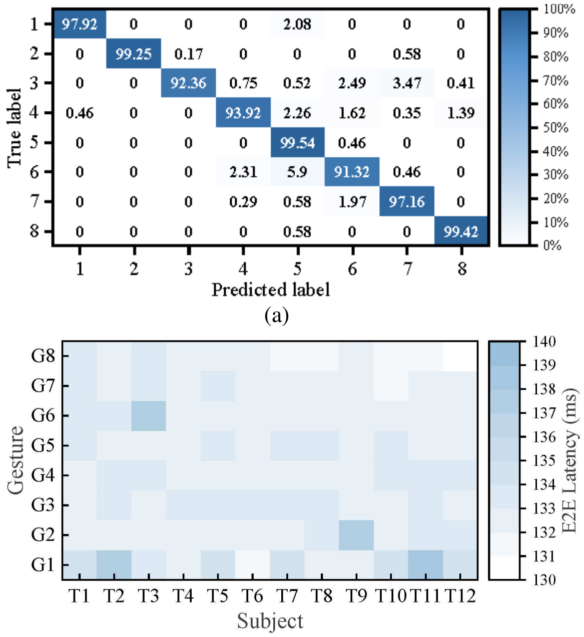
3.7. Real-time inference validation

To further evaluate the practical ability of the proposed DiffHGR framework in real-world deployment, we conducted real-time inference validation on our self-collected XDHDEMG dataset under both intraday and interday conditions. Twelve healthy subjects were recruited to perform eight hand gestures, each repeated six times as a session, two sessions were executed. For each subject, data collected on session 1 were used to pretrain the DiffHGR model offline. In the intraday scenario, real-time inference experiments were conducted approximately 30 minutes later on the same day. The pretrained AE encoder and classifier was directly used for online gesture prediction. In the interday scenario, each subject returned after a delay of 3-7 days to participate in the interday validation session. Given the distributional drift typically induced by factors such as electrode repositioning, skin impedance changes, and muscle fatigue, a lightweight calibration procedure was applied before online inference. Specifically, two calibration trials per gesture were collected on the new day. During calibration, we first update the

Table 5
Summary of offline training and calibration efficiency for DiffHGR on the Hyser PR and XDHDEMG datasets.

Dataset	Phase	Method	Time (min)	Params (M)	MACs / sample (M)	Inference time / sample (ms)
Hyser	Offline training	DANN_CRC	398.9 ± 96.1	2.57	59.03	–
		DCGAN	48.1 ± 4.6	15.04	1891.8	–
		DiffHGR	55.8 ± 7.1	1.46	42.82	–
	Calibration	DANN_CRC	1.23 ± 0.23	1.12	59.03	0.016 ± 0.002
		DCGAN	5.03 ± 0.98	0.17	16.11	0.065 ± 0.001
		DiffHGR	6.66 ± 0.95	0.25	0.115	0.451 ± 0.023
XDHDEMG	Offline training	DiffHGR	20.59 ± 2.58	0.09	2.75	–
	Calibration	DiffHGR	4.00 ± 0.13	0.016	0.019	0.373 ± 0.034

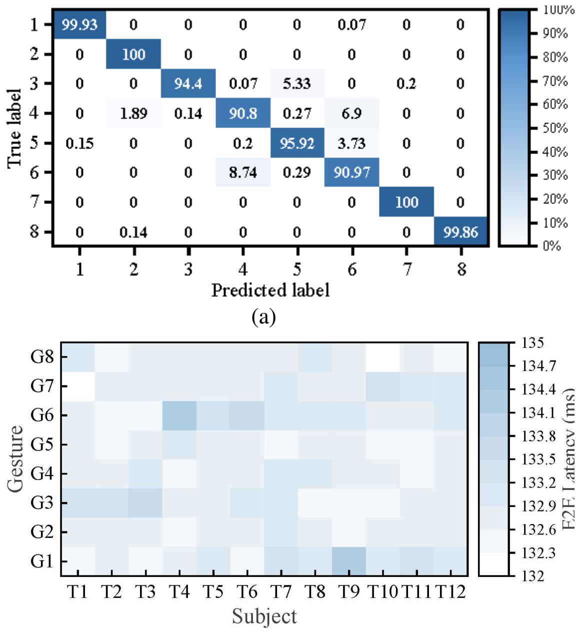
encoder of the AE component while the Diff component is frozen and used only to synthesize generated samples for data augmentation. Then, the classifier is fine-tuned using a mix of real and generated synthetic data. After calibration, real-time prediction was performed using only the fine-tuned AE encoder and classifier, whereas the diffusion-based Diff component remained inactive in the online inference path. The deployed model is lightweight, with only 0.016 M trainable parameters involved in calibration, which supports low-latency deployment. During the online experiments, we adopted the same post-processing strategy as our previous work [37] to ensure consistent decision smoothing.



(a)

(b)

Figure 15: Real-time inference performance of the proposed system evaluated during intraday conditions. (a) Aggregated confusion matrix of gesture classification in the real-time setting. (b) Distribution of average end-to-end latency across all subject–gesture pairs.



(a)

(b)

Figure 16: Real-time inference performance of the proposed system evaluated during interday conditions. (a) Aggregated confusion matrix of gesture classification in the real-time setting. (b) Distribution of average end-to-end latency across all subject–gesture pairs.

The real-time prediction performance of the proposed DiffHGR framework was evaluated under both intraday and interday conditions, as shown in Figure 15 and Figure. 16, respectively. The confusion matrices (top) illustrate the per-class classification performance, while the gesture-subject heatmaps (bottom) visualize the distribution of end-to-end (E2E) latency across different gestures and subjects. Under the intraday condition (Figure 15), the DiffHGR model demonstrated consistently high gesture recognition accuracy of 96.36% ± 9.78% and maintained a low average E2E latency of 132.83 ms ± 0.34 ms, indicating stable and efficient real-time inference without the need for recalibration. In the interday setting (Figure 16), although slight distributional shifts emerged due to electrode repositioning and physiological variability, the model maintained strong performance, with 96.48% ± 14.42% classification accuracy, after rapid adaptation without retraining the entire model. This minimal supervision proved sufficient to compensate for cross-day signal drift. The average E2E latency increased only marginally to 132.97 ms, with a slightly higher standard deviation (± 1.19 ms), reflecting increased temporal variability across different days. These results collectively demonstrate the suitability of DiffHGR for real-time applications.

4. Discussion

4.1. Challenges in practical applications

For EMG-based hand gesture interfaces to be clinically and practically usable, robustness to distribution shift is often more critical than peak performance under a single controlled session. In real-world deployment, performance degradation is mainly driven by two related but distinct sources of shift: interday drift and intersubject variability. Interday drift arises from electrode repositioning and attachment inconsistencies, fluctuations at the skin-electrode interface (e.g., impedance changes due to perspiration and temperature), and changes in muscle state and recruitment patterns across days. Intersubject variability is caused by physiological and anatomical differences (e.g., muscle morphology and activation strategies) as well as individual-specific execution styles, which lead to notable changes in both signal statistics and discriminative patterns. These factors are further compounded by wearability constraints (comfort, attachment stability, and long-term usability), making it difficult to maintain consistent acquisition conditions outside laboratory settings. Consequently, EMG interfaces face an inherent requirement that models must remain reliable under non-stationary and population-dependent conditions rather than assuming stationary signal statistics.

4.2. Validity of synthetic augmentation

To address the above shifts, DiffHGR adopts a unified deployment-oriented strategy consisting of offline generative augmentation training and fast calibration adaptation, which together improve robustness under both cross-day and cross-user conditions. A key concern is that synthetic augmentation is helpful only if it preserves task-relevant EMG structure and does not drift toward artifact-dominated or task-irrelevant modes. For this reason, the diffusion generator (Diff component) is not trained in isolation. Instead, it is jointly optimized with an auxiliary autoencoder (AE component) under explicit reconstruction supervision, and multi-scale structural cues learned during diffusion encoding are reused via skip-connected feature transfer to the reconstruction pathway. This joint optimization constrains the synthetic distribution while encouraging the model to retain gesture-discriminative patterns, thereby reducing reliance on day-specific or subject-specific nuisance factors. The efficiency of this design is supported by our evaluation of the quality and diversity of the generated data (Section 3.4) as well as component-wise ablations (Section 3.5), which together indicate that performance gains are attributable to trustworthy augmentation rather than uncontrolled distribution drift.

4.3. Cross-day and cross-subject generalization

Building upon the offline-trained model, DiffHGR further introduces a lightweight few-shot calibration procedure to efficiently align the model to the actual target condition (a new day or a new user). This design explicitly frames calibration as a controllable trade-off between reliability and user burden. While fully calibration-free transfer across different users and long-term use conditions remains challenging in myoelectric recognition, a small amount of target data can substantially improve robustness when the adaptation is designed to be efficient. In our evaluation, the calibration overhead is modest (6.66 ± 0.95 min on Hyser in Table 5) yet yields consistent improvements under cross-day protocols compared with the real-only Baseline model under the same calibration trials (Section 3.5.2, Figure 13). Moreover, the reduced performance dispersion across subjects suggests that the proposed method improves not only mean accuracy but also stability across individuals, which is particularly relevant for reliable assistive and rehabilitative use. Collectively, a series of experiments including cross-day and cross-subject evaluations, minimal calibration trials analyses, ablation studies, and evaluation of quality and diversity collectively demonstrate that the proposed DiffHGR effectively mitigates practical distribution shift.

4.4. Practical deployability and real-time feasibility

Beyond accuracy under offline benchmarks, a deployable EMG-based interface must satisfy the practical requirements of low computational and calibration overhead to support real-time interaction. The proposed offline generative augmentation training and fast calibration adaptation strategy could keep the user burden and runtime cost manageable. The diffusion model is exploited offline to learn a robust recognition model via generative augmentation, and it is jointly trained with an auxiliary autoencoder (AE) under reconstruction supervision. In particular, the AE leverages multi-scale structural cues transferred from the Diff encoder (via skip connections) to preserve gesture-discriminative structures and constrain the synthetic distribution, thus promoting a more stable feature representation for recognition. In contrast, real-time deployment relies only on the AE encoder and classifier, i.e., the Diff module is not involved during online inference. As a result, the deployed model remains compact and computation-efficient, with a small parameters and low MACs (Table 5), and the measured inference latency per sample is correspondingly low, meeting the timing constraints of interactive applications. In addition, the calibration procedure is lightweight and can be completed within a modest time budget (Table 5), enabling quick adaptation to a new day or a new user without extensive re-collection or long retraining cycles. To this end, we explicitly evaluated DiffHGR in real-time settings under both intraday and interday conditions, demonstrating that the proposed framework maintains reliable online recognition ability (Section 3.7).

4.5. Limitations and future works

Several limitations should be acknowledged. First, although DiffHGR has been evaluated on multiple HD-sEMG datasets with different subject counts, gesture sets, channel configurations, and recording protocols (Section 3.1-Section 3.3 and Section 3.7), and we provide various evaluations of the quality and diversity of the generated data (Section 3.4), the generalization across arbitrary datasets remain bounded by the diversity of available training data. Future work will focus on validating the proposed framework on more diverse cross-day datasets, which are collected under broader participant demographics and recording conditions, to better characterize generalization under real-world day-to-day shifts and to further assess robustness across more diverse participants. Second, our current formulation primarily captures variability at a statistical level, explicitly modeling richer physiological priors (e.g., muscle synergy constraints) may further improve robustness in unconstrained usage. Third, beyond the cross-day setting, we also evaluate the performance in a cross-subject setting (Section 3.2). In this protocol, the model is trained on all source subjects and then adapted to an unseen target subject using only two labeled calibration trials, after which it achieves acceptable recognition performance. While this few-shot personalization is practical for myoelectric interfaces, further reducing the calibration burden remains important. Future work will investigate more efficient inter-subject adaptation, e.g., by integrating DiffHGR with domain adaptation or meta-learning to improve transfer to unseen users with fewer (or even no) labeled calibration samples. In addition, we will investigate online/continual learning strategies that can update the model under long-term drift using lightweight incremental optimization, confidence-aware pseudo-labeling, and drift-triggered calibration, while mitigating catastrophic forgetting through regularization or replay mechanisms. Finally, practical wearable control often benefits from complementary sensing. We will extend the framework toward multimodal control by fusing HD-sEMG with other biosignals such as EEG (for capturing cortical intent under weak EMG conditions), using modality-aware fusion or reliability-weighted decision schemes. Such multimodal extensions may support more stable, user-friendly control in real-world assistive applications.

5. Conclusion

This paper proposes DiffHGR, a diffusion-based framework for robust EMG-based hand gesture recognition. DiffHGR comprises a diffusion-based generator (Diff) and an auxiliary autoencoder (AE). Diff is used to synthesize diverse samples for data augmentation, while the AE helps preserve task-relevant structure and prevents uncontrolled drift. To address cross-day shifts issue, we employ a lightweight few-shot calibration. Specifically, Diff is kept frozen and used only to generate augmentation samples, and the AE encoder and classifier are updated for fast adaptation. Comprehensive experiments validate the effectiveness, demonstrating that DiffHGR achieves a cross-day average accuracy of 90.27% for recognizing 34 gestures across 20 subjects, using calibration on only two trials per gesture, significantly surpassing other benchmark methods. In addition, we reported offline training, calibration, and inference efficiency metrics, and validated DiffHGR in real-time intraday and interday settings. During real-time prediction, only the calibrated AE encoder and classifier are executed, with Diff remaining inactive. Experimental results show that the DiffHGR demonstrated consistently high gesture recognition accuracy of around 96% and maintained a low average end-to-end latency of around 132 ms for recognizing 8 gestures across 12 subjects.

Despite the above results, inter-day or inter-subject evaluation still relies on few-shot labeled calibration for fast personalization, and further reducing or eliminating this labeling requirement (e.g., via domain adaptation or meta-learning) remains an important direction. Moreover, while real-time feasibility is validated under intraday and interday experiments with reported efficiency indicators, real-time evaluation under unseen users (inter-subject) remains to be further investigated.

Acknowledgment

All authors would like to thank the support of the Key Research and Development Program of Shaanxi (ProgramNo. 2024GX-ZDCYL-02-15), and Natural Science Basic Research Program of Shaanxi (ProgramNo.2025SYS-SYSZD-079).

References

[1] J. O. d. O. de Souza, M. D. Bloedow, F. C. Rubo, R. M. de Figueiredo, G. Pessin, S. J. Rigo, Investigation of different approaches to real-time control of prosthetic hands with electromyography signals, IEEE Sens. J. 21 (18) (2021) 20674–20684. doi:10.1109/JSEN.2021.3099744.

[2] C. V. Anikwe, H. F. Nweke, A. C. Ikegwu, C. A. Egwuonwu, F. U. Onu, U. R. Alo, Y. W. Teh, Mobile and wearable sensors for data-driven health monitoring system: State-of-the-art and future prospect, Expert Syst. Appl. 202 (2022) 117362. doi:https://doi.org/10.1016/j.eswa.2022.117362.

[3] T.-Y. Pan, W.-L. Tsai, C.-Y. Chang, C.-W. Yeh, M.-C. Hu, A hierarchical hand gesture recognition framework for sports referee training-based emg and accelerometer sensors, IEEE Trans. Cybern. 52 (5) (2020) 3172–3183. doi:10.1109/TCYB.2020.3007173.

[4] U. Côté-Allard, G. Gagnon-Turcotte, A. Phinyomark, K. Glette, E. Scheme, F. Laviolette, B. Gosselin, A transferable adaptive domain adversarial neural network for virtual reality augmented emg-based gesture recognition, IEEE Trans. Neural Syst. Rehabil. Eng. 29 (2021) 546–555. doi:10.1109/TNSRE.2021.3059741.

[5] Y. Tang, M. Pan, H. Li, X. Cao, A convolutional-transformer based approach for dynamic gesture recognition of data gloves, IEEE Trans. Instrum. Meas. 73 (2518813) (2024) 1–13. doi:10.1109/TIM.2024.3400361.

[6] J. Qi, L. Ma, Z. Cui, Y. Yu, Computer vision-based hand gesture recognition for human-robot interaction: a review, Complex Intell. Syst. 10 (1) (2024) 1581–1606. doi:10.1007/s40747-023-01173-6.

[7] S. Ni, M. A. Al-qaness, A. Hawbani, D. Al-Alimi, M. Abd Elaziz, A. A. Ewees, A survey on hand gesture recognition based on surface electromyography: Fundamentals, methods, applications, challenges and future trends, Appl. Soft Comput. (2024) 112235doi:https://doi.org/10.1016/j.asoc.2024.112235.

[8] Q. Hu, G. A. Azar, A. Fletcher, S. Rangan, S. F. Atashzar, Vit-mdhgr: Cross-day reliability and agility in dynamic hand gesture prediction via hd-semg signal decoding, IEEE J. Sel. Top. Signal Process. 18 (3) (2024) 419–430. doi:10.1109/JSTSP.2024.3402340.

[9] Y. Zhao, S. Jing, H. Wu, H. Li, M. Todoh, E-trgan: A novel transformer generative adversarial network for high-density surface electromyography signal reconstruction, IEEE Trans. Instrum. Meas. 73 (4011013) (2024) 1–13. doi:10.1109/TIM.2024.3472778.

[10] L. Wu, A. Liu, X. Zhang, X. Chen, X. Chen, Electrode shift robust cnn for high-density myoelectric pattern recognition control, IEEE Trans. Instrum. Meas. 71 (2518010) (2022) 1–10. doi:10.1109/TIM.2022.3204996.

[11] M. J. Islam, S. Ahmad, F. Haque, M. B. I. Reaz, M. A. S. Bhuiyan, M. R. Islam, Application of min-max normalization on subject-invariant emg pattern recognition, IEEE Trans. Instrum. Meas. 71 (2521612) (2022) 1–12. doi:10.1109/TIM.2022.3220286.

[12] Y. Okawa, S. Kanoga, T. Hoshino, T. Nitta, Sequential learning on semgs in short-and long-term situations via self-training semi-supervised support vector machine, in: IEEE Eng. Med. Biol. Soc. Annu. Conf., IEEE, 2022, pp. 3183–3186. doi:10.1109/EMBC48229.2022.9871311.

[13] C. Shen, Z. Pei, W. Chen, J. Wang, X. Wu, J. Chen, Lower limb activity recognition based on semg using stacked weighted random forest, IEEE Trans. Neural Syst. Rehabil. Eng. 32 (2024) 166–177. doi:10.1109/TNSRE.2023.3346462.

[14] W. Batayneh, E. Abdulhay, M. Alothman, Comparing the efficiency of artificial neural networks in semg-based simultaneous and continuous estimation of hand kinematics, Digit. Commun. Netw. 8 (2) (2022) 162–173. doi:https://doi.org/10.1016/j.dcan.2021.08.002.

[15] N. Jiang, D. Farina, Myoelectric control of upper limb prosthesis: current status, challenges and recent advances, Front. Neuroeng. 7 (4) (2014) 7–9. doi:10.3389/conf.fneng.2014.11.00004.

[16] X. Jiang, X. Liu, J. Fan, X. Ye, C. Dai, E. A. Clancy, D. Farina, W. Chen, Optimization of hd-semg-based cross-day hand gesture classification by optimal feature extraction and data augmentation, IEEE Trans. Hum.-Mach. Syst. 52 (6) (2022) 1281–1291. doi:10.1109/THMS.2022.3175408.

[17] Z. Yang, D. Jiang, Y. Sun, B. Tao, X. Tong, G. Jiang, M. Xu, J. Yun, Y. Liu, B. Chen, et al., Dynamic gesture recognition using surface emg signals based on multi-stream residual network, Front. Bioeng. Biotechnol. 9 (2021) 779353. doi:https://doi.org/10.3389/fbioe.2021.779353.

[18] N. K. Karnam, S. R. Dubey, A. C. Turlapaty, B. Gokaraju, Emghandnet: A hybrid cnn and bi-lstm architecture for hand activity classification using surface emg signals, Biocybern. Biomed. Eng. 42 (1) (2022) 325–340. doi:https://doi.org/10.1016/j.bbe.2022.02.005.

[19] Z. Zhang, Q. Shen, Y. Wang, Electromyographic hand gesture recognition using convolutional neural network with multi-attention, Biomed. Signal Process. Control 91 (2024) 105935. doi:https://doi.org/10.1016/j.bspc.2023.105935.

[20] M. Montazerin, S. Zabihi, E. Rahimian, A. Mohammadi, F. Naderkhani, Vit-hgr: vision transformer-based hand gesture recognition from high density surface emg signals, in: IEEE Eng. Med. Biol. Soc. Annu. Conf., IEEE, 2022, pp. 5115–5119. doi:10.1109/EMBC48229.2022.9871489.

[21] X. Wang, D. Ao, L. Li, Robust myoelectric pattern recognition methods for reducing users' calibration burden: challenges and future, Front. Bioeng. Biotechnol. 12 (2024). doi:https://doi.org/10.3389/fbioe.2024.1329209.

[22] U. Côté-Allard, C. L. Fall, A. Drouin, A. Campeau-Lecours, C. Gosselin, K. Glette, F. Laviolette, B. Gosselin, Deep learning for electromyographic hand gesture signal classification using transfer learning, IEEE Trans. Neural Syst. Rehabil. Eng. 27 (4) (2019) 760–771. doi:10.1109/TNSRE.2019.2896269.

[23] D. Wu, J. Yang, M. Sawan, Transfer learning on electromyography (emg) tasks: Approaches and beyond, IEEE Trans. Neural Syst. Rehabil. Eng. 31 (2023) 3015–3034. doi:10.1109/TNSRE.2023.3295453.

[24] X. Chen, Y. Li, R. Hu, X. Zhang, X. Chen, Hand gesture recognition based on surface electromyography using convolutional neural network with transfer learning method, IEEE J. Biomed. Health Inform. 25 (4) (2020) 1292–1304. doi:10.1109/JBHI.2020.3009383.

[25] Y. Wang, P. Zhao, Z. Zhang, A deep learning approach using attention mechanism and transfer learning for electromyographic hand gesture estimation, Expert Syst. Appl. 234 (2023) 121055. doi:https://doi.org/10.1016/j.eswa.2023.121055.

[26] Z. Chen, Y. Qian, Y. Wang, Y. Fang, Deep convolutional generative adversarial network-based emg data enhancement for hand motion classification, Front. Bioeng. Biotechnol. 10 (2022) 909653. doi:https://doi.org/10.3389/fbioe.2022.909653.

[27] Y. Shi, S. Ma, Y. Zhao, C. Shi, Z. Zhang, A physics-informed low-shot adversarial learning for semg-based estimation of muscle force and joint kinematics, IEEE J. Biomed. Health Inform. 28 (3) (2024) 1309–1320. doi:10.1109/JBHI.2023.3347672.

[28] D. Lee, D. You, G. Cho, H. Lee, E. Shin, T. Choi, S. Kim, S. Lee, W. Nam, Emg-based hand gesture classifier robust to daily variation: Recursive domain adversarial neural network with data synthesis, Biol. Signal Process. Control 88 (2024) 105600. doi:https://doi.org/10.1016/j.bspc.2023.105600.

[29] Z. Lin, P. Liang, X. Zhang, Z. Qin, Toward robust high-density emg pattern recognition using generative adversarial network and convolutional neural network, in: IEEE/EMBS Neural Eng. Conf., IEEE, 2023, pp. 1–5. doi:10.1109/NER52421.2023.10123910.

[30] J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, Adv. Neural Inf. Process. Syst. 33 (2020) 6840–6851.

[31] H. Li, G. Ditzler, J. Roveda, A. Li, Descod-ecg: Deep score-based diffusion model for ecg baseline wander and noise removal, IEEE J. Biomed. Health Inform. (2023) 1–11doi:10.1109/JBHI.2023.3237712.

[32] X. Jiang, X. Liu, J. Fan, X. Ye, C. Dai, E. A. Clancy, M. Akay, W. Chen, Open access dataset, toolbox and benchmark processing results of high-density surface electromyogram recordings, IEEE Trans. Neural Syst. Rehabil. Eng. 29 (2021) 1035–1046. doi:10.1109/TNSRE.2021.3082551.

[33] W. Geng, Y. Du, W. Jin, W. Wei, Y. Hu, J. Li, Gesture recognition by instantaneous surface emg images, Scientific reports 6 (1) (2016) 36571. doi:10.1038/srep36571.

[34] C. Amma, T. Krings, J. Böer, T. Schultz, Advancing muscle-computer interfaces with high-density electromyography, in: Proc. ACM Hum.-Comput. Interact., 2015, pp. 929–938. doi:10.1145/2702123.2702501.

[35] B. Hudgins, P. Parker, R. N. Scott, A new strategy for multifunction myoelectric control, IEEE J. Biomed. Eng. 40 (1) (1993) 82–94. doi:10.1109/10.204774.

[36] R. Hu, X. Chen, H. Zhang, X. Zhang, X. Chen, A novel myoelectric control scheme supporting synchronous gesture recognition and muscle force estimation, IEEE Trans. Neural Syst. Rehabil. Eng. 30 (2022) 1127–1137. doi:10.1109/TNSRE.2022.3166764.

[37] K. Su, K. Liu, B. Wan, H. Qiao, J. Huang, M. Feng, J. Liu, Multisource adversarial feature disentanglement method for cross-subject gesture recognition using semg signals, IEEE Trans. Instrum. Meas. 74 (2025) 1–12. doi:10.1109/TIM.2025.3557823.