Comparative Learning for Cross-Subject Finger Movement Recognition in Three Arm Postures via Data Glove

Lei Jiang, Fengmeng Zeng, and Annie Yu

***Abstract*— Reliable recognition of therapeutic hand and finger movements is a prerequisite for effective home-based rehabilitation, where patients must exercise without continuous therapist supervision. Inter-subject variability, stemming from differences in hand size, joint flexibility, and movement speed limit the generalization of data-glove models. We present CLAPISA, a contrastive-learning framework that embeds a Siamese network into a CNN–LSTM spatiotemporal pipeline for cross-subject gesture recognition. Training employs a 1: 2 positive-to-negative pairing strategy and an empirically optimized margin of 1.0, enabling the network to form subject-invariant, rehabilitation-relevant embeddings. Evaluated on a bending-sensor dataset containing twenty young adults, CLAPISA attains an average accuracy of 96.71 % under leave-one-subject-out cross-validation outperforming five baseline models and reducing errors for the most challenging subjects by up to 12.3 %. Although current validation is limited to a young cohort, the framework's data efficiency and subject-invariant design indicate strong potential for extension to elderly and neurologically impaired populations, our next work will be to collect such data for further verification.**

***Index Terms*— Siamese network, comparative learning, cross-subject, data glove, finger movement recognition.**

I. Introduction

THE human hand plays a central role in executing fine motor tasks such as eating, dressing, and operating electronic devices. With population ageing and the growing prevalence of neurological disorders—including stroke and Parkinson's disease—the number of individuals suffering hand-motor dysfunction is rapidly rising, seriously undermining independence and quality of life [1]. Although in-clinic hand-rehabilitation programmers improve motor function and facilitate recovery [2], they depend heavily on therapist supervision, imposing high labour costs and limiting continuous, individualized training after discharge. Home-based rehabilitation systems therefore offer a promising alternative, allowing patients to exercise independently while being monitored remotely [3], [4], [5]. A cornerstone of such systems is reliable, calibration-free recognition of hand and finger movements, which enables progress assessment, automated feedback, and adaptive therapy.

Among sensing technologies, data gloves with flexible bending sensors provide an attractive compromise between accuracy, wearability, and cost. They capture joint-level motion precisely while remaining comfortable for long-term use by older adults. Previous research has trailed many other sensing approaches, camera tracking [6], electromyography (EMG), force myography (FMG), inertial measurement units (IMUs) [7], [8], [9] and multimodal fusion can raise accuracy in complex conditions [10]. Yet, gesture-recognition models still struggle to generalize across subjects. Physiological and biomechanical differences, hand size, muscle tone, joint flexibility, movement speed, introduce large inter-subject signal variations and degrade performance [11], [12]. Because extensive subject-specific calibration is impractical in-home settings, developing robust, calibration-free models remains essential for scalable rehabilitation technology. However, data gloves themselves are not without drawbacks, bulkier designs can impede natural movement, and emerging alternatives such as acoustic, vibratory, and optical sensors [13], [14], [15], [16], [17] are still in validation and not yet ready for everyday home use. Given ease-of-use requirements for elderly users, this study focuses on flexible-sensor data gloves and on algorithms that tolerate inter-subject variability.

To achieve anatomically informed monitoring, sensors must be positioned at the three primary joints of every finger—metacarpophalangeal (MCP), proximal interphalangeal (PIP), and distal interphalangeal (DIP) (Fig. 1a). Because the motions of these joints are highly coupled and often nonlinear [18], [19], [20], measurements from all three locations are required to capture subtle, gesture-specific dynamics (Fig. 1b). Consequently, our gesture-recognition system fuses MCP, PIP, and DIP signals for fine-grained analysis. Hand-gesture recognition algorithms fall into two broad categories: static-posture

Received 20 March 2025; revised 13 June 2025; accepted 23 June 2025. Date of publication 26 June 2025; date of current version 2 July 2025. This work was supported in part by the Key Laboratory of Intelligent Textile and Flexible Interconnection, Zhejiang Province under Grant YB16, in part by China Postdoctoral Science Foundation under Grant 2024M750518, and in part by the Natural Science Foundation of Ningbo under Grant 2024J235 and Grant 2022J138. *(Corresponding author: Annie Yu.)*

Lei Jiang is with the Laboratory of Intelligent Home Appliances, College of Science and Technology, Ningbo University, Ningbo 315300, China, and also with Fudan Institute on Ageing, Fudan University, Shanghai 200433, China (e-mail: jianglei2@nbu.edu.cn).

Fengmeng Zeng is with the Key Laboratory of Intelligent Textile and Flexible Interconnection of Zhejiang Province, Hangzhou 310018, China (e-mail: zfmeng@zstu.edu.cn).

Annie Yu is with the School of Fashion and Textiles, The Hong Kong Polytechnic University, Hong Kong, China (e-mail: annie.tw.yu@polyu.edu.hk).

Digital Object Identifier 10.1109/TNSRE.2025.3583303

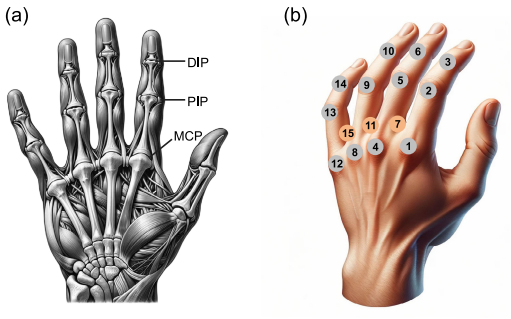

Fig. 1. Overview of the hand. a. Finger knuckles. b. Suggested positions of the finger sensors.

classification and dynamic-trajectory recognition. Classical pipelines rely on handcrafted features [9], [21], [22], [23] and usually attain only moderate accuracy because they model shallow patterns.

Modern deep-learning approaches learn features end-to-end: CNNs capture spatial sensor relationships [24], RNNs/LSTMs model temporal dynamics [25], [26], and hybrid or attention-based architectures further boost performance [27], [28], [29], [30], [31], [32]. Nevertheless, physiological and kinematic variability, hand size, muscle strength, joint flexibility, movement speed, remains a major obstacle to cross-subject generalization [33], [34]. Recent adaptive strategies (e.g. multi-dataset pre-training [34], ALS-SVM [35], and transfer-learning CNNs with attention [36]) improve subject-specific accuracy, yet substantial room remains to cut calibration effort and increase robustness for unseen users.

To address the above challenges, we propose a novel neural network architecture named Contrastive Learning for Arm Posture and Intersubject Alignment (CLAPISA), which combines Siamese networks with contrastive learning to significantly enhance cross-subject generalization capabilities in gesture recognition. Compared to more complex contrastive learning frameworks [37], the Siamese structure offers advantages such as straightforward implementation, reduced data requirements, and robust feature consistency, making CLAPISA particularly suitable for deployment in home-based rehabilitation scenarios with limited resources. The main contributions of this study are as follows:

(1) A two-stage spatiotemporal feature extraction module is developed within each branch of the Siamese network, where convolutional layers extract spatial features and an LSTM module captures the temporal dynamics of hand movements. By incorporating contrastive learning, CLAPISA explicitly pulls intra-class samples closer and pushes inter-class samples farther apart in the embedding space. This leads to significantly improved feature compactness (intra-class distance: 2.26) and separability (inter-class distance: 15.77) compared to baseline models.

(2) Extensive cross-subject experiments using leave-one-subject-out cross-validation (LOSOCV) demonstrate the effectiveness of CLAPISA. The model achieves an average classification accuracy of 96.71%, with individual accuracies ranging from 91.57% to 100%. Notably, for hard-to-classify subjects such as S02, S05, S13, and S19, CLAPISA outperforms the CNN–LSTM baseline by a substantial margin (up to +12.31%), reflecting its strong robustness and reduced performance variance across subjects.

(3) CLAPISA exhibits superior capability in distinguishing kinematically similar gestures, such as G03, G04, and G07, with significantly lower misclassification rates than baseline models. This confirms the model's effectiveness in capturing fine-grained differences in hand movement patterns, which is critical for practical deployment in rehabilitation monitoring and gesture-controlled interfaces.

In summary, CLAPISA provides an integrated, data-efficient, and highly generalizable framework by leveraging the complementary strengths of Siamese networks and contrastive learning for multisensor gesture recognition. The remainder of this paper is organized as follows: Section II reviews related work, Section III introduces the proposed method, Section IV presents experimental evaluations of CLAPISA, Section V discusses the findings, and Section VI concludes the paper.

II. Related Work

A. General Flow of Finger Movement Recognition

Finger movement recognition using data gloves has attracted significant research interest, driven by advancements in flexible bending sensor technology [38]. The recognition process generally comprises four key stages: data recording, data preprocessing, feature extraction, and classification modeling. Each stage addresses critical challenges, such as inter-subject variability and environmental noise, to ensure accurate and robust recognition.

1) Data Recording: The experimental setup involves defining essential parameters, including the type of data glove, sensor placement, sampling rate, trial duration, repetitions, and finger movement categories. Participant-related factors, such as sample size, demographic diversity, and hand dominance, are carefully controlled to ensure the representativeness of the data. Standardized instructions and controlled conditions, such as consistent lighting and minimal noise, minimize variability and enhance data reproducibility.

2) Data Preprocessing: Raw sensor data often exhibit variability owing to differences in physiology, sensor placement, and movement patterns. Normalization techniques, such as Z-score scaling, standardize sensor outputs across subjects, while noise reduction methods, including low-pass filtering and wavelet denoising, suppress artifacts and improve signal fidelity. These preprocessing steps ensure the quality and comparability of data for subsequent analysis.

3) Feature Extraction: Feature extraction involves multidimensional analysis to derive discriminative patterns. Time-domain features, such as mean, variance, and skewness, characterize temporal dynamics. Frequency-domain features, including spectral entropy and power spectral density, capture repetitive motion characteristics. Spatial-domain features, such as joint trajectories and inter-joint angles, provide insights into biomechanics and coordination. By integrating these domain-specific features, a comprehensive representation is achieved, supporting robust recognition and generalization.

4) Classification Model: Machine learning models classify the extracted features into predefined categories. Subject-dependent models achieve high accuracy in personalized tasks,

whereas subject-independent models are designed for generalization across subjects. Recent advancements, such as transfer learning and domain adaptation, have been utilized to bridge the gap between these approaches, enabling broader and more adaptable applications.

B. Contrastive Learning

Contrastive learning, as a self-supervised learning paradigm, effectively extracts meaningful representations by contrasting positive and negative sample pairs [39]. This mechanism has been widely adopted across various fields, demonstrating exceptional performance in fine-grained image recognition, text classification, and cross-modal analysis. For instance, instance-based contrastive learning distinguishes individual objects or images, making it ideal for tasks requiring detailed differentiation [40]. Class-based contrastive learning, on the other hand, focuses on separating data by class labels, which is particularly advantageous in label-scarce scenarios [41]. In medical imaging, global-local contrastive learning has been employed to improve lesion detection by contrasting holistic and localized features [42].

In the domain of hand movement and gesture recognition, contrastive learning has shown significant promise in feature extraction from large-scale datasets. For example, Lai et al. [43] pre-trained hand gesture models on surface EMG (sEMG) data using contrastive learning, followed by fine-tuning and domain adaptation with limited labeled samples. Similarly, Dai et al. [44] utilized contrastive learning to integrate Wireless Fidelity and video data, facilitating complementary feature extraction across modalities and enhancing cross-target gesture recognition.

The proposed method in this study generates self-supervised labels by aligning inter-subject data through feature alignment, allowing the model to learn shared flexion patterns reflected in joint trajectories and movement dynamics across individuals and postures. This approach highlights the potential of contrastive learning to enhance generalization across subjects, hand motion types, and environmental variations, offering a robust framework for advancing hand motion recognition in small-sample and diverse datasets.

III. Methods

A. Data Materials

The dataset provided by Hu et al. [45] was employed in this study. Table I summarizes the specifications of the sensor glove dataset, which was collected in real-time using a CyberGlove II. This device records the bending angles of 15 finger joints during various hand movements, as illustrated in Fig. 1(b). The dataset includes 20 participants performing 12 distinct single- and multi-finger flexion and extension gestures across three arm postures. Each gesture was repeated three times at both fast and slow speeds, with each repetition lasting 10 seconds and recorded at a sampling rate of 20 Hz (Fig. 2).

B. Dataset Preprocessing

Individual differences in factors such as sex, age, hand size, muscle strength, and joint flexibility introduce variability in

TABLE I
SUMMARY OF EXPERIMENT DATASET

Participants	11 males, 9 females (24± 2 years), right-handed
Arm Postures	P01: Elbow flexed at 90°, palm parallel to the ground P02: Elbow flexed at 90°, palm perpendicular to the ground P03: Elbow flexed at 45°, palm perpendicular to the ground
Trials	Six trials (three fast and three slow)
Time	10 s × 20 Hz (sample rate)
Gestures	One-finger flexion: G01–G04 Two-finger flexion: G05–G07 Three-finger flexion: G08–G09 Four-finger flexion: G10 Finger abduction and adduction: G11 and G12
Dataset	20 subjects × 3 arm postures × 6 trails × 10 s × 20 sample point/s × 12 gestures × 15 sensors

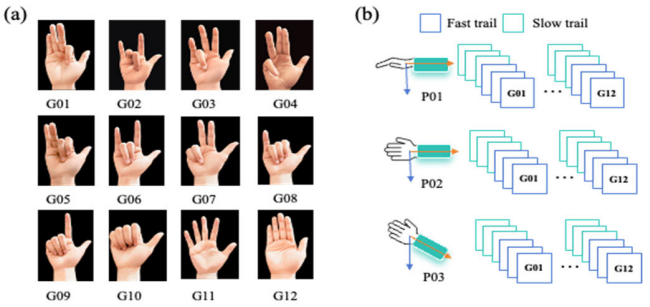

Fig. 2. Experimental procedure and data acquisition. a. Designed gestures for single- and multi-finger movements. b. Finger movement data recording process in three arm positions.

finger movement recognition benchmarks. To address these discrepancies and ensure data consistency, we applied zero-mean normalization across each sensor axis for all samples:

$$X_{i,j}^{k*} = \frac{X_{i,j}^{k} - \bar{X^k}}{\sigma^C}. \tag{1}$$

Here, $X_{i,j}^{k}$ represents the raw data from the k-th sensor, where $i = 1, 2, \ldots, Time$ denotes the time index, and $j = 1, 2, \ldots, Point$ signifies the sampling index within each second. $\bar{X^k}$ and σ^C are the mean and standard deviation of the k-th sensor, respectively.

This normalization ensures that each sensor's processed data has a mean of 0 and a standard deviation of 1, which is critical for maintaining consistency and comparability across subjects. By mitigating the impact of individual differences and sensor discrepancies, this process enhances the robustness of gesture-recognition models. Furthermore, normalization minimizes the influence of sensor-specific noise, enabling the model to focus on gesture-specific features rather than device-dependent variations. This step is particularly important for multisensory data to ensure model generalizability and accuracy.

C. CLAPISA Framework

The CLAPISA framework for cross-subject finger-movement recognition under three arm postures (Fig. 3) adopts a two-stage design: Stage I employs a Siamese network and contrastive loss to learn a subject-invariant embedding, and Stage II keeps this encoder fixed while

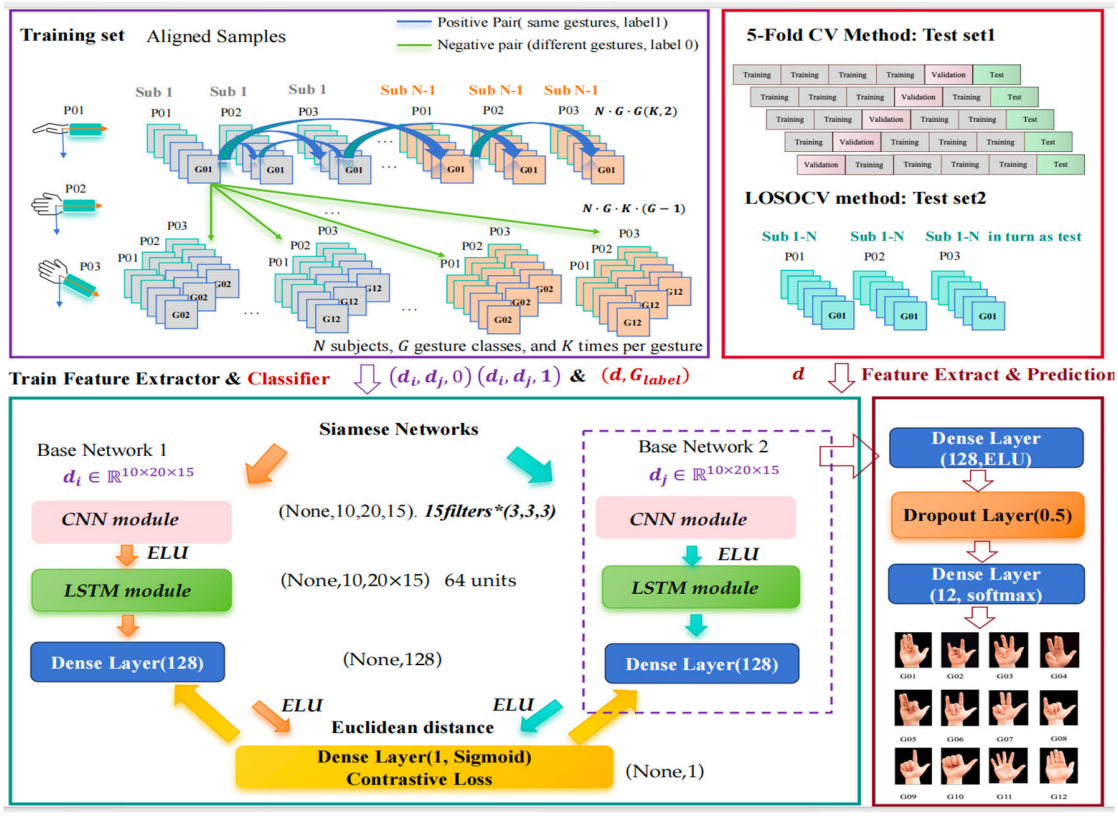

Fig. 3. Proposed CLAPISA method for cross-subject finger movement recognition in three arm postures.

training a lightweight classifier that maps single-sample embeddings to gesture labels.

1) Contrastive Learning for Feature Extractor:

a) Data preparation and pair creation: The dataset $\boldsymbol{D}$ consists of individual data points $\boldsymbol{d_i} \in \boldsymbol{D}$, each representing a gesture instance. Each data point is structured as a three-dimensional (3D) array, where the first dimension corresponds to the gesture sequence duration (10s), the second dimension represents the 20 sampling points per second, and the third dimension corresponds to the 15 sensor channels. Formally, each data point is expressed as $\boldsymbol{d_i} \in \mathbb{R}^{10\times20\times15}$. The gesture label set is defined as $\boldsymbol{L} = \{1, 2, \ldots, 12\}$, and for each label $l \in \boldsymbol{L}$, an index set $\mathbf{I_l}$ contains all gesture instances with label l.

Positive Pairs: For each $\mathbf{I}_l$, two distinct samples $\boldsymbol{d_i}, \boldsymbol{d_j}$ (where $i, j \in \mathbf{I_l}$ and $i \neq j$) are randomly selected to form a positive pair $(\boldsymbol{d_i}, d_j, 1)$, represented as:

$$Positive\ Pairs = \left\{ \left(\boldsymbol{d_i}, \boldsymbol{d_j}, 1\right) | i, j \in \mathbf{I_l}\ , i \neq j, \forall l \in \boldsymbol{L} \right\}. \tag{2}$$

Negative Pairs: For each index $i \in \mathbf{I}_l$, a different label $m \in \boldsymbol{L}(m \neq l)$ is selected, and a sample index $j \in \mathbf{I_m}$ is randomly chosen to form a negative pair $(\boldsymbol{d_i}, \boldsymbol{d_j}, 0)$, represented as:

$$\begin{aligned} &Negative\ Pairs \\ &= \left\{ \left(\boldsymbol{d_i}, \boldsymbol{d_j}, 0\right) | i \in \mathbf{I_l}\ , j \in \mathbf{I_m}, l \neq m, \forall l, m \in \boldsymbol{L} \right\}. \end{aligned} \tag{3}$$

For N subjects, G gesture classes, and K times per gesture, the number of positive pairs is $N \cdot G \cdot C(K, 2)$, and the number of negative pairs is $N \cdot G \cdot K \cdot (G - 1)$. In this study, we retain ordered pairs (d_i, d_j) and (d_j, d_i) to increase gradient diversity.

b) Siamese network: The Siamese network comprises two identical subnetworks, sharing the same architecture and parameters. The goal is to project gesture samples into a feature space where positive pairs are close together, and negative pairs are far apart.

The network processes the input gesture data $\boldsymbol{d_i}, \boldsymbol{d_j} \in \mathbb{R}^{10\times20\times15}$, and each subnetwork extracts features to compare the input pairs using a contrastive loss function. The specific details are as follows:

Step 1: CNN Module—Spatial Feature Extraction

The CNN module is responsible for extracting spatial features from the input data. The convolutional layers apply filters to the input data and learn spatial patterns within the gesture sequences. The output of the convolutional layer is computed as:

$$\boldsymbol{X_{conv}} = ELU\left(\boldsymbol{W}_{conv} \cdot \boldsymbol{d} + b_{conv}\right). \tag{4}$$

where W_{conv} and b_{conv} are the weights and biases of the convolutional layer. The exponential linear unit (ELU) activation function is applied to improve gradient propagation and mitigate the vanishing gradient issue. The ELU function is expressed as:

$$f(x) = \begin{cases} e^x - 1, & x < 0 \\ x, & x \geq 0. \end{cases} \tag{5}$$

The ELU activation function helps suppress irrelevant data by transforming negative values while enhancing feature representation.

Step 2: LSTM Module—Temporal Feature Capture

After spatial feature extraction, the LSTM module captures long-term dependencies in the gesture data. This module captures the temporal dynamics of the gesture sequences, which is crucial for accurate recognition. The operation is defined as:

$$h_t = LSTM(\boldsymbol{X_{conv}}, h_{t-1}), \tag{6}$$

where h_t is the hidden state at time t, and h_{t-1} is the hidden state from the previous time step.

Step 3: Fully Connected Layer—Feature Representation

To refine the extracted spatial and temporal features, a fully connected layer is applied

$$f = ELU\left(\boldsymbol{W_{dense}} h_t + b_{dense}\right). \tag{7}$$

where W_{dense} and b_{dense} are the weights and biases of the fully connected layer. The ELU activation preserves nonlinear characteristics while mitigating gradient vanishing issues.

Step 4: Distance Evaluation

The feature representations f_i and f_j from the two subnetworks are compared using the Manhattan distance (L1 distance). The L1 distance between two feature vectors f_i and f_j is calculated as:

$$d\left(f_i, f_j\right) = \sum \left|f_i - f_j\right|. \tag{8}$$

This distance measures the sum of absolute differences between corresponding components of the two vectors, reflecting their dissimilarity. During training, positive pairs (similar gestures) are optimized to minimize this distance, while negative pairs (dissimilar gestures) are encouraged to have larger distances.

In this framework, L1 distance is preferred over Euclidean distance (L2 distance) for the following reasons: (1) L1 distance is less sensitive to outliers compared to L2 distance. In gesture recognition tasks, where certain data points may be noisy or contain outliers (e.g., sensor noise), L1 distance helps prevent these outliers from disproportionately affecting the model's performance; (2) L1 distance is often preferred in high-dimensional spaces, as it tends to be more efficient for certain types of features (e.g., sparse or categorical data). In this case, the gesture data is high-dimensional, and L1 distance has shown to yield better performance in previous studies for similar tasks; (3) Based on preliminary experiments, L1 distance was found to outperform L2 distance in terms of classification accuracy and robustness to noise, making it a more suitable choice for this specific application.

c) Contrastive loss: The contrastive loss function is designed to encourage similar samples to be closer together while pushing dissimilar samples farther apart in the feature space. The contrastive loss function is defined as:

$$L\left(y_{true}, y_{pred}\right) = mean(y_{true} \cdot y_{pred}^2 + (1 - y_{true}) \cdot \max(margin - y_{pred}, 0)^2). \tag{9}$$

where $y_{true} \in \{0, 1\}$ indicates whether the pair is similar (1) or dissimilar (0). y_{pred} is the predicted distance between the pair. The $margin$ is a hyperparameter that defines the threshold for dissimilar samples.

For similar pairs ($y_{true} = 1$), the loss reduces as y_{pred} decreases, minimizing the distance between the feature representations. For dissimilar pairs ($y_{true} = 0$), the loss penalizes the model if y_{pred} is less than the $margin$, encouraging the network to increase the separation between dissimilar samples.

In summary, the Siamese network in the CLAPISA framework effectively combines CNN and LSTM modules to process multisensory gesture data. The CNN module extracts spatial features, while the LSTM module captures temporal dependencies. The feature representations are then compared using Manhattan distance, and the network is trained with a contrastive loss function. This approach ensures robust and accurate gesture recognition by optimizing the model to distinguish between similar gestures in the feature space.

2) Classifier and Evaluation Procedure: The CLAPISA framework is evaluated through a structured two-stage training and validation pipeline, combining subject-invariant representation learning with classification performance assessment. This section introduces the classifier training strategy, evaluation protocols, and evaluation metrics.

a) Classifier and training strategy: The CLAPISA framework adopts a two-stage training strategy to decouple representation learning and classification. In Stage I representation learning, a Siamese encoder is trained using contrastive loss to learn subject-invariant and gesture-discriminative embeddings. The embedding space is optimized such that similar gestures are pulled closer together, while dissimilar ones are pushed apart; (2) In Stage II classifier training, once the encoder is trained, its weights are frozen. One branch of the Siamese network is used as a fixed feature extractor. Individual gesture samples are encoded into embeddings, which are then fed into a lightweight classifier composed of a fully connected layer, a dropout layer (to prevent overfitting), and a final SoftMax layer for 12-class gesture prediction. The classifier is trained using cross-entropy loss to assign class labels based on the extracted embeddings.

b) Evaluation protocols: To evaluate the framework's generalization ability and stability, two validation protocols are employed: (1) 5-Fold CV, used to assess robustness across random data splits. The dataset is divided into five subsets, each serving as the validation set once, while the remaining four are used for training. This process is repeated across 10 random seeds to ensure statistical consistency and stability; (2) LOSOCV, used to evaluate generalization to unseen subjects. In each iteration, data from one subject is held out for testing, while the model is trained on the remaining subjects.

c) Performance metrics and evaluation stages: To comprehensively evaluate CLAPISA, we adopt a two-stage evaluation strategy aligned with the two-stage training pipeline. Each stage focuses on different capabilities and uses tailored metrics.

Stage I Representation Quality Assessment (Feature Extractor Evaluation): this stage focuses on evaluating the learned embedding space generated by the Siamese network. To measure how well the network captures discriminative and compact representations, we construct ordered positive and negative pairs from test set and compute the following metrics.

Class-Intra Distance (Within-Class Distance). Measures the compactness of samples within the same class, where smaller values indicate better intra-class consistency.

$$D_{intra} = \frac{1}{|C|} \sum_{i,j \epsilon C, i \neq j} \left\| f_i - f_j \right\| \tag{10}$$

Class-Inter Distance (Between-Class Distance). Measures the separation between different gesture classes in the feature space.

$$D_{inter} = \frac{1}{|C_1|\,|C_2|} \sum_{i \epsilon C_1, j \epsilon C_2, i \neq j} \left\| f_i - f_j \right\| \tag{11}$$

Fisher Discriminant Ratio (FDR). This metric evaluates the separability of classes by comparing within-class variance to between-class variance. A higher FDR indicates better class separability.

$$FDR = \frac{S_{between}}{S_{within}} \tag{12}$$

where $S_{between}$ is the between-class scatter matrix and S_{within} is the within-class scatter matrix.

Stage II Classification Performance Assessment (Classifier Evaluation): in this stage, the encoder is fixed, and a SoftMax classifier is trained and evaluated on individual samples, simulating real-world deployment. The following methods are used.

Accuracy. The overall accuracy, representing the proportion of correctly classified samples across all fold or all subjects, given by:

$$Accuracy = \frac{\sum_{i=1}^{12} C_{(i,i)}}{\sum_{i=1}^{12} \sum_{j=1}^{12} C_{(i,j)}}. \tag{13}$$

Loss. The cross-entropy loss computed on the test set during classification, indicating classification confidence and convergence.

In summary, the classifier training and evaluation process ensures a comprehensive and robust assessment of the model's performance. By employing a combination of cross-validation methods (5-Fold and LOSOCV), class-intra and class-inter distances, and FDR, the framework is thoroughly evaluated for its generalization ability across subjects and its ability to discriminate between different gestures.

IV. Results and Analysis

A. Division and Data Preparation

To evaluate the CLAPISA framework, we adopted two validation strategies: 5-Fold CV and LOSOCV. Each strategy follows a distinct data partitioning and sample pair generation protocol.

1) 5-Fold Cross-Validation: In the 5-Fold CV setting, the full dataset of 4320 samples was first split into a training-validation set (3456 samples) and a test set (864 samples) in an 8:2 ratio. This split was performed under 10 different random seeds (2, 12, 22, 32, 42, 52, 62, 72, 82, 92), each time generating a new training-test partition to obtain ten independent experimental results. These repetitions serve to evaluate the stability and generalization of the model under varying data splits.

TABLE II
SUMMARY OF DATASET AND PAIR DISTRIBUTION (5-FOLD CV)

Dataset	Data Array	Label	Positive Pairs	Negative Pairs*
Train-Val	(3456,10, 20, 15)	(3456, 12)	991,872	1,983,744
Test	(864, 10, 20, 15)	(864, 12)	-	-

*Negative pairs are down-sampled to twice the positive-pair count.

TABLE III
SUMMARY OF DATASET AND PAIR DISTRIBUTION (LOSOCV)

Dataset	Data Array	Label	Positive Pairs	Negative Pairs*
Train-Val	(4104,10, 20, 15)	(4104, 12)	1,399,464	2,798,928
Test	(216, 10, 20, 15)	(216, 12)	-	-

*Negative pairs are down-sampled to twice the positive-pair count.

Positive and negative sample pairs were constructed on the entire 3456-sample training-validation set before applying K-fold partitioning. For each gesture category (288 samples), intra-class combinations were used to generate positive pairs based on the formula $C\,(288, 2) = 41,328$. Thus, across 12 gesture classes, we obtained 991,872 positive pairs. Negative pairs were generated by pairing each sample from one gesture class with all samples from the remaining 11 classes, with a theoretical number of 5,474,304 ($C\,(12, 2) \cdot 288^2$). To control for imbalance, we down-sampled negative pairs to maintain a 2:1 negative-to-positive ratio, yielding 1,983,744 negative pairs. These 2,975,616 total training pairs are then evenly partitioned into 5 folds for contrastive learning training and validation. For final evaluation, the fixed test set (864 samples) was used to input to the SoftMax classifier for final gesture recognition.

2) Leave-One-Subject-Out Cross-Validation (LOSOCV): In LOSOCV, each round selects one subject (216 samples) as the independent test set, while the remaining 19 subjects (4104 samples) form the training-validation set. A total of 20 rounds are performed to ensure every subject is used once for testing. On the 4104-sample training-validation set, we constructed 1,399,464 positive pairs ($C(342, 2) \cdot 12$) and 2,798,928 negative pairs (down-sampled from $C\,(12, 2) \cdot 342^2$) using the same intra- and inter-class strategy as in 5-Fold CV. During testing, only the 216 test samples from the held-out subject were used to evaluate the quality of the embedding space learned in Stage I, while the 216 test samples were input to the softmax classifier for final gesture recognition. A detailed summary of data partitions and corresponding pair distributions is provided in Tables II and III.

B. Proposed Framework and Comparative Results

As illustrated in Fig. 3, CLAPISA consists of two weight-sharing Siamese branches. Each branch follows the sequence Conv2D → TimeDistributed-Flatten → LSTM → Dense 1 embedding. The two embeddings are compared through an L1-distance layer followed by a sigmoid unit, and the network is pre-trained with a contrastive loss. After pre-training, one branch is frozen as the feature extractor. A classification head (Dense2 → Dropout → SoftMax) is appended and fine-tuned on single-sample inputs. The key hyperparameter settings are report in Table IV, which were

TABLE IV
HYPERPARAMETER SEARCH SPACE AND BEST VALUES

Hyperparameter	Search range	Best value
2D Conv-kernel	3×3, 5×5, 7×7	3×3
Filters number	15, 20, 30, 60	15
LSTM units	16, 32, 64, 128	64
Dense1 units	32, 64, 128, 256	128
Dropout	0.1-0.5	0.5
Dense2 units	32, 64,128,256	128
Learning rate	1×10^{-3}, 5×10^{-4}, 1×10^{-4}	1×10^{-4}
Batch size	8, 16, 32, 64	32

TABLE V
ARCHITECTURAL COMPOSITION OF THE SIX MODELS

Model	Spatial Module	Temporal Module	Siamese branches + Contrastive Loss
CNN	✓	×	×
LSTM	×	✓	×
CNN-LSTM	✓	✓	×
CL-CNN	✓	×	✓
CL-LSTM	×	✓	✓
CLAPISA	✓	✓	✓

tuned by Keras-Tuner RandomSearch under 5-fold cross-validation. The full search space is shown in Table IV. Other parameters, ELU activation, Adam optimizer and the final SoftMax-12 layer were fixed. The best setting, selected by the highest validation accuracy and lowest validation loss, employs a 3 × 3 kernel with 15 filters, 64 LSTM units, two 128-dimensional embedding layers, 0.5 dropout, learning-rate 1×10^{-4} and batch-size 32.

To evaluate the contribution of each module in CLAPISA, we conducted comparative experiments on six models (see Table V): CNN (spatial-only), LSTM (temporal-only), CNN–LSTM (spatio-temporal), CL-CNN, CL-LSTM and CLAPISA. No additional tuning was performed for the baselines, each baseline inherits layer-by-layer the optimal hyper-parameters listed above in Table IV. Thus, CNN (3 × 3 × 3 kernels (15 filters, ELU) → 2 Dense-128 → Dropout 0.5 → Dense-12); LSTM (LSTM-64 → 2 Dense-128 → Dropout 0.5 → Dense-12); CNN-LSTM (CNN → TimeDistributed-Flatten → LSTM → 2 Dense-128 → Dropout 0.5 → Dense-12). All models share the same learning-rate 1×10^{-4}, batch-size 32 and Adam optimizer. For CL variants, the twin-branch structure and contrastive loss are added, while main layers and all hyperparameters are identical to their baselines.

To further investigate the impact of the margin hyper-parameter in the contrastive loss function, we conducted additional experiments by varying the margin among {0.9, 1.0, 1.1} during the Siamese encoder training in the CLAPISA framework. The results show that a margin of 1.0 yields the best classification performance. A smaller margin (e.g., here 0.9) tends to cause insufficient separation between dissimilar pairs, thereby reducing inter-class distance and impairing class discriminability. Conversely, a larger margin (e.g., here 1.1) may over-penalize closely spaced dissimilar pairs, leading to increased intra-class variance and reduced feature compactness. Accordingly, margin of 1.0 was adopted as the default configuration for all CL-based models in subsequent experiments.

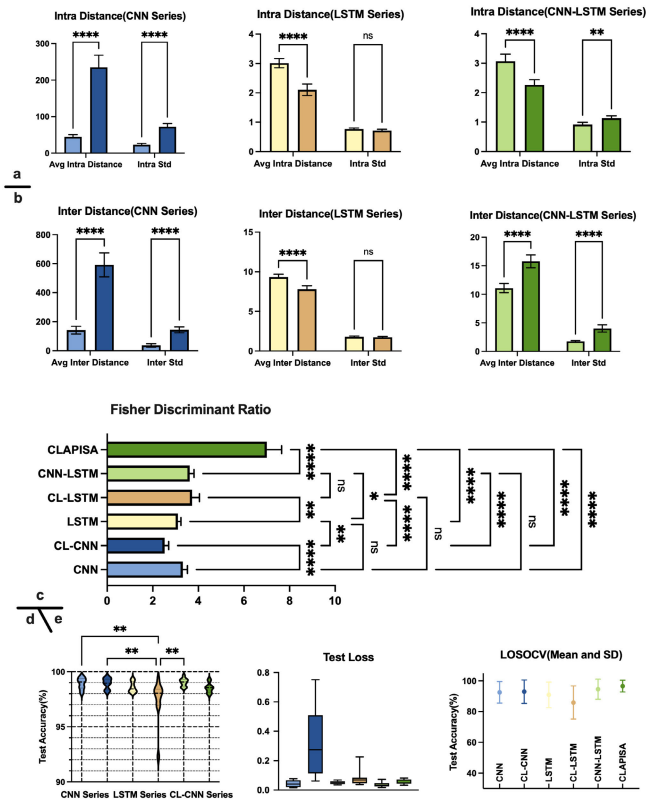

Fig. 4. Comparison of the proposed method with the baseline method under 5-fold CV and the LOSOCV. a. Intra-class distances of feature representations across six models. b. Inter-class distances of feature representations across six models. c. Comparison of Fisher discriminant ratio (FDR) among six models. d. Test accuracy and loss under 5-fold CV. e. Test accuracy under LOSOCV. *Statistical significance: $p<0.05$(*), $p<0.01$(**), $p<0.001$(***), and $p<0.0001$(****).*

All experiments were implemented using TensorFlow and Keras on an Apple M2 Max (64 GB RAM). Under each of 10 random seeds, we conducted 5-fold cross-validation on a 3456-sample training/validation set. For CL-based models, contrastive training was performed on 991,872 positive and 1,983,744 negative pairs. The model with the lowest validation loss in each seed-fold combination was selected and evaluated on a held-out 864-sample test set. Final results report seed-averaged metrics on both feature embedding and classification performance. We evaluated representational quality using average intra-class distance (lower is better for compactness), inter-class distance (higher is better for separability), and Fisher Discriminant Ratio (FDR), as shown in Fig. 4(a) and (b).

The CNN-only model showed poor intra-class compactness (44.71) and moderate inter-class separability (141.64), indicating that relying solely on spatial features was insufficient to effectively distinguish gestures. The LSTM-only model significantly improved feature compactness (intra-class distance = 3.01) but exhibited limited inter-class separation (9.33), highlighting that temporal feature alone, though beneficial, are insufficient for reliable gesture differentiation. The CNN–LSTM model further improved feature compactness (3.07) and separability (11.09), demonstrating the benefits of spatial–temporal feature integration. Introducing contrastive learning significantly increased inter-class distances, especially

in spatial features (CL-CNN: 591.85; CL-LSTM: 177.79), confirming that contrastive learning effectively enhances class separability. The complete CLAPISA framework achieved the best performance, exhibiting the lowest intra-class distance (2.26) and a high inter-class distance (15.77). This result highlights the synergistic effect of integrating spatiotemporal features with contrastive learning, significantly surpassing all other comparative models.

For classification performance evaluated using accuracy, loss, and FDR under the 5-Fold CV setting (Fig. 4 (c) and (d)), the CNN–LSTM baseline achieved the highest test accuracy (99.03%) and lowest test loss (0.038), indicating strong generalization ability. The proposed CLAPISA framework closely followed, achieving 98.53% accuracy and significantly lower test loss (0.060). Notably, although the accuracy of CLAPISA was slightly below CNN–LSTM, it showed the highest FDR (7.00), indicating superior feature separability and robustness in dealing with complex cross-subject gesture recognition tasks.

C. Cross-Subject Generalization Analysis

To further assess the generalization ability of the proposed CLAPISA framework and its ablated variants, we employed the LOSOCV method. The experimental results are depicted in Fig. 4(e). The CNN model achieved an accuracy of 92.55%, suggesting that while spatial features alone provide some degree of generalization, they are insufficient for handling cross-subject variability. The LSTM model reached 90.88% accuracy, indicating moderate classification performance but limited stability, suggesting that relying solely on temporal features hinders robust generalization. In contrast, the CNN–LSTM model, which integrates spatial and temporal features, achieved 94.47% accuracy, confirming that fusing spatial and temporal information effectively enhances cross-subject generalization. When contrastive learning was applied individually to spatial or temporal features (CL-CNN and CL-LSTM), the accuracies were 91.83% and 85.92%, respectively, both lower than that of the CNN–LSTM model. This finding suggests that incorporating contrastive learning into either spatial or temporal features alone does not substantially improve generalization performance. The proposed CLAPISA framework outperformed all models, attaining an average accuracy of 96.71%, demonstrating that the synergy between spatiotemporal feature fusion and contrastive learning significantly enhances cross-subject generalization.

To further explore model generalization, we examined individual subject accuracy under both the baseline methods and the CLAPISA framework, as illustrated in Fig. 4(e) and Fig. 5. The LSTM and CL-LSTM models performed poorly on certain subjects (e.g., S11, S13, S15), yielding the lowest mean accuracy and the highest variance, reinforcing that temporal features alone are insufficient for robust generalization. The standalone contrastive-learning models (CL-CNN and CL-LSTM) exhibited inconsistent generalization, highlighting the importance of integrating contrastive learning with spatiotemporal features. The CNN–LSTM model demonstrated relatively stable performance. However, CLAPISA outperformed CNN–LSTM for most subjects, underscoring the added benefits of contrastive learning combined with spatiotemporal

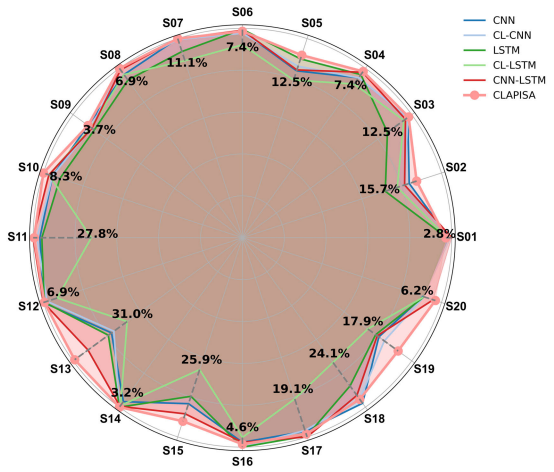

Fig. 5. Comparison of the proposed method with the baseline method via the LOSOCV method.

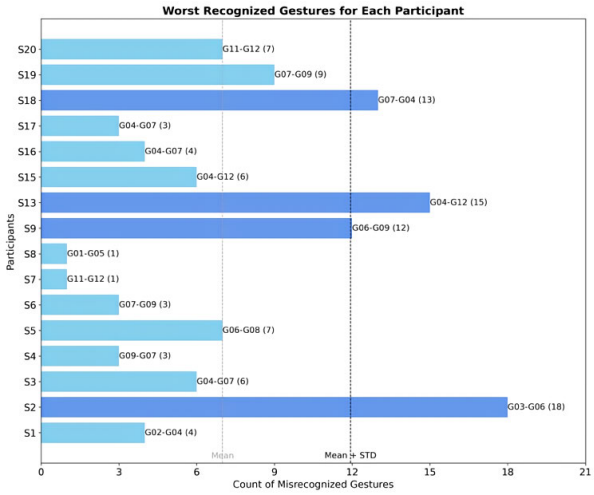

Fig. 6. Statistical analysis of worst-case gesture misrecognition across subjects.

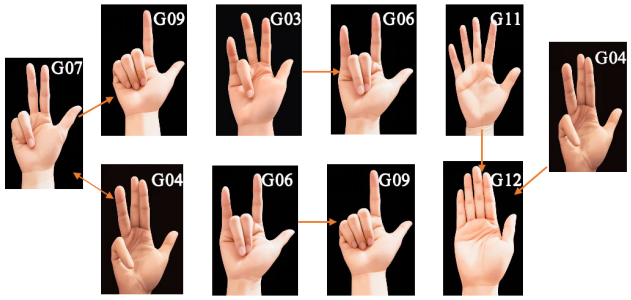

Fig. 7. Most frequently recognized incorrect gestures.

feature fusion. This advantage was particularly pronounced for challenging subjects such as S02 (+6.02%), S05 (+7.42%), S09 (+2.31%), S13 (+7.87%), S15 (+3.7%), S18 (+2.31%), and S19 (+12.31%).

To gain deeper insight into the challenges of cross-subject generalization, we analyzed the performance of the CNN–LSTM model at the individual-subject level. Fig.6 highlight the most frequently misclassified target classes for each subject, gestures G07, G03, G06, G04, and G11, likely due to the high kinematic similarity among these gestures. while Fig.7 shows the most commonly confused gesture pairs across subjects.

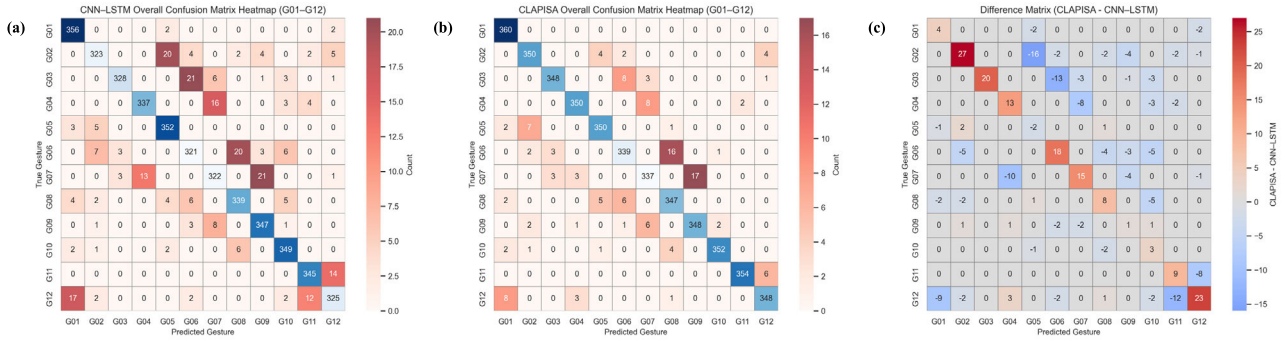

Fig. 8. Misclassification Heatmaps for CNN–LSTM and CLAPISA (LOSOCV, G01–G12). a. Confusion matrix of CNN–LSTM. B. Confusion matrix of the proposed CLAPISA. c. Difference matrix between CLAPISA and CNN–LSTM.

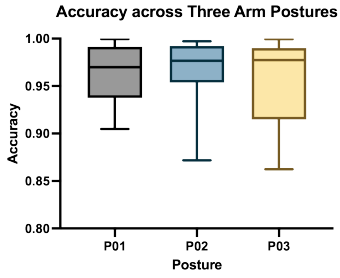

Fig. 9. Recognition accuracy across arm postures under LOSOCV.

Building upon this observation, we aggregated confusion matrices from all 20 subjects under the LOSOCV setting to compare global misclassification patterns between CNN–LSTM and CLAPISA. As illustrated in Fig. 8(a–c), CNN–LSTM frequently misclassified gestures such as G02→G05 (20 times), G03→G06 (21 times), G04→G07 (18 times), G06→G08 (20 times), G07→G09 (21 times), G11→G12 (14 times), and G12→G01 (17 times), revealing its limited capacity to distinguish between highly similar gestures. In contrast, the CLAPISA framework substantially reduced misclassification in these key gesture pairs. Specifically, it lowered the misclassification counts by 16, 13, 8, 4, 4, 8, and 9 respectively (see Fig. 8(b) and (c)). These quantitative results confirm that the integration of contrastive learning and spatiotemporal feature fusion in CLAPISA significantly enhances the model's ability to differentiate between confusable gestures.

To further validate the robustness of the proposed framework under varying arm configurations, for each test subject in the leave-one-subject-out validation, the 18 trials were divided according to arm posture: P01 (Trials 1-6, elbow flexed at 90°, palm parallel to the ground), P02 (Trials 7-12, elbow flexed at 90°, palm perpendicular to the ground), and P03 (Trials 13-18, elbow flexed at 45°, palm perpendicular to the ground). Mean recognition accuracy for every subject in each posture was analyzed with a one-way repeated-measures ANOVA (GraphPad Prism 9). The test revealed no systematic effect of posture on classification accuracy, $F = 0.4612, p = 0.6329$ (Fig. 9). Variance homogeneity was satisfied (Brown–Forsythe $p = 0.352$; Bartlett's $p = 0.113$). These results demonstrate that CLAPISA maintains consistent performance across the three arm configurations. In summary, while the CNN–LSTM model performs well in cross-subject tasks, the CLAPISA framework achieves more robust generalization by integrating spatiotemporal features with contrastive learning strategies.

In summary, the experimental results comprehensively demonstrate that the proposed CLAPISA framework, through the integration of contrastive learning and spatiotemporal features, significantly enhances feature representation and classification robustness, yielding superior performance in cross-subject gesture recognition compared to all baseline and ablated models.

V. Discussion

This study proposed the CLAPISA framework for cross-subject gesture recognition, integrating a Siamese network and contrastive learning into a CNN–LSTM spatiotemporal model to address generalization challenges posed by inter-individual variability. To rigorously evaluate the effectiveness of this approach, we conducted experiments using 5-Fold cross CV and LOSOCV methods. The experimental results clearly demonstrated that CLAPISA significantly improved several critical metrics, including reduced intra-class distances, increased inter-class distances, and enhanced FDR. These outcomes confirm the effectiveness of combining contrastive learning with spatiotemporal features to enhance feature representation and generalization.

Detailed analyses further revealed that CNN-based spatial modeling alone provided good inter-class discrimination but lacked intra-class compactness, whereas LSTM-based temporal modeling achieved better feature cohesion but compromised inter-class separation. Although fusing CNN and LSTM improved the overall performance, it remained insufficient for resolving gesture confusion across subjects, particularly for highly similar motions. To overcome this, CLAPISA incorporated a Siamese architecture to construct contrastive pairs, which significantly optimized the feature space. Intra-class distances were reduced to 2.26, and inter-class distances were increased to 15.77, resulting in an average LOSOCV accuracy of 96.7%, outperforming both the CNN–LSTM model (94.47%) and other baseline methods.

Sampling strategies also played a key role. Contrastive learning relies heavily on the balance between positive and

TABLE VI
COMPARISONS WITH STATE-OF-THE-ART METHODS

Method	Type	Dataset	Accuracy
Transfer Learning (CNN) [46]	sEMG	Myo [34]	93.88%
MBAGGDLM (Graph, General DL) [49]	skeleton	MSRA [50]	94.12%
		DHG [51]	92.00%
CLAPISA (ours)	bending	CGII [45]	96.71%

negative pairs. We empirically adopted a 1:2 positive-to-negative ratio, which enhanced model stability and representation reliability, consistent with previous findings on contrastive frameworks [47], [48]. Furthermore, margin values in the contrastive loss function were tuned (0.9, 1.0, 1.1), with the best results achieved at margin of 1.0. This configuration balanced inter-class separation and intra-class cohesion, further improving classification accuracy. Future research could explore adaptive margin mechanisms or distribution-aware contrastive losses for increased flexibility in real-world applications.

In addition, CLAPISA delivered consistent recognition accuracy across the three arm postures (P01-P03). On LOSOCV accuracy, a one-way repeated-measures ANOVA revealed no significant main effect of posture. Nevertheless, persistent misclassifications were observed for gesture pairs with high kinematic similarity (e.g., G03, G04, G06, G07, G11, G12). For example, minimal differences in middle and ring finger bending led to confusion between G06 and G07. Future work will incorporate finer kinematic descriptors or complementary sensing modalities—such as surface electromyography (sEMG) or vision-based tracking, to further improve discrimination.

It is also important to contextualize our findings with respect to other state-of-the-art methods. While Table VI summarizes reported accuracies from representative models using diverse datasets and sensor types, direct comparison should be made with caution due to variations in experimental settings and modalities. To ensure fairness, we implemented baseline models (CNN, LSTM, CNN–LSTM, CL-CNN, and CL-LSTM) under identical conditions on the same dataset (see Table V). CLAPISA consistently outperformed all these baselines, supporting its superior feature learning capabilities. The references in Table VI serve primarily as contextual benchmarks rather than direct competitors.

Despite the encouraging results, several limitations warrant should be discussed. First, the dataset comprises only 20 healthy young participants (age 24 $\pm$ 2), which limits generalizability to broader populations, including elderly individuals and those with motor impairments. Nevertheless, CLAPISA is inherently well-suited for such scenarios. Its subject-invariant feature encoding and contrastive learning mechanism allow it to generalize effectively from limited data, making it ideal for transfer learning, few-shot learning, and incremental learning tasks. These capabilities are crucial for deployment in clinical rehabilitation and assistive technology for aging populations. Second, although the CyberGlove II sensor used in our study provides high-resolution joint motion data that contribute to high recognition accuracy, its cost, potential drift, and comfort limitations may hinder scalability. Future work should consider lower-cost, ergonomic alternatives or adopt multimodal sensing approaches (e.g., sEMG, inertial sensors, vision-based tracking) to broaden practical applicability.

In summary, CLAPISA significantly advances cross-subject gesture recognition by integrating contrastive learning and Siamese networks with spatiotemporal feature fusion. It demonstrates high generalization capacity, even for difficult-to-distinguish gestures and diverse arm postures. The framework offers substantial potential for practical deployment in rehabilitation, virtual reality, and human–computer interaction, while paving the way for future developments in personalized and adaptive gesture recognition systems.

VI. CONCLUSION

This work presented CLAPISA, a cross-subject gesture-recognition framework that embeds a Siamese network with contrastive learning into a CNN-LSTM spatiotemporal pipeline. The model forms compact intra-class clusters and well-separated inter-class distributions, thereby alleviating subject-to-subject heterogeneity. Under leave-one-subject-out cross-validation, CLAPISA achieved 96.7 % accuracy, outperforming five strong baselines and markedly reducing errors for subjects with high variability. Performance gains stem from a balanced 1: 2 positive/negative pairing scheme and an empirically optimized margin of 1.0, which stabilize contrastive representation learning. CNN-based spatial encoding and LSTM-based temporal modeling further sharpen discrimination of subtle hand-motion nuances. A key next step is deployment in larger, clinically relevant cohorts, including elderly users and patients with motor impairments. Domain-adaptation, few-shot, and incremental strategies will be explored to maintain performance across disease stages and rehabilitation timelines. Remaining challenges include confusion among gestures with near-identical kinematics. Future work will incorporate finer motion descriptors and complementary sensing (e.g., surface-EMG, skeletal or vision data). In summary, by unifying spatiotemporal fusion, Siamese architectures, and contrastive learning, CLAPISA delivers a transferable, generalizable solution for cross-subject gesture recognition and establishes a solid foundation for real-world systems in hand rehabilitation, human–computer interaction, and VR/AR interfaces.

REFERENCES

[1] A. Rashid and O. Hasan, "Wearable technologies for hand joints monitoring for rehabilitation: A survey," *Microelectron. J.*, vol. 88, pp. 173–183, Jun. 2019.

[2] P. Langhorne, F. Coupar, and A. Pollock, "Motor recovery after stroke: A systematic review," *Lancet Neurol.*, vol. 8, no. 8, pp. 741–754, Aug. 2009.

[3] C. Winstein and R. Varghese, "Been there, done that, so what's next for arm and hand rehabilitation in stroke?" *NeuroRehabilitation*, vol. 43, no. 1, pp. 3–18, Jul. 2018.

[4] K. J. Waddell, M. J. Strube, R. G. Tabak, D. Haire-Joshu, and C. E. Lang, "Upper limb performance in daily life improves over the first 12 weeks poststroke," *Neurorehabilitation Neural Repair*, vol. 33, no. 10, pp. 836–847, Oct. 2019.

[5] L. Yu, D. Xiong, L. Guo, and J. Wang, "A remote quantitative fugl-meyer assessment framework for stroke patients based on wearable sensor networks," *Comput. Methods Programs Biomed.*, vol. 128, pp. 100–110, May 2016.

[6] M. C. Ariesta, F. Wiryana, and G. P. Kusuma, "A survey of hand gesture recognition methods in sign language recognition," *Pertanika J. Sci. Technol.*, vol. 26, no. 4, pp. 1659–1675, Oct. 2018.

[7] J. Kawaguchi, S. Yoshimoto, Y. Kuroda, and O. Oshiro, "Estimation of finger joint angles based on electromechanical sensing of wrist shape," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 9, pp. 1409–1418, Sep. 2017.

[8] Y. Geng, O. W. Samuel, Y. Wei, and G. Li, "Improving the robustness of real-time myoelectric pattern recognition against arm position changes in transradial amputees," *BioMed Res. Int.*, vol. 2017, pp. 1–10, Apr. 2017.

[9] X. Song, S. S. Van De Ven, L. Liu, F. J. Wouda, H. Wang, and P. B. Shull, "Activities of daily living-based rehabilitation system for arm and hand motor function retraining after stroke," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 621–631, 2022.

[10] J. Cheng, F. Wei, C. Li, Y. Liu, A. Liu, and X. Chen, "Position-independent gesture recognition using sEMG signals via canonical correlation analysis," *Comput. Biol. Med.*, vol. 103, pp. 44–54, Dec. 2018.

[11] H. E. Williams, A. W. Shehata, M. R. Dawson, E. Scheme, J. S. Hebert, and P. M. Pilarski, "Recurrent convolutional neural networks as an approach to position-aware myoelectric prosthesis control," *IEEE Trans. Biomed. Eng.*, vol. 69, no. 7, pp. 2243–2255, Jul. 2022.

[12] L. Dipietro, A. M. Sabatini, and P. Dario, "A survey of glove-based systems and their applications," *IEEE Trans. Syst. Man, Cybern., C, Appl. Rev.*, vol. 38, no. 4, pp. 461–482, Jul. 2008.

[13] L. Guo, Z. Lu, L. Yao, and S. Cai, "A gesture recognition strategy based on A-mode ultrasound for identifying known and unknown gestures," *IEEE Sensors J.*, vol. 22, no. 11, pp. 10730–10739, Jun. 2022.

[14] M.-K. Liu, Y.-T. Lin, Z.-W. Qiu, C.-K. Kuo, and C.-K. Wu, "Hand gesture recognition by a MMG-based wearable device," *IEEE Sensors J.*, vol. 20, no. 24, pp. 14703–14712, Dec. 2020.

[15] H. Kato and K. Takemura, "Hand pose estimation based on active bone-conducted sound sensing," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput., Adjunct*, Sep. 2016, pp. 109–112.

[16] A. Currà et al., "Near-infrared spectroscopy as a tool for in vivo analysis of human muscles," *Sci. Rep.*, vol. 9, no. 1, p. 8623, Jun. 2019.

[17] K. Subramanian, C. Savur, and F. Sahin, "Using photoplethysmography for simple hand gesture recognition," in *Proc. IEEE 15th Int. Conf. Syst. Syst. Eng. (SoSE)*, Jun. 2020, pp. 307–312.

[18] W. G. Darling, K. J. Cole, and G. F. Miller, "Coordination of index finger movements," *J. Biomech.*, vol. 27, no. 4, pp. 479–491, Apr. 1994.

[19] F.-C. Su, Y. L. Chou, C. S. Yang, G. T. Lin, and K. N. An, "Movement of finger joints induced by synergistic wrist motion," *Clin. Biomechanics*, vol. 20, no. 5, pp. 491–497, Jun. 2005.

[20] D. G. Kamper, T. G. Hornby, and W. Z. Rymer, "Extrinsic flexor muscles generate concurrent flexion of all three finger joints," *J. Biomechanics*, vol. 35, no. 12, pp. 1581–1589, Dec. 2002.

[21] T. Hamaguchi et al., "Support vector machine-based classifier for the assessment of finger movement of stroke patients undergoing rehabilitation," *J. Med. Biol. Eng.*, vol. 40, no. 1, pp. 91–100, Feb. 2020.

[22] J. G. Colli-Alfaro, A. Ibrahim, and A. L. Trejos, "Design of user-independent hand gesture recognition using multilayer perceptron networks and sensor fusion techniques," in *Proc. IEEE 16th Int. Conf. Rehabil. Robot. (ICORR)*, Jun. 2019, pp. 1103–1108.

[23] W. N. Al-Sharu and A. M. Alqudah, "Enhancing prediction of prosthetic fingers movement based on sEMG using mixtures of features and random forest," *Int. J. Recent Technol. Eng. (IJRTE)*, vol. 8, no. 4, pp. 289–294, Nov. 2019.

[24] V. B. Srinivasan, M. Islam, W. Zhang, and H. Ren, "Finger movement classification from myoelectric signals using convolutional neural networks," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2018, pp. 1070–1075.

[25] C. Millar, N. Siddique, and E. Kerr, "LSTM network classification of dexterous individual finger movements," *J. Adv. Comput. Intell. Intell. Informat.*, vol. 26, no. 2, pp. 113–124, Mar. 2022.

[26] C. Millar, N. Siddique, and E. Kerr, "LSTM classification of sEMG signals for individual finger movements using low cost wearable sensor," in *Proc. Int. Symp. Community-centric Syst. (CcS)*, Sep. 2020, pp. 1–8.

[27] G. Yuan, X. Liu, Q. Yan, S. Qiao, Z. Wang, and L. Yuan, "Hand gesture recognition using deep feature fusion network based on wearable sensors," *IEEE Sensors J.*, vol. 21, no. 1, pp. 539–547, Jan. 2021.

[28] N. K. Karnam, S. R. Dubey, A. C. Turlapaty, and B. Gokaraju, "EMGHandNet: A hybrid CNN and bi-LSTM architecture for hand activity classification using surface EMG signals," *Biocybernetics Biomed. Eng.*, vol. 42, no. 1, pp. 325–340, Jan. 2022.

[29] Y. Geng et al., "A CNN-attention network for continuous estimation of finger kinematics from surface electromyography," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 6297–6304, Jul. 2022.

[30] M. Ur Rehman et al., "Dynamic hand gesture recognition using 3D-CNN and LSTM networks," *Comput., Mater. Continua*, vol. 70, no. 3, pp. 4675–4690, 2022.

[31] W. Qi, S. E. Ovur, Z. Li, A. Marzullo, and R. Song, "Multi-sensor guided hand gesture recognition for a teleoperated robot using a recurrent neural network," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 6039–6045, Jul. 2021.

[32] M. Montazerin, E. Rahimian, F. Naderkhani, S. F. Atashzar, S. Yanushkevich, and A. Mohammadi, "Transformer-based hand gesture recognition from instantaneous to fused neural decomposition of high-density EMG signals," *Sci. Rep.*, vol. 13, no. 1, p. 11000, Jul. 2023.

[33] T. Tommasi, F. Orabona, C. Castellini, and B. Caputo, "Improving control of dexterous hand prostheses using adaptive learning," *IEEE Trans. Robot.*, vol. 29, no. 1, pp. 207–219, Feb. 2013.

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

[35] J. G. Colli Alfaro and A. L. Trejos, "User-independent hand gesture recognition classification models using sensor fusion," *Sensors*, vol. 22, no. 4, p. 1321, Feb. 2022.

[36] Y. Wang, P. Zhao, and Z. Zhang, "A deep learning approach using attention mechanism and transfer learning for electromyographic hand gesture estimation," *Expert Syst. Appl.*, vol. 234, Dec. 2023, Art. no. 121055.

[37] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, Jan. 2020, pp. 1597–1607.

[38] M. Bhuyan, A. K. Talukdar, P. Gupta, and R. H. Laskar, "Low cost data glove for hand gesture recognition by finger bend measurement," in *Proc. Int. Conf. Wireless Commun. Signal Process. Netw. (WiSPNET)*, Aug. 2020, pp. 25–31.

[39] O. Saha and S. Maji, "PARTICLE: Part discovery and contrastive learning for fine-grained recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2023, pp. 167–176.

[40] F. Wang, L. Chen, F. Xie, C. Xu, and G. Lu, "Few-shot text classification via semi-supervised contrastive learning," in *Proc. 4th Int. Conf. Natural Lang. Process. (ICNLP)*, Mar. 2022, pp. 426–433.

[41] H. Lu, Y. Huo, M. Ding, N. Fei, and Z. Lu, "Cross-modal contrastive learning for generalizable and efficient image-text retrieval," *Mach. Intell. Res.*, vol. 20, no. 4, pp. 569–582, Aug. 2023.

[42] C. Jia, J. Xue, K. Lu, and Z. Wu, "Semi-supervised contrastive learning of global and local representation for 3D medical image segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2023, pp. 26–30.

[43] Z. Lai, X. Kang, H. Wang, X. Zhang, W. Zhang, and F. Wang, "Contrastive domain adaptation: A self-supervised learning framework for sEMG-based gesture recognition," in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, Oct. 2022, pp. 1–7.

[44] Z. Dai, S. Zhai, P. Qin, R. Chai, and P. Zhao, "WVGR: Gesture recognition based on WiFi-video fusion," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Aug. 2023, pp. 1–6.

[45] X. Hu, A. Song, J. Wang, H. Zeng, and W. Wei, "Finger movement recognition via high-density electromyography of intrinsic and extrinsic hand muscles," *Sci. Data*, vol. 9, no. 1, p. 373, Jun. 2022.

[46] Z. Zhang, S. Liu, Y. Wang, W. Song, and Y. Zhang, "Online cross session electromyographic hand gesture recognition using deep learning and transfer learning," *Eng. Appl. Artif. Intell.*, vol. 127, Jan. 2024, Art. no. 107251.

[47] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Netw.*, vol. 106, pp. 249–259, Oct. 2018.

[48] R. Jiang, T. Nguyen, P. Ishwar, and S. Aeron, "Supervised contrastive learning with hard negative samples," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jun. 2022, pp. 1–8.

[49] A. S. M. Miah, M. A. M. Hasan, and J. Shin, "Dynamic hand gesture recognition using multi-branch attention based graph and general deep learning model," *IEEE Access*, vol. 11, pp. 4703–4716, 2023.

[50] X. Sun, Y. Wei, S. Liang, X. Tang, and J. Sun, "Cascaded hand pose regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 824–832.

[51] Q. De Smedt, H. Wannous, and J.-P. Vandeborre, "Skeleton-based dynamic hand gesture recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2016, pp. 1206–1214.