Multimedia Tools and Applications (2024) 83:68013–68040
https://doi.org/10.1007/s11042-024-18283-z

Fusion of kinematic and physiological sensors for hand gesture recognition

Aiguo Wang[1] · Huancheng Liu[1] · Chundi Zheng[1] · Huihui Chen[1] · Chih-Yung Chang[2]

Received: 12 May 2023 / Revised: 8 January 2024 / Accepted: 15 January 2024 /
Published online: 29 January 2024

Abstract
The uncertainty of hand gestures, the variability of gestures across subjects, and the high cost of collecting a large amount of annotated data lead to a great challenge to the robust recognition of gestures, and thus it remains quite crucial to capture the informative features of hand movements and to mitigate inter-subject variations. To this end, we propose a gesture recognition model that uses two different types of sensors and optimizes the feature space towards enhanced accuracy and better generalization. Specifically, we use an accelerometer and a surface electromyography sensor to capture kinematic and physiological signals of hand movements. We use a sliding window to divide the streaming sensor data and then extract time-domain and frequency-domain features from each segment to return feature vectors. Afterwards, the feature space is optimized with a feature selector and a gesture recognizer is optimized. To handle the case where no labeled training data are available for a new user, we apply the transfer learning technique to reuse the cross-subject knowledge. Finally, extensive comparative experiments concerning different classification models, different sensors, and different types of features are conducted. Results show that the joint use of kinematic and physiological sensors generally outperforms the use of single sensor, indicating the synthetic effect of different sensors, and that the use of transfer learning helps improve the cross-subject recognition accuracy. In addition, we quantitatively investigate the impact of null gesture on a gesture recognizer and results indicate that null gesture would lower its accuracy, enlightening related studies to consider it.

Keywords Gesture recognition · Electromyography · Accelerometer · Feature extraction · Cross-subject

✉ Chih-Yung Chang
cychang@mail.tku.edu.tw

[1] School of Electronic Information Engineering, Foshan University, Foshan 528225, China

[2] Department of Computer Science and Information Engineering, Tamkang University, Taipei 251301, Taiwan

1 Introduction

With the rapid development of pervasive computing and the ever-increasing demands for intelligent services, the way of human–computer interaction has changed from the computer-centered scheme to human-centered one [1], and researchers have conducted considerable work on the interaction methods, among which, hand gestures are a convenient and natural medium [2]. Compared with other modalities (e.g., voice, keyboard, mouse, and camera), gestures have the advantages of naturalness, directness, simplicity, robustness, and portability, and thus they are widely used in rich scenarios (e.g., entertainment, healthcare, security, intelligent control, biometrics, soft biometrics, and intelligent protections for users) [3–6]. However, activity complexity is inherent to the nature of human beings and hand gestures typically have the characteristics of uncertainty, diversity, and concurrency [7]. For example, different people have different ways of performing an activity, known as *inter-subject* variation; even an individual may perform an activity differently at different places and times, known as *intra-subject* variation. Besides, some activities, even with different semantics, can trigger similar sensor data, which would hinder the ability of a recognizer and result in degraded accuracy [8]. Hence, the accurate and robust recognition of hand gestures remains challenging and crucial and has been attracting considerable attention from industry and academia [9–11].

Towards higher recognition accuracy and better generalization of gesture recognition methods, researchers have conducted studies on the sensing technology and classification models. From the aspect of sensing technology, we can group them into vision-based, motion sensor-based, and physiological sensor-based approaches [12]. Vision-based approaches apply vision units (e.g., cameras and videos) to capture the images of the target action and recognize gestures via computer vision techniques [13, 14]. Such methods are often used in gaming, entertainment, and security precautions [14]. However, they have the privacy issue and are not applicable in living environments, which greatly limits their practical use [15]. Also, the captured images are easily affected by background changes, illumination changes, environmental noise, occlusions, and low contrast resolution [16, 17]. In contrast to vision-based gesture recognition, motion sensor-based methods have a wider range of applications in indoor and outdoor settings and even in the extreme conditions such as deep sea and high-altitude environments [18]. Motion sensor-based models are usually built into the wearable devices that can be worn on different parts of the human body (e.g., hands, chest, waist, wrist, legs, and feet). Particularly, multiple devices can be simultaneously worn [19, 20]. However, motion sensors mainly reflect the overall movements of a subject and have limited power in capturing the dynamic and rich hand movements. Considering that hand movements are accompanied by changes in physiological signals, researchers also explore the use of physiological sensors, among which, the surface electromyographic (sEMG) sensor is the widely used one in gesture recognition. The sEMG signals are collected by the patch electrodes, which is safe and non-invasive compared with the pinhole electrodes. Hence, the bio-signals of muscle changes can be collected by placing the electrodes to corresponding muscles. Though the sEMG signals are different across subjects, the patterns are similar during hand movements and can thus be used to recognize gestures [21, 22].

According to the used classification models, we can categorize existing methods into classical machine learning-based and deep learning-based gesture recognition. Deep learning techniques have the end-to-end feature learning and recognizer training capability, but they require massive computing resources and a large amount of collected data [23], which

is often inapplicable to a portable and light-weighted gesture recognizer. Thus, machine learning-based methods remain a viable option and even have the priority in building a gesture recognizer. Commonly used machine learning models range from generative models (e.g., naïve baye, and hidden Markov model) to discriminative models (e.g., *k*-nearest neighbors, support vector machine, and random forests). Particularly, for machine learning-based models, the extraction and use of features largely determines the performance of a gesture recognizer, while most of existing studies aim to extract a large number of features in developing a gesture recognizer and ignore the existence of irrelevant and redundant features. These unimportant features would potentially increase the model complexity, prediction time, and energy consumption. In addition, since different users may perform gestures in a quite different way, the sensor data of an existing user and sensor data of a new user probably follow a different distribution, which would inevitably lead to degraded recognition accuracy [24]. One natural way is to collect and annotate sufficient labeled data for each user, which is time-consuming. Hence, a way of transferring cross-subject knowledge is expected. Besides, gesture recognition faces the null gesture issue, that is, parts of the sensor data do not correspond to the predefined gestures in an application. To this end, we in this study propose a gesture recognition model with two different types of sensors. Specifically, we use the accelerometer and sEMG sensor to capture both kinematic and physiological signals of hand movements and extract a large number of features from the segments returned by a sliding window. To remove irrelevant and redundant features, we apply feature selection algorithms to optimize the feature space of a gesture recognizer. For the cross-subject problem, we utilize the transfer learning technique to reuse training data of an existing user for training a robust recognizer. The main contributions of this study are as follows.

- A device embedded with an accelerometer and an sEGM sensor is developed to capture hand movements. A large number of time-domain and frequency-domain features are extracted from each sensor. Particularly, we consider features from different axes of the 3-axis accelerometer and different channels of the sEMG sensor. Afterwards, three different feature selection algorithms are used and evaluated to optimize the feature space towards a robust gesture recognizer.
- Transfer learning technique is utilized to train a gesture recognition model for the cross-subject scenario where sufficient labeled data is unavailable for a new user to train an accurate gesture recognizer. We illustrate the learning scheme and present a transfer learning-based gesture recognizer.
- Extensive comparative studies are conducted on experimental data. Results show the superiority of the use of a feature selector and the joint use of sEMG sensor and accelerometer in achieving higher accuracy. Furthermore, three widely used transfer learning techniques are utilized and compared in the cross-subject setting, where the joint distribution adaptation algorithm generally performs better.
- Considering that null gesture is an inherent part of hand movements, we quantitatively evaluate its impact on a gesture recognizer to prompt researchers to notice its existence and effect in designing a recognizer. Experimental results show that the inclusion of null gesture would lower the accuracy.

The rest of the paper is organized as follows. Section 2 presents related work on gesture recognition from the aspects of sensor technology and classification model. Section 3 details the proposed gesture recognition model. Section 4 gives the experimental setup, results, and analysis, followed by the conclusion section.

2 Related work

The choice of sensing units and the use of machine learning model largely determine the application scope and the power of a gesture recognizer. We in this section summarize related work from the aspects of sensing units and gesture recognition model and compare them with our work.

2.1 Sensing units

To adapt to different human–computer interactions and better serve human-centered applications, researchers have conducted a wealth of work on sensor technologies and classification models. For the use of sensors, different sensors have their own advantages in capturing human activities in different scenarios and they are basically grouped into vision-based sensors, motion-based sensors, and physiological signal-based sensors. Vision-based sensors usually use devices (e.g., Kinect and EyeToy) to sense human motion in real-time. For example, Ren et al. designed a Kinect-based gesture recognition system that achieved the average accuracy of 93.20% [25]. Song et al. proposed a new quality-aware human–computer interaction framework to recognize hand gestures, where a fusion algorithm was used to combine hand contours, different positions, and hand motions [26]. Chen et al. recognized gestures with a hidden Markov model that combined spatial and temporal features of the input sequences into a feature vector [27].

In contrast to vision-based sensors that have limited use and often suffer from privacy issues, motion-based sensors capture the kinematic signals of hand movements to infer the on-going gestures [28]. Motion sensors are often embedded into a wearable device for pervasive use. There are a variety of motion sensors available for use, among which the accelerometer is the widely used one. For example, Xia et al. utilized a wearable device with motion sensors to identify the work status of assembly line workers in order to optimize the operational flow [29]. Considering that a single accelerometer sensor has limited power in recognizing complex activities, Jiang et al. explored the convolutional neural networks and multilayer perceptron neural networks to extract features and classify gestures with the inertial measurement unit. Their results achieved 97% recognition accuracy [30]. Wang et al. used the accelerometer and gyroscope to infer activities and obtained better accuracy with the two sensors [31]. However, since the mapping of hand characteristics by motion sensors is not comprehensive and easily confused by similar gestures, physiological sensors remain an option to sense and analyze the physiological signals, among which sEGM sensors infer corresponding actions by measuring the changes in muscle contractions. For example, Sapienza et al. used sEMG sensors to infer four different wrist movements and got 92.87% recognition accuracy [32]. Yang et al. developed a gesture recognition system with sEMG sensors to help those with prostheses. They tested its performance on five subjects and obtained the average accuracy of 94.66% [33]. Since motion sensors and physiological sensors can reflect hand movements from different views [34, 35], researchers proposed to combine different types of sensors towards enhanced accuracy. For example, Ceolini et al. proposed a fully neuromorphic sensor fusion approach to combining sEMG and visual information for gesture recognition [36]. Lu et al. evaluated a group of wearable devices embedded with the sEMG sensor and accelerometer [37]. Jiang et al. used an sEMG sensor and motion measurement unit to design a gesture recognizer [38]. Though most of the existing studies obtain satisfactory accuracy with different sensors, but there

is still a lack of further research and analysis on which features of sensor data to extract and use and on the power of different types of sensors when they are utilized jointly or separately. Hence, it is necessary to further compare time-domain and frequency-domain features and evaluate the features extracted from different channels of an sEMG sensor and different axes of an accelerometer. Besides, irrelevant and redundant features probably exist in the feature space, and hence feature selection is expected.

2.2 Gesture recognition model

Besides exploring different sensing units, researchers have designed a large number of gesture recognition models and we can broadly group them into traditional machine learning- and deep learning- based models. Deep learning techniques, a subset of machine learning, have the ability of end-to-end feature learning and classifier training and can relieve users of extracting discriminate features. For example, Guo et al. used a convolutional neural network (CNN) to recognize static gestures [39]. Sun et al. designed a CNN to recognize gesture regions in real-time with 98.3% accuracy [40]. Although deep learning models achieve satisfactory accuracy, they generally require large amounts of data and abundant computational resources to train a robust recognizer. Hence, traditional machine learning models still remain a priority in developing a light-weighted gesture recognizer especially in the edge computing setting. For example, Lee et al. proposed a hidden Markov model-based gesture recognizer that can recognize gestures with a 93.14% reliability [41]. Bargellesi et al. proposed a random forest-based gesture recognizer to infer ten gestures [42]. Pomboza-Junez et al. used the support vector machine to recognize gestures with sEMG sensors embedded in a bracelet [43]. Importantly, for machine learning-based models, the extraction and use of features greatly determines the performance of a gesture recognizer, while most of the existing studies, to the best of our knowledge, extract a large number of features without paying much attention to irrelevant and redundant features. These unimportant features would result in a gesture recognizer with higher computational complexity and energy consumption, which motivates us to utilize feature selection methods to optimize the feature space before building a recognizer.

Due to the inter-subject variation, the distribution between sensor data of an existing user and that of a new user could be quite different. Particularly, for a new user, we usually do not have his/her (sufficient) labeled data to train a robust gesture recognition model, since the collection of data is time-consuming and often impractical. Consequently, if we directly apply an existing gesture recognizer to new users, we would suffer from degraded accuracy. One simple solution is to collect a large amount of labeled data for each user to train a gesture recognizer. Clearly, this plan is time-consuming. Accordingly, there are studies that use transfer learning techniques to reuse knowledge in a new scenario. For example, Yu et al. used a convolutional neural network-based transfer learning model for sEMG-based instantaneous hand gesture recognition. Their experimental results show that the proposed model had better generalization and greatly reduced the training time [44]. Wang et al. proposed a cross-domain stratified transfer learning framework that can exploit the internal affinity of classes for intra-class knowledge transfer. Experimental results show that the proposed model obtains a 7.68% accuracy enhancement [45]. Côté-Allard et al. used the sEMG sensor to collect two sets of data for training a gesture recognizer, where the first data set was from 18 subjects and the second data set was from other 17 subjects. They then utilized the convolutional neural network-based transfer learning to train a gesture recognizer on the first dataset and make predictions on the second dataset [46]. Although achieving better performance, most of the existing studies mainly work on single-source data other than

multi-source data and often ignore the optimization of feature space. Therefore, we in this study would explore and compare transfer learning techniques in the cross-subject setting, where the accelerometer and sEMG sensors are used to recognize gestures and labeled data from the new user are unavailable. In addition, previous studies often consider the predefined gestures in a specific application and ignore the null gesture in developing a practical recognizer, which further motivates us to conduct an experimentally comparative study. Table 1 summarizes the comparisons between our work and related work concerning information fusion, feature selection, cross-subject learning, and null gesture.

3 The proposed gesture recognition model

We in this section first present the proposed gesture recognition chain and then detail the hardware used for capturing the movements of hands. Afterwards, we introduce how to collect experimental datasets for four different gestures from the volunteers. Finally, we illustrate how to extract and select features from sensor data.

3.1 Gesture recognition chain

Figure 1 presents the gesture recognition chain, which mainly consists of the training phase and prediction phase. In the training phase, we segment the raw sensor data into segments using the sliding window technique and extract a variety of features from each segment to derive a feature vector. Obviously, feature extraction plays an important role in the chain. To optimize the feature space and reduce model complexity, a feature selection algorithm is utilized before training the gesture recognizer. In the prediction process, we segment the test data, extract features from the segment, and infer the on-gong gestures.

Accordingly, algorithm 1 gives the pseudo-code of the gesture recognition model, where lines 1–3 indicate the training stage and lines 4–6 are the prediction phase.

Algorithm 1 Pseudo-code of the gesture recognition model

Input: a labeled train set D, gesture label L, a test sample x
Output: the gesture label A of x

// the training of a gesture recognition model
1. extract feature sets F_1 and F_2 from D
2. feature selection on F_1 and F_2 to get a feature subset F
3. train a gesture recognizer GR using D and F

// gesture recognition using GR
4. extract feature F from x
5. $A = GR(x, F)$ // return the gesture label of x
6. **return** A

Table 1 Comparisons between our work and related work

	Information fusion	Feature selection	Cross-subject learning	Null gesture
Wang et al. [31]	√	√	×	×
Ceolini et al. [36]	√	×	×	×
Jiang et al. [38]	√	×	×	×
Bhushan et al. [3]	×	√	×	×
Yu et al. [44]	×	×	√	×
Wang et al. [45]	√	×	√	×
Our work	√	√	√	√

3.2 Hardware

To capture kinematic and physiological signals of hand movements, we design a sensing unit with an sEMG sensor and accelerometer. The choice of accelerometer is that it is closely related to gesture kinematic signals and widely used in previous studies [29]. The accelerometer signals are smoothed with a filter and the sEMG signals are preprocessed with rectification and integration operations. Both the processed signals are then transmitted to the server for subsequent analysis. Figure 2 shows the placements of the sensors. Since hand movements are closely related to the muscle cooperation of the forearm, the functions of each muscle are different. In our experiments, the palmaris longus and extensor digitorum muscles are more active than others when one performs gestures, so the sEMG sensor is placed on the two positions [47]. The accelerometer is worn on the position that has the greatest movement range.

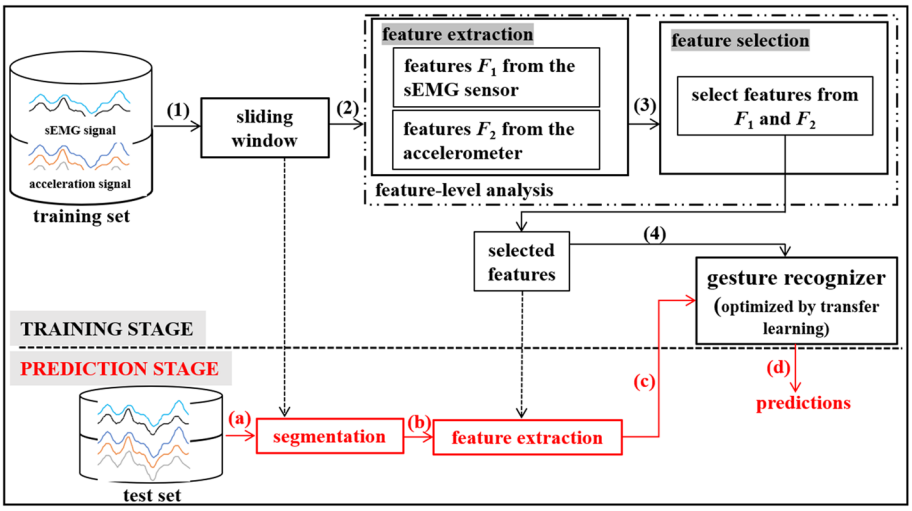
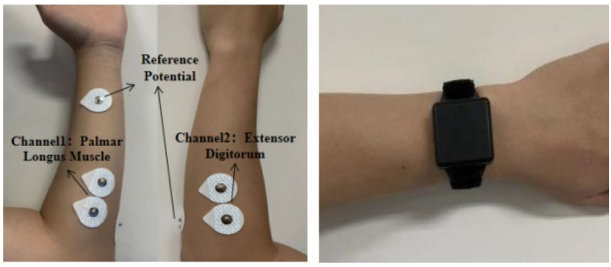


Fig. 1 The gesture recognition chain. The upper part denotes the training stage with procedures of (1)—(4); the lower part is the prediction stage with procedures of (**a**)–(**d**)



(a) the sEMG sensor (b) accelerometer

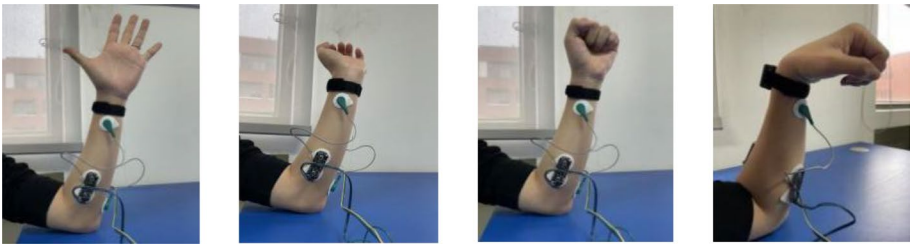
Fig. 2 The placements of the sensors

3.3 Data collection

In this study, we recruited four male volunteers (named S1, S2, S3, and S4) with the average age weight of 66.5 kg, average height of 174.5 cm, and mean age of 24 years old. Before data collection, subjects perform the simple hand movement training in their own styles. During data collection, the volunteer sat on a chair and was in a relaxed state. Streaming sensor data related to four commonly used hand gestures, including *open*, *turn up*, *turn down*, and *fist* are collected. Particularly, we also collect the sensor data of null gesture. Figure 3 presents the exemplary gestures. For each of the gestures, we collect two-hundred groups of data from each subject.

3.4 Feature extraction and selection

For building a gesture recognizer, the streaming sensor data need to be first segmented. We here use a sliding non-overlapping window to segment sensor data and extract a variety of time-domain and frequency-domain features. Time-domain and frequency-domain features are two views of the raw signals. Frequency-domain features are obtained by transforming the raw signals into frequency domain with fast Fourier transform [48]. For the dual-channel sEMG data, we extract time-domain features (e.g., *mean*, *maximum*, *minimum*, *standard deviation*, *mode*, *difference between maximum* and *minimum*) and frequency-domain features (e.g., shape features and amplitude features) from each channel. Since correlation exists between the two channels, we also extract features from the differences between readings of the two channels. For the 3-axis accelerometer, we extract features from each of its axes and its resultant axis (called *rlt*-axis in this study) [49]. Suppose $\{\boldsymbol{a}_x, \boldsymbol{a}_y, \boldsymbol{a}_z\}$



(a) *open* (b) *turn up* (c) *fist* (d) *turn down*

Fig. 3 Illustration of different gestures

correspond to the readings of the three axes, then the value $\boldsymbol{r}$ of the resultant axis is $\boldsymbol{r} = \sqrt{a_x^2 + a_y^2 + a_z^2}$. Table 2 lists the extracted time-domain and frequency-domain features.

During the feature extraction phase, we usually extract a large number of features in order to better capture the gesture characteristics, which probably introduces irrelevant and redundant features and further degrades the performance of a gesture recognizer. Accordingly, one feasible way is to reduce the data dimensionality with a feature selector for feature space optimization. Noteworthily, we extract all the features in Table 2 during the training stage and only extract the optimal feature set returned by the feature selection algorithm during the prediction stage.

4 Experimental setup and results

4.1 Experimental setup

To demonstrate of the roles of accelerometer and sEMG sensor, we conduct the experiments in three settings: (1) only accelerometer is used; (2) only sEMG sensor is used; and (3) both accelerometer and sEMG sensor are used. The sEMG sensor and accelerometer work at sampling rates of 200 Hz and 100 Hz, respectively. The streaming sensor signals are segmented using a non-overlapping 1 s sliding window, which generates 757 *open* samples, 655 *turn up* samples, 799 *turn down* samples, 628 *fist* samples, and 441 *null gesture* samples. Then, time-domain and frequency-domain features are extracted from the segment. To obtain a set of discriminate features, we evaluate three widely used and powerful feature selectors, including reliefF, correlation-based feature selection (CFS), and minimal redundancy maximal relevancy (MRMR). We adopt the best feature selector to obtain the finally selected features. The number of selected features is experimentally determined.

Besides, towards fair comparisons, we evaluate five commonly used classification models with different metrics, including support vector machine (SVM) with radial basis function kernels, random forest (RF), naive Bayes (NB), logistic regression (LR), and *k*-nearest neighbor (KNN). A five-fold cross-validation is used to generate independent training and test sets, where the former is used to train a gesture recognizer. As for performance metrics, we use accuracy (*acc*), precision (*pre*), recall (*rec*), and *F*1.

Table 2 Extracted features

		Time-domain features	Frequency-domain features
Accelerometer	X-axis Y-axis Z-axis *rlt*-axis	*mean*, *maximum*, *minimum*, *standard deviation*, *difference between maximum and minimum*, *mode*	shape features: *mean*, *standard deviation*, *skewness*, *kurtosis*; amplitude features: *mean*, *standard deviation*, *skewness*, *kurtosis*
sEMG	channel 1 channel 2 difference of two channels	*mean*, *maximum*, *minimum*, *standard deviation*, *difference between maximum* and *minimum*, *mode*	shape features: *mean*, *standard deviation*, *skewness*, *kurtosis*; amplitude features: *mean*, *standard deviation*, *skewness*, *kurtosis*

$$acc = \sum_{i=1}^{C} T_i / \sum_{i=1}^{C} NT_i \tag{1}$$

$$pre = \frac{1}{C} \sum_{i=1}^{C} \frac{T_i}{NP_i} \tag{2}$$

$$rec = \frac{1}{C} \sum_{i=1}^{C} \frac{T_i}{NT_i} \tag{3}$$

$$F1 = \frac{2 * pre * rec}{pre + rec} \tag{4}$$

where $L = \{L_1, L_2, \ldots, L_C\}$ denotes different gestures, C is the number of gestures of interest, T_i is number of samples from class L_i that are correctly classified, NP_i is the number of samples predicted with L_i, and NT_i is the number of samples from L_i ($1 \le i \le C$).

In the cross-subject experiments, we first present the degraded accuracy of traditional gesture recognizers and then preliminarily explore transfer learning techniques. Transfer learning aims to reuse the knowledge from one or more source tasks and applies the knowledge to a target task and it has been successfully applied in many fields such as computer vision and natural language processing [50]. This motivates us to explore the use of transfer learning. We herein reuse data from an existing user under the paradigm of transfer learning. Figure 4 presents the scheme, where an improved gesture recognizer is adjusted with data from subject V1 and V2. Specifically, we adopt the feature-based transfer learning technique that first transforms the sensor data of V1 and V2 into a common subspace and then trains a gesture recognizer on the transformed data of V1 to make predictions for the transformed data of V2.

Specifically, we use three widely used transfer learning methods (i.e., transfer component analysis (TCA) [50], joint distribution adaptation (JDA) [51], and geodesic flow kernel (GFK) [52]). The main idea of TCA is to map the data from both domains into a high-dimensional Reproducing Kernel Hilbert Space (RKHS) when the data between source domain (denoting an existing user in this study) and target domain (denoting a new user) have different distributions. In RKHS, TCA minimizes the distance between source and

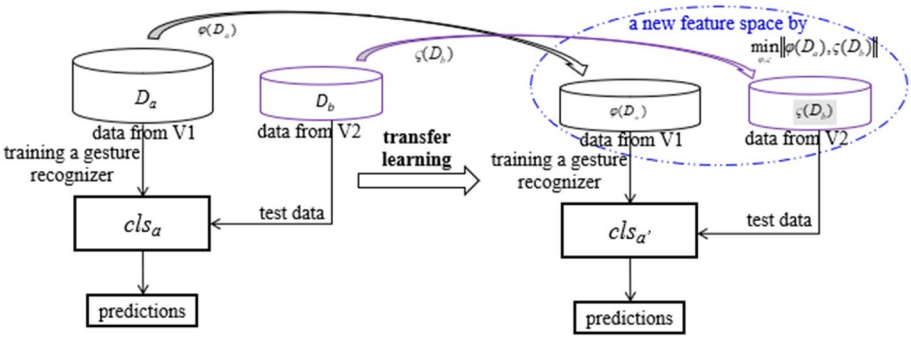


Fig. 4 Transfer learning-based gesture recognition model

target domains as much as possible while preserving their own properties to a large extent. JDA is a distribution adaptation method that considers both marginal and conditional probability distributions. The idea of GFK is to first determine the optimal subspace dimension and construct geodesic lines, then calculate the geodesic flow kernel, and finally transform the source domain data into target domain. TCA and JDA project the source-domain data and target-domain data into a subspace, so the gesture recognizer is trained on the projected source-domain data and predicts the projected target-domain data. Since GFK transforms the source-domain data into the target domain, we train the recognizer on the transformed source-domain data and use it to make predictions for the target domain. Noteworthily, cross-subject setting is a special case of leave-one-user-out cross validation (in leave-one-use-out scheme, data from one user is taken as the test set and data from other users form the training set). If there are only two users, cross-subject evaluation equals leave-one-user-out cross validation. Since leave-one-user-out cross validation could use data from multiple users to better reflect the characteristics of a test user, cross-subject setting is a more challenging case.

4.2 Results of feature selection

To determine the optimal number of finally selected features for the three feature selectors, we first rank all the features in descending order and then choose the top-k features as the selected ones. Specifically, we take 15, 20, and 25 as the candidates of k for the sEMG sensor and the accelerometer. For the fusion of sEMG sensor and accelerometer, we take 25, 40, and 50 as the candidates. The one achieving the best accuracy is used in the following study. Figures 5, 6, and 7 present the experimental results of sEMG sensor, accelerometer, and the fusion of sEMG sensor and accelerometer, respectively, where "w/o" denotes the results without using feature selection. We can see that feature selection indeed influences the accuracy of a gesture recognizer and that the optimal number of features for different types of sensors may be different. For example, we observe from Fig. 5 that reliefF generally achieves stable accuracy and performs better than CFS and MRMR. For reliefF, the use of top 25 features is a better choice.

Similar results can be observed in Fig. 6. From Fig. 7, we can observe that the use of 50 is a good choice. Since reliefF works better in the majority of cases, we use it to optimize the feature space.

After determining the finally selected features, we present the gesture recognition results of different types of sensors in Table 3, where the first column indicates the sensors and the last row "Both" denotes the results of using both the sEMG sensor and accelerometer. The results are organized by the used classification model and the best accuracy and F1 are shown in bold. From Table 3, we can observe that the use of random forest outperforms its competitors. For example, if the sEMG sensor is used, RF gets 98.96% accuracy, compared to 97.38% of SVM, 75.13% of NB, 79.22% of LR, and 97.64% of KNN. This is mainly because random forest inherits the merits of ensemble learning to learn a robust recognizer with better generalization. Second, we observe that the use of sEMG sensor tends to obtain higher accuracy than the use of the accelerometer. For example, SVM obtains 97.38% accuracy with the sEMG sensor and only gets 86.65% accuracy with the accelerometer. This partially indicates that the sEMG sensor better captures hand movements. Third, we can observe that the joint use of sEMG sensor and accelerometer generally works better than their single use. For example, SVM gets 98.69% accuracy with the two sensors, compared to the 97.38% accuracy of sEMG sensor, and 86.65% accuracy of accelerometer.

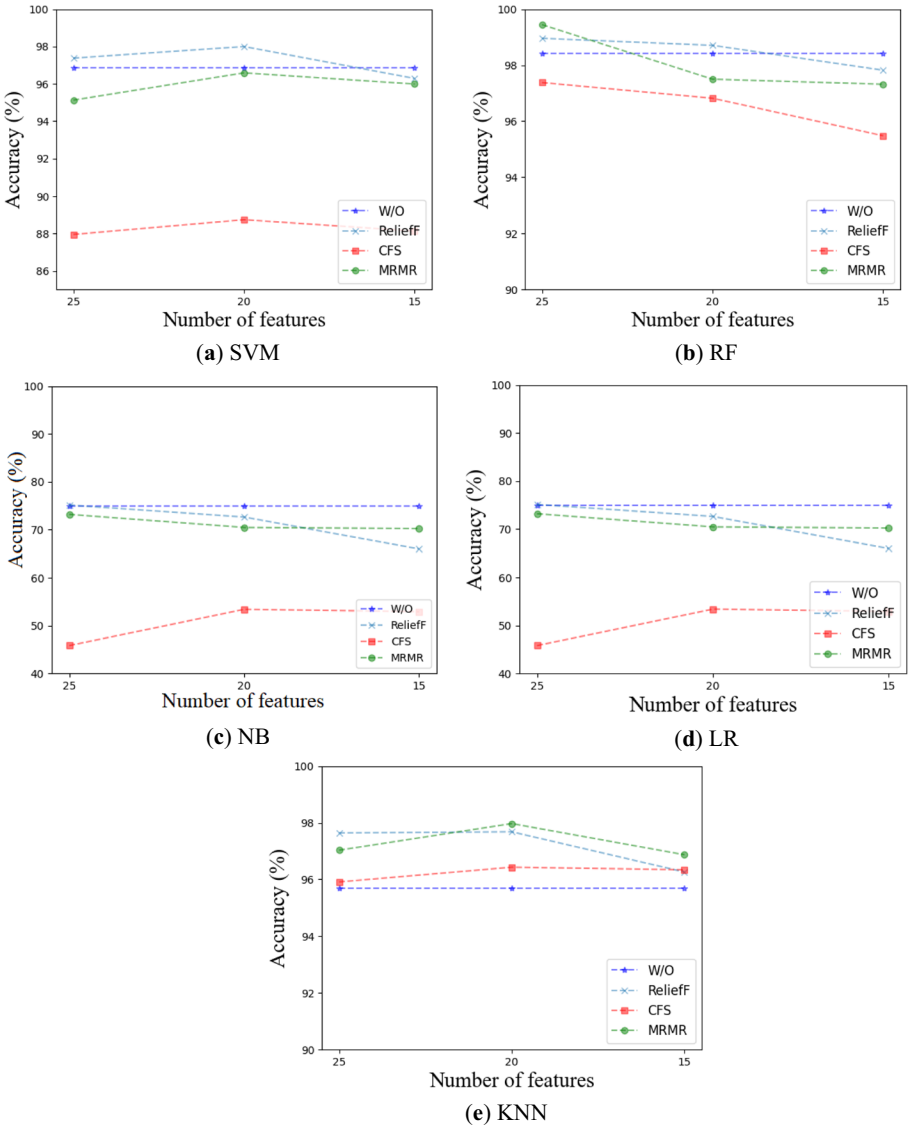


Fig. 5 Feature selection for the sEMG sensor

This is mainly because the accelerometer and sEMG sensor could provide complementary information to each other, which demonstrates the effectiveness of using different types of sensors in designing a gesture recognizer.

Furthermore, Fig. 8 shows the confusion matrix obtained using RF to study the recognition errors between gestures, whose columns (rows) denote the predicted (actual) labels and numbers correspond to the gestures ({1: *open*, 2: *turn up*, 3: *turn down*, 4: *fist*, 5: *null gesture*}). 10 samples of *turn up* are misclassified into *open* for the sEMG sensor, 69 *turn down* samples are misclassified as *open* for the accelerometer, and

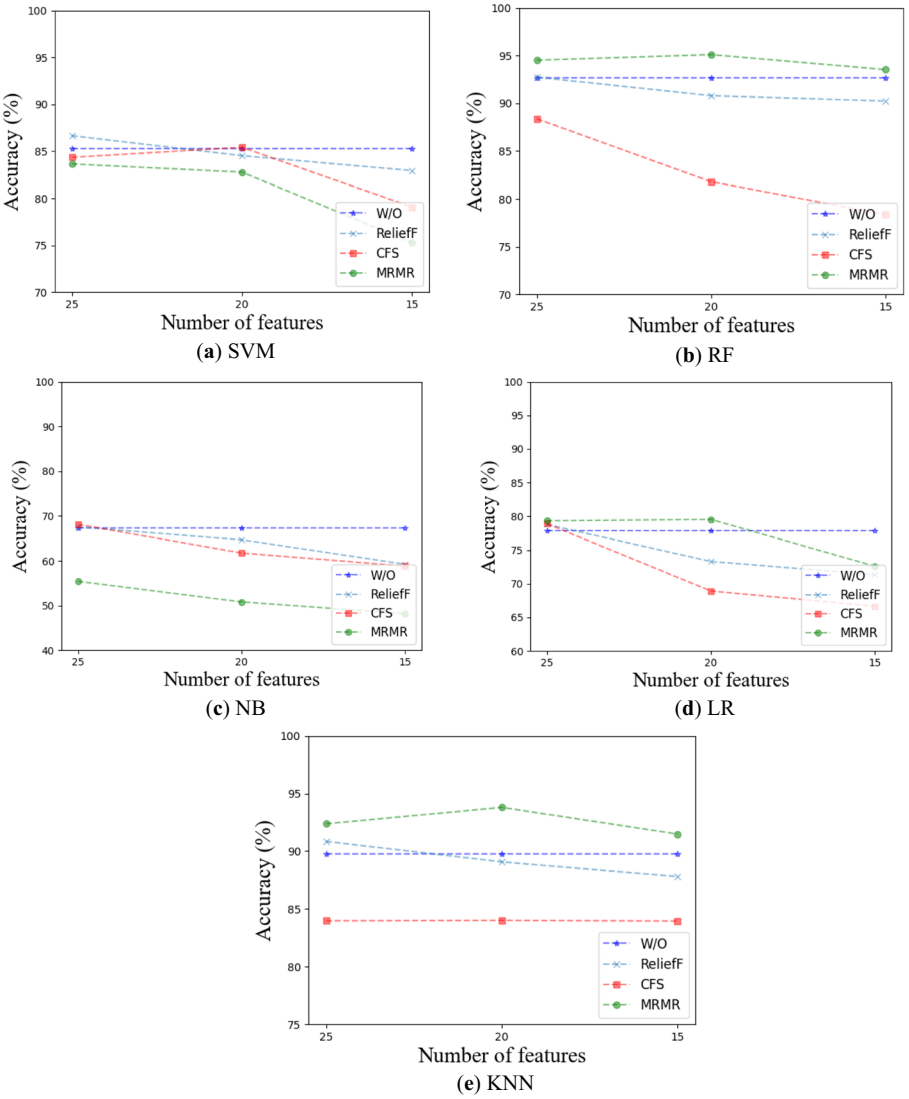


Fig. 6 Feature selection for the accelerometer

the use of sEMG sensor and accelerometer reduces the number of errors to 4 and 5, respectively.

Furthermore, we experimentally investigate the role of different channels of the sEMG sensor and different axes of the accelerometer. Table 4 shows the results of sEMG sensor. The first column shows the used features, where TD (FD) denotes time-domain (frequency-domain) features and TFD indicates the concatenation of TD and FD. We organize the results by the used classification model. The best accuracy and F1 in each group are shown in bold and the best F1 and accuracy for each type features are underlined.

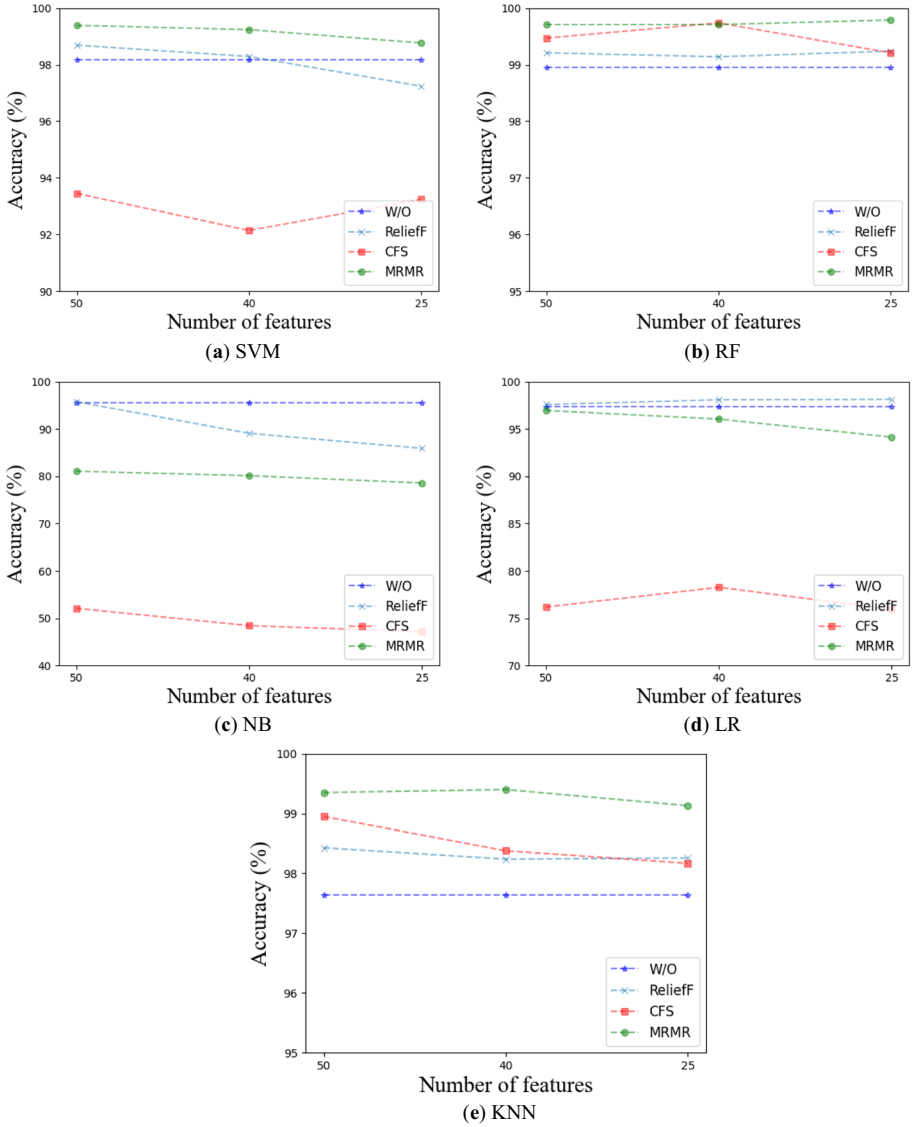


Fig. 7 Feature selection results for the two sensors

From Table 4, we observe that the use of RF outperforms its competitors with TD, FD, as well as TFD. For example, RF achieves 98.53% accuracy in the time domain, compared to the 95.29% of SVM, 58.12% of NB, 81.94% of LR, and 97.66% of KNN. This indicates the superiority of RF in building a gesture recognizer. Second, we obtain mixed results in comparing the use of time-domain and frequency-domain features. For example, RF obtains 98.35% accuracy with TD and 94.50% accuracy with FD, while the use of FD performs better than TD with NB. Third, we can see that the joint use of time-domain and frequency-domain features generally obtains better results than the single use of TD

Table 3 Results of different sensors

	SVM				RF				NB				LR				KNN			
	Acc	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1
sEMG	97.38	97.31	97.25	97.28	98.96	98.91	98.99	98.95	75.13	79.22	74.70	75.44	90.05	90.12	89.87	90.04	97.64	97.83	97.36	97.56
Accelerometer	86.65	88.28	87.95	88.08	92.74	93.47	93.18	93.32	67.54	70.33	69.75	69.93	78.80	80.74	80.84	80.73	90.84	92.13	91.65	91.79
Both	**98.69**	98.80	98.63	**98.77**	**99.21**	99.28	99.22	**99.25**	**95.76**	95.68	95.71	**95.70**	**97.58**	97.57	97.62	**97.59**	**98.43**	98.73	98.27	**98.47**

	1	2	3	4	5
1	**748**	8	0	0	1
2	10	**643**	0	0	2
3	0	0	**792**	6	1
4	0	0	0	**625**	3
5	3	0	0	0	**438**

(a) sEMG sensor

	1	2	3	4	5
1	**697**	0	54	6	0
2	3	**623**	19	10	0
3	69	5	**716**	0	9
4	10	25	15	**578**	0
5	0	0	5	8	**428**

(b) accelerometer

	1	2	3	4	5
1	**752**	2	3	0	0
2	4	**648**	1	2	0
3	5	0	**792**	2	0
4	0	0	4	**624**	0
5	0	0	0	3	**438**

(c) sEMG sensor and accelerometer

Fig. 8 Confusion matrices obtained using RF and different sensors

and FD. For example, with SVM, 95.29% accuracy is obtained using time-domain features and 91.62% accuracy is obtained with frequency-domain features, while the use of features from both domains gets 97.38% accuracy. This indicates that complementary information exists between time-domain and frequency-domain features.

Table 5 shows the results of accelerometer, where X-TFD, Y-TFD, and Z-TFD denote the time-domain and frequency-domain features extracted from X-axis, Y-axis, and Z-axis of a 3-axis accelerometer, respectively. Also, XYZ-TFD is the concatenation of X-TFD, Y-TFD, and Z-TFD. The rlt-TFD is the time-domain and frequency-domain features extracted from the resultant axis.

From Table 5, we can observe that the use of RF outperforms its competitors in all cases of different types of features (i.e., X-TFD, Y-TFD, Z-TFD, XYZ-TFD, and rlt-TFD), which indicates the superiority of RF over SVM, NB, LR, and KNN. For example, the RF gets 68.06% accuracy with X-TFD, compared to the 63.61% of SVM, 45.03% of NB, 55.24% of LR, and 67.54% of KNN. Second, we observe that the concatenation of features of the three axes performs better than the features of the resultant axis. For example, SVM obtains 86.65% accuracy for the concatenation of features and only gets 76.18% accuracy for the resultant axis. This is possibly because loss of information occurs when we only use the resultant axis to extract features. Third, we observe that the joint use of features of three axes obtains better accuracy than their single use, which indicates that different axes contain complementary information in better discriminating gestures. For example, in terms of RF, 92.74% accuracy is obtained with XYZ-TFD, compared to the 68.06% of X-TFD, 81.15% of Y-TFD, and 81.94% Z-TFD.

4.3 Results of cross-subject experiments

For the typical gesture recognition procedure, we train a recognizer on the collected data and then use it to predict gestures. Due to the inter-subject variation, however, we often suffer from degraded accuracy if we directly apply a recognizer that is trained for one user to the other user. Figure 8 shows the experimental results where we train a gesture recognizer on user S1 and use it to predict gestures for users S1, S2, S3, and S4. In Fig. 9, the first group of results correspond to the accuracy of S1 (i.e., *subject-dependent* prediction) and the other three groups are cross-subject recognition results (i.e., *cross-subject* prediction). We can observe that the recognition accuracy is greater than 98.0% for S1. However, the accuracy decreases significantly for all the five models, if we apply the recognizer to S2, S3, and S4. For example, the accuracy of RF decreases from 98.31% to 58.10% for S2. This indicates the difference between different subjects in performing (even the same) gestures.

Table 4 Experimental results using sEMG sensor

	SVM				RF				NB				LR				KNN			
	Acc	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1
TD	95.29	95.25	95.47	95.33	98.53	98.63	98.96	98.75	58.12	69.26	53.55	56.49	81.94	82.51	81.27	81.91	97.64	97.66	96.85	97.20
FD	91.62	91.97	91.11	91.47	94.50	94.43	93.14	94.30	71.99	81.01	70.47	76.43	82.46	83.14	82.75	82.83	93.46	93.52	93.12	93.30
TFD	**97.38**	97.31	97.25	**97.28**	**98.96**	98.91	98.99	**98.95**	**75.13**	79.22	74.70	**75.44**	**90.05**	90.12	89.87	**90.04**	**97.91**	97.83	97.36	**97.56**

Table 5 Experimental results using accelerometer

	SVM				RF				NB				LR				KNN			
	Acc	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1
X-TFD	63.61	66.18	65.17	65.45	68.06	71.43	71.11	71.33	45.03	53.22	45.31	48.80	55.24	55.81	56.65	55.99	67.54	70.25	69.64	69.92
Y-TFD	71.20	74.12	73.55	73.76	81.15	82.91	80.61	82.39	49.48	55.97	49.32	52.63	62.57	63.46	63.10	63.23	77.23	80.20	78.72	79.32
Z-TFD	74.35	76.45	76.33	76.40	81.94	84.51	84.17	84.27	57.33	67.57	57.90	62.86	68.06	70.98	70.06	70.29	78.80	80.36	80.11	80.20
XYZ-TFD	**86.65**	88.28	87.95	**88.08**	**92.74**	92.83	92.26	**92.64**	**67.54**	70.33	69.75	**69.83**	**78.80**	80.74	80.84	**80.78**	**90.84**	92.13	91.65	**91.79**
rlt-TFD	76.18	79.65	78.09	78.57	80.10	83.29	82.04	82.40	42.41	54.20	42.21	48.60	70.16	73.43	72.26	72.54	78.80	81.43	81.00	81.17

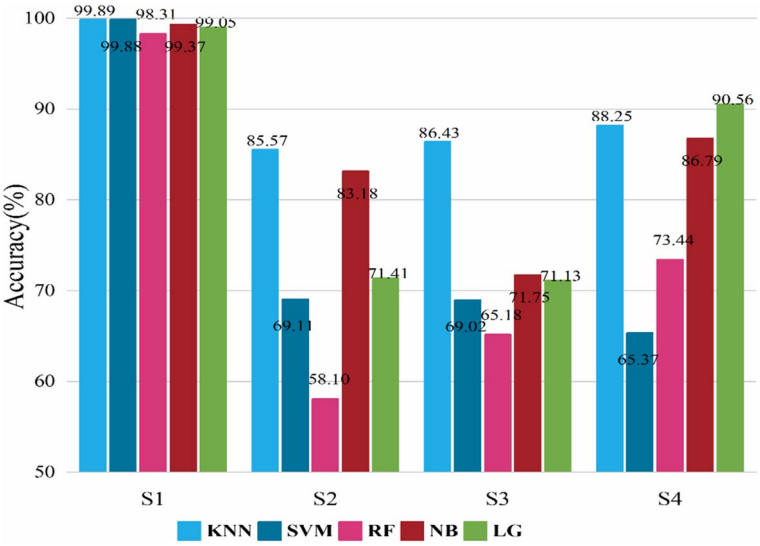


Fig. 9 Recognition accuracy for different subjects with the recognizer trained on S1

Clearly, one solution to mitigate the problem is to collect a large amount of labeled data for each new user. However, the collection of data is time-consuming and often impossible, which limits its practical use. The other solution is to reuse data from other users with transfer learning techniques. Accordingly, we explore three different transfer learning algorithms (i.e., TCA, JDA, and GFK) and present their results in Tables 6, 7, and 8 for the case of sEMG sensor, accelerometer, and the fusion of sEMG sensor and accelerometer, respectively. The best accuracy and F1 are shown in bold and the symbol "A = > B" in the tables denotes that data from subject A (i.e., source domain) is used to

Table 6 Experimental results using the sEMG sensor

	W/O		TCA		JDA		GFK	
Task	*acc*	*F*1	*acc*	*F*1	*acc*	*F*1	*acc*	*F*1
S1 = > S2	76.76	74.34	68.84	66.23	**78.01**	**76.49**	74.64	75.79
S1 = > S3	54.06	47.97	52.35	43.57	**58.28**	**53.54**	51.40	52.02
S1 = > S4	58.51	45.28	56.78	45.57	56.53	43.65	**59.45**	**55.26**
S2 = > S1	**78.06**	**76.26**	69.42	61.92	75.25	75.69	74.47	74.28
S2 = > S3	56.20	52.63	52.14	49.77	**60.97**	**59.84**	54.87	56.48
S2 = > S4	69.40	66.75	68.19	66.83	**73.89**	**75.44**	66.52	65.55
S3 = > S1	48.89	46.68	48.15	40.22	**50.56**	**49.84**	48.90	48.95
S3 = > S2	44.96	45.19	**55.68**	**56.89**	52.36	55.05	48.62	52.14
S3 = > S4	70.61	69.76	71.82	73.70	**76.42**	**76.75**	72.46	75.74
S4 = > S1	56.51	42.09	49.52	37.07	58.04	43.42	**59.70**	**56.85**
S4 = > S2	63.35	56.63	**71.39**	**66.20**	69.58	63.81	60.87	59.44
S4 = > S3	83.65	83.40	78.31	78.26	**85.79**	**84.79**	79.49	79.40

Table 7 Experimental results using the accelerometer

	W/O		TCA		JDA		GFK	
Task	Acc	F1	Acc	F1	Acc	F1	Acc	F1
S1 = > S2	33.97	32.66	**50.45**	**46.83**	46.32	44.73	41.94	43.84
S1 = > S3	68.70	70.33	63.35	62.36	**69.46**	**70.71**	64.20	67.68
S1 = > S4	65.77	62.09	70.27	72.94	**71.17**	**71.50**	65.08	66.92
S2 = > S1	45.40	34.23	**57.57**	**53.88**	50.01	42.31	46.76	45.53
S2 = > S3	47.01	43.82	**61.86**	**61.68**	52.99	51.51	52.13	53.94
S2 = > S4	21.18	25.02	22.3	31.19	**25.91**	**31.99**	23.50	27.60
S3 = > S1	63.60	60.52	63.17	59.06	**69.68**	**66.89**	62.65	63.36
S3 = > S2	56.32	49.47	**58.37**	**56.22**	57.15	51.42	51.81	48.67
S3 = > S4	64.91	66.03	68.37	71.44	**71.50**	**73.78**	60.20	62.30
S4 = > S1	45.61	41.16	**56.4**	**53.77**	48.10	45.26	53.66	55.91
S4 = > S2	31.16	26.32	**46.36**	**43.56**	34.10	31.17	31.53	30.18
S4 = > S3	50.21	52.61	54.17	55.35	**56.77**	**58.98**	56.69	58.97

Table 8 Experimental results using both the sEMG sensor and accelerometer

	W/O		TCA		JDA		GFK	
Task	Acc	F1	Acc	F1	Acc	F1	Acc	F1
S1 = > S2	85.57	83.98	86.08	84.9	**91.12**	**90.52**	77.32	79.91
S1 = > S3	86.43	86.90	87.71	88.09	**96.05**	**96.26**	87.12	88.38
S1 = > S4	88.25	86.84	79.43	77.64	**97.71**	**97.89**	86.53	87.12
S2 = > S1	80.21	76.54	76.72	73.15	**86.76**	**84.65**	79.23	79.68
S2 = > S3	73.72	70.12	**85.68**	**85.85**	81.38	81.62	75.61	78.33
S2 = > S4	71.05	70.21	**81.76**	**81.84**	77.23	78.09	68.89	71.18
S3 = > S1	86.24	87.56	86.24	86.62	**94.25**	**94.46**	84.67	85.90
S3 = > S2	86.33	85.94	**96.93**	**96.85**	95.31	95.32	88.97	88.82
S3 = > S4	89.97	90.95	90.23	91.28	**97.23**	**97.47**	90.77	92.25
S4 = > S1	87.51	87.06	91.96	92.88	**95.99**	**96.21**	88.56	88.56
S4 = > S2	66.28	55.65	**86.59**	**85.43**	79.91	80.14	68.81	67.59
S4 = > S3	93.80	93.99	95.51	95.56	**98.29**	**98.35**	92.63	92.71

train a recognizer to infer gestures of subject B (i.e., target domain). For the purpose of comparison, we also present the results, denoted by W/O, without using transfer learning. From Table 6, we observe that the use of transfer learning helps to obtain higher accuracy for all cases except S2 = > S1. For example, the use of JDA improves the accuracy to 78.01% for S1 = > S2. Compared with TCA and GFK, JDA performs better in the majority of cases. From Table 7, we can observe that JDA improves the accuracy compared to the case of without using transfer learning. For example, for S1 = > S2, JDA improves the accuracy to 46.32% from 33.97%; and JDA improves the accuracy from 68.70% to

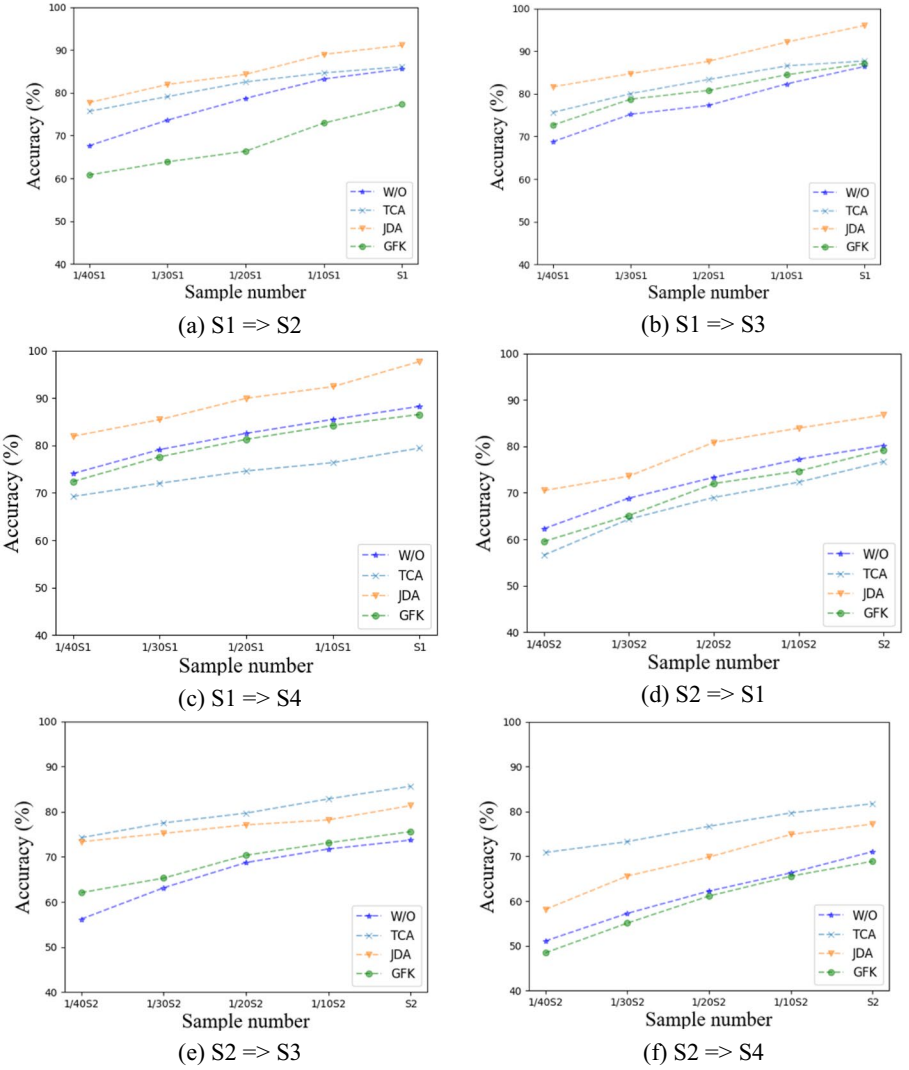
69.46% for S1 = > S3. TCA and GFK have mixed accuracy in some cases. For example, for S1 = > S2, TCA obtains the accuracy of 50.45% and GFK gets 41.94% accuracy, compared to the 33.97% accuracy that is obtained without transfer learning; for S1 = > S3, the accuracy without using transfer learning is 68.70%, while TCA and GFK obtain the accuracy of 63.35% and 64.20%, respectively. This indicates the superiority of JDA over TCA and GFK. Similar results can be observed from Table 8. We observe from Tables 6, 7, and 8 that the use of accelerometer and sEMG sensor generally obtains better accuracy than the single use of accelerometer and sEMG sensor in the cross-subject setting, which indicates the synthetic effects of different sensors. For example, for S1 = > S2, JDA gets accuracy of 78.01%, 46.32%, and 91.12% for the single use of accelerometer and sEMG sensor, and their joint use, respectively.

Furthermore, we vary the percentage of the used source-domain data to evaluate its influence on the gesture recognizer. Figure 10 presents the results, where the Y-axis denotes accuracy and X-axis refers to the percentage of source-domain data (e.g., 1 corresponds to all the source-domain data, and 1/10 means ten percent of the source-domain data). From Fig. 10, we observe a general trend that the accuracy increases with the increase of the percentages. This is mainly because more data help reduce the distribution discrepancy between source and target domains. Second, compared with TCA and GFK, JDA generally performs better and hence remains a priority.

4.4 Evaluation of null gesture

We in this subsection experimentally evaluate the impact of null gesture on a recognizer. Table 9 presents the results without using null gesture, where the best F1 and accuracy are shown in bold for each classifier. Each row corresponds to the used sensors and the last row is the joint use of accelerometer and sEMG sensor. We can see from Table 9 that the joint use of accelerometer and sEMG sensor tends to obtain higher accuracy than their single use, which is consistent with previous results and indicates the complementarity of different types of sensors. For example, in the case of SVM, the use of two types of sensors gets accuracy of 99.09%, compared to the 98.18% of sEMG and 87.27% of accelerometer; in the case of KNN, we obtain 98.77% F1 with the two types of sensors, compared to the 92.21% F1 of accelerometer and 98.46% F1 of sEMG sensor. From the results of Table 3 (considering null gesture) and Table 9 (without considering null gesture), we can observe that the inclusion of null gesture generally lowers the accuracy of a recognizer. For example, SVM gets the accuracy of 97.38%, 86.65%, and 98.69% for sEMG, accelerometer, and the two types of sensors, respectively, if null gesture is considered; and the accuracy increase to 98.18%, 87.27%, and 99.09%, respectively, if null gesture is not considered.

In addition, we evaluate the null gesture in the cross-subject setting. Figure 11 shows the comparative results, where the blue bar (left bar) denotes the results considering null gesture and the red bar (right bar) shows its comparative results. We can observe that there is a general trend of better recognition accuracy if null gesture is not considered. This, to a certain extent, indicates the similarity between null gesture and the predefined gestures and further inspires us of taking into account null gesture in developing and implementing a practical gesture recognizer.



(a) S1 => S2 (b) S1 => S3

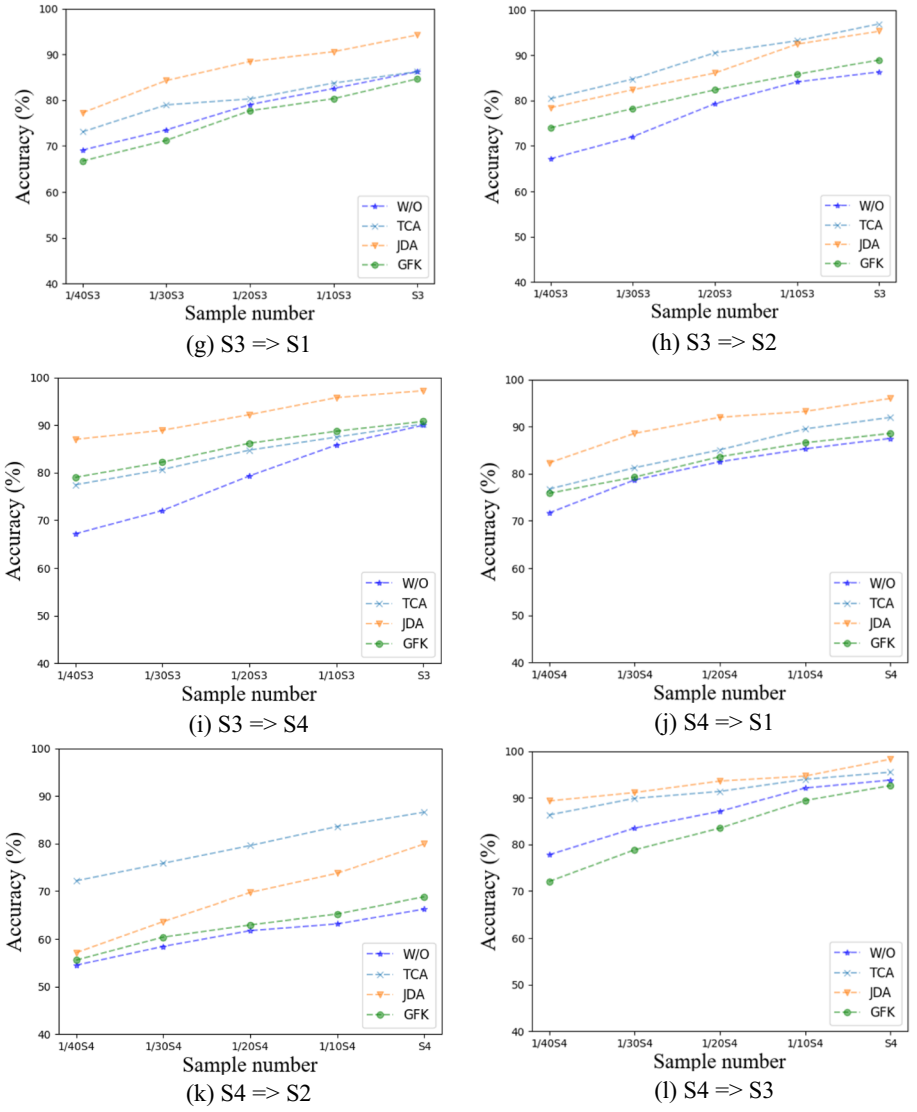
(c) S1 => S4 (d) S2 => S1

(e) S2 => S3 (f) S2 => S4

Fig. 10 Transfer learning results versus different sizes of source-domain data

5 Conclusion

Gesture recognition has a crucial role in a variety of pervasive computing scenarios, where the choice of sensing units, the extraction and selection of informative features, and the transfer of knowledge to new users are three key components in developing a practical gesture recognizer. In this study, we propose a gesture recognition model based on two different types of sensors (i.e., accelerometer and sEMG sensor) that can capture the kinematic and physiological signals of hand movements. First, we consider the features extracted from different axes of accelerometer and different channels of sEMG sensor. Three powerful feature selectors are explored to optimize the feature space and to find a subset of



(g) S3 => S1 (h) S3 => S2

(i) S3 => S4 (j) S4 => S1

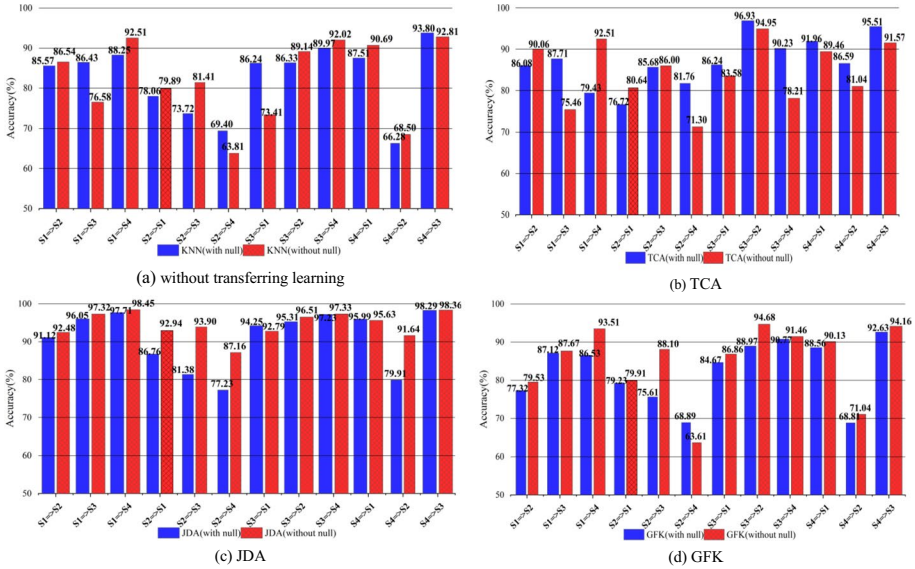
(k) S4 => S2 (l) S4 => S3

Fig. 10 (continued)

discriminant features. Second, to handle the scenario where no labeled data are available for new users to train a gesture recognizer, we utilize and evaluate three transfer learning techniques to reuse training data of other users in the cross-subject setting. Third, extensive comparative experimental results show that the joint use of sEMG sensor and accelerometer generally outperforms their single use, indicating the synthetic effect of different types of sensors, and that transfer learning improves the accuracy of a gesture recognizer in the cross-subject setting and JDA tends to perform better than TCA and GFK. In addition, we experimentally evaluate the impact of null gesture on a gesture recognizer and results show that null gesture generally lowers the accuracy.

Table 9 Experimental results of different sensors without null gesture

	SVM				RF				NB				LR				KNN			
	Acc	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1	*Acc*	*Pre*	*Rec*	*F*1
sEMG	98.18	98.30	98.04	98.16	99.09	99.34	99.32	99.33	76.97	77.61	76.27	76.46	90.30	90.46	89.95	90.15	98.48	98.49	98.45	98.46
Accelerometer	87.27	88.38	87.84	88.00	93.72	94.25	93.40	93.53	66.47	67.67	66.59	66.86	77.88	78.31	78.27	78.28	91.82	92.29	92.16	92.21
Both	**99.09**	99.18	98.96	**99.07**	**99.39**	99.34	98.97	**99.29**	**95.81**	96.03	95.82	**95.93**	**97.89**	97.93	97.89	**97.90**	**98.79**	98.91	98.63	**98.77**



(a) without transferring learning (b) TCA (c) JDA (d) GFK

Fig. 11 Performance comparisons between with and without null gesture in the cross-subject setting

One limitation of the study is the feature engineering, which is often a tedious and knowledge-driven procedure and largely rely on the domain knowledge. Representation learning techniques have the capacity of automatically and effectively learning high-level latent features from the raw sensor signals [53], so how to integrate deep learning models into the gesture recognition chain to mine the time and spatial dependency deserves further study. Since deep learning models usually require high computing power, this requires us to develop light-weight model to enable the deployment of edge devices. Besides, we plan to conduct future work along with the following aspects. First, combined sEMG and accelerometer signals essentially belongs to the multi-view learning paradigm. There probably exist high-level interaction among the signals, other advanced multi-view learning algorithms could be explored to better capture the relationship between sEMG and accelerometer signals, where we should consider their different sampling rates. Second, although experiments, especially the transfer learning ones, conducted on the collected datasets achieve the objectives of this study and demonstrate our points, experiments on a larger number of participants and more datasets, such as those mentioned in [11], would be expected to gain deeper insights and to provide a broader perspective on the research questions. Third, we only demonstrate the power of single-source transfer learning in obtaining improved cross-subject gesture recognition accuracy. Could we obtain higher accuracy, if multi-source transfer learning techniques are used? How many sources should be then used and chosen? This also deserves further study.

Acknowledgements The authors would like to express sincere gratitude to the reviewers.

Author contributions Conceptualization: Aiguo Wang, Huancheng Liu, Chundi Zheng, Huihui Chen; Methodology, Aiguo Wang, Huancheng Liu, Chih-Yung Chang; Formal analysis, Aiguo Wang, Chih-Yung Chang; Writing—original draft preparation, Huancheng Liu; Writing—review and editing, Aiguo Wang, Chih-Yung Chang. All authors have read and agreed to the published version of the manuscript.

Funding This work was partially supported by the National Natural Science Foundation of China (Grant No. 62176082) and the Featured Innovation Project of the Department of Education of Guangdong Province (Grant No. 2021KTSCX117).

Data availability The data of this study are available from the corresponding author upon reasonable request.

Declarations

Institutional review board statement Not applicable.

Informed consent Not applicable.

Conflicts of interest The authors declare no conflict of interest.

References

1. Jia L, Zhou X, Xue C (2022) Non-trajectory-based gesture recognition in human-computer interaction based on hand skeleton data. Multimed Tools Appl 81(15):20509–20539
2. Gadekallu TR, Srivastava G, Liyanage M, Iyapparaja M, Chowdhary CL, Koppu S, Maddikunta PR (2022) Hand gesture recognition based on a Harris hawks optimized convolution neural network. Computers Electrical Engin 100:107836
3. Bhushan S, Alshehri M, Keshta I, Chakraverti AK, Rajpurohit J, Abugabah A (2022) An experimental analysis of various machine learning algorithms for hand gesture recognition. Electronics 11(6):968
4. Zhang Z, Tian Z, Zhou M (2018) Latern: Dynamic continuous hand gesture recognition using FMCW radar sensor. IEEE Sensors J 18(8):3278–3289
5. Qi J, Jiang G, Li G, Sun Y, Tao B (2019) Intelligent human-computer interaction based on surface EMG gesture recognition. IEEE Access 7:61378–61387
6. Guarino A, Malandrino D, Zaccagnino R, Capo C, Lettieri N (2023) Touchscreen gestures as images. A transfer learning approach for soft biometric traits recognition. Expert Syst Appl 219:119614
7. Kim E, Helal S, Nugent C, Beattie M (2015) Analyzing activity recognition uncertainties in smart home environments. ACM Trans Intelli Syst Tech 6(4):1–28
8. Wang A, Chen G, Wu X, Liu X, An N, Chang CY (2018) Towards human activity recognition: A hierarchical feature selection framework. Sensors 18(11):3629
9. Bai L, Yao L, Wang X, Kanhere SS, Guo B, Yu Z (2020), Adversarial multi-view networks for activity recognition. Proc ACM Interactive, Mob, Wearable Ubiquit Technol 4(2):1-22
10. Lima WS, Braganca HL, Souto EJ (2021) NOHAR: Novelty discrete data stream for human activity recognition based on smartphones with inertial sensors. Expert Syst Appl 166:114093
11. Jaramillo-Yánez A, Benalcázar M, Mena-Maldonado E (2020) Real-time hand gesture recognition using surface electromyography and machine learning: A systematic literature review. Sensors 20(9):2467
12. Guo L, Lu Z, Yao L (2021) Human-machine interaction sensing technology based on hand gesture recognition: A review. IEEE Trans Human-Mach Syst 51(4):300–309
13. Si Y, Chen S, Li M, Li S, Pei Y, Guo X (2022) Flexible strain sensors for wearable hand gesture recognition: From devices to systems. Adv Intell Syst 4(2):2100046
14. Al Farid F, Hashim N, Abdullah J, Bhuiyan MR, Shahida Mohd Isa WN, Uddin J, Haque MA, Husen MN (2022) A structured and methodological review on vision-based hand gesture recognition system. J Imaging 8(6):153
15. Bhaumik G, Verma M, Govil M, Vipparthi S (2023) Hyfinet: Hybrid feature attention network for hand gesture recognition. Multimed Tools Appl 82(4):4863–4882
16. Dhiman C, Vishwakarma D (2019) A robust framework for abnormal human action recognition using R-transform and Zernike moments in depth videos. IEEE Sensors J 19(3):5195–5203
17. Nyo MT, Mebarek-Oudina F, Hlaing SS, Khan NA (2022) Otsu's thresholding technique for MRI image brain tumor segmentation. Multimed Tools Appl 81(30):43837–43849
18. Pagan J, Fallahzadeh R, Pedram M, Jl R-M, Moya JM, Ayala JL, Ghasemzadeh H (2019) Toward ultra-low-power remote health monitoring: An optimal and adaptive compressed sensing framework for activity recognition. IEEE Trans Mobile Comput 18(3):658–673

19. Chen L, Hoey J, Nugent CD, Cook DJ, Yu Z (2012) Sensor-based activity recognition. IEEE Trans Syst Man, and Cybernetics, Part C (Applications and Reviews) 42(6):790–808
20. Zhao H, Xu R, Shu M, Hu J (2016) Physiological-signal-based key negotiation protocols for body sensor networks: A survey. Simul Model Pract Theory 65:32–44
21. Chen G, Wang W, Wang Z et al (2020) Two-dimensional discrete feature based spatial attention capsnet for sEMG signal recognition. Appl Intell 50:3503–3520
22. Sayin FS, Ozen S, Baspinar U (2018) Hand gesture recognition by using sEMG signals for human machine interaction applications. In: 2018 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA). IEEE, pp 27–30
23. Zou W, Zhang D, Lee DJ (2022) A new multi-feature fusion based convolutional neural network for facial expression recognition. Appl Intell 52(3):2918–2929
24. Saeedi R, Sasani K, Norgaard S, Gebremedhin AH (2018) Personalized human activity recognition using wearables: a manifold learning-based knowledge transfer. In: 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, pp 1193–1196
25. Ren Z, Yuan J, Meng J, Zhang Z (2013) Robust part-based hand gesture recognition using Kinect sensor. IEEE Trans Multimed 15(5):1110–1120
26. Song T, Zhao H, Liu Z, Liu H, Sun D (2021) Intelligent human hand gesture recognition by local–global fusing quality-aware features. Future Gener Comp Syst 115(7043):298–303
27. Chen FS, Fu CM, Huang CL (2003) Hand gesture recognition using a real-time tracking method and hidden Markov models. Image Vision Comput 21(8):745–758
28. Wang A, Wu X, Zhao L, Chen H, Zhao S (2021) Physical activity recognition from accelerometer data using multi-view aggregation. J Appl Sci Engineer 24(4):611–620
29. Xia Q, Wada A, Korpela J, Maekawa T, Namioka Y (2019) Unsupervised factory activity recognition with wearable sensors using process instruction information. Proc ACM Interactive, Mob, Wearable Ubiquit Technol 3(2):1-23
30. Jiang W, Ye X, Chen R, Su F, Lin M, Ma Y, Huang S (2021) Wearable on-device deep learning system for hand gesture recognition based on FPGA accelerator. Math Biosci Eng 18(1):132–153
31. Wang A, Chen G, Yang J, Zhao S, Chang CY (2016) A comparative study on human activity recognition using inertial sensors in a smartphone. IEEE Sensors J 16(11):4566–4578
32. Sapienza S, Ros PM, Guzman D, Rossi F, Terracciano R, Cordedda E, Demarchi D (2018) On-line event-driven hand gesture recognition based on surface electromyographic signals. In: 2018 IEEE international symposium on circuits and systems (ISCAS). IEEE, pp 1–5
33. Yang Y, Duan F, Liu Z, Zhu C (2018) A time-domain hand gesture recognition system based on two-channel sEMG signals. In: Proceedings of the 12th International Convention on Rehabilitation Engineering and Assistive Technology. ICREAT, pp 31–36
34. Zhang Y, Chen Y, Yu H, Yang X, Zeng B (2021) A feature adaptive learning method for high-density sEMG-based gesture recognition. Proc ACM Interactive, Mob, Wearable Ubiquit Technol 5(1):1–26
35. Dardas NH, Georganas ND (2011) Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques. IEEE Trans Instrumen Meas 60(11):3592–3607
36. Ceolini E, Frenkel C, Shrestha SB, Taverni G, Khacef L, Payvand M, Donati E (2020) Hand-gesture recognition based on EMG and eventbased camera sensor fusion: A benchmark in neuromorphic computing. Front Neurosci 14:637
37. Lu Z, Chen X, Li Q, Zhang X, Zhou P (2014) A hand gesture recognition framework and wearable gesture-based interaction prototype for mobile devices. IEEE Trans Human-Mach Syst 44(2):293–299
38. Jiang S, Lv B, Guo W, Zhang C, Wang H, Sheng X, Shull PB (2018) Feasibility of wrist-worn, real-time hand, and surface gesture recognition via sEMG and IMU sensing. IEEE Trans Ind Inf 14(8):3376–3385
39. Guo X, Xu W, Tang WQ, Wen C (2019) Research on optimization of static gesture recognition based on convolution neural network. In: 2019 4th International Conference on Mechanical, Control and Computer Engineering (ICMCCE). IEEE, pp 398–3982
40. Sun JH, Ji TT, Zhang SB, Yang JK, Ji GR (2018) Research on the hand gesture recognition based on deep learning. In: 2018 12th International Symposium on Antennas, Propagation and EM Theory (ISAPE). IEEE, pp 1–4
41. Lee HK, Kim JH (1999) An HMM-based threshold model approach for gesture recognition. IEEE Trans Pattern Analy Mache Intelli 21(10):961–973

42. Bargellesi N, Carletti M, Cenedese A, Susto G, Terzi M (2019) A random forest-based approach for hand gesture recognition with wireless wearable motion capture sensors. IFAC-PapersOnLine 52(11):128–133
43. Pomboza-Junez G, Terriza JH (2016) Hand gesture recognition based on sEMG signals using support vector machines. In: 2016 IEEE 6th International Conference on Consumer Electronics-Berlin (ICCE-Berlin). IEEE, pp 174–178
44. Yu Z, Zhao J, Wang Y, He L, Wang S (2021) Surface EMG-based instantaneous hand gesture recognition using convolutional neural network with the transfer learning method. Sensors 21(7):2540
45. Wang J, Chen Y, Hu L, Peng X, Philip SY (2018) Stratified transfer learning for cross-domain activity recognition. In: 2018 IEEE international conference on pervasive computing and communications (PerCom). IEEE, pp 1–10
46. Côté-Allard U, Fall CL, Campeau-Lecours A, Gosselin C, Laviolette F, Gosselin B (2017) Transfer learning for sEMG hand gestures recognition using convolutional neural networks. In: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE, pp 1663–1668
47. Colacino FM, Emiliano R, Mace BR (2012) Subject-specific musculoskeletal parameters of wrist flexors and extensors estimated by an emg-driven musculoskeletal model. Med Eng Phys 34(5):531–540
48. Banos O, Galvez JM, Damas M, Pomares H, Rojas I (2014) Window size impact in human activity recognition. Sensors 14(4):6474–6499
49. Wang A, Meng Y, Liu J, Zhao S, Chen G (2021) Multi-domain feature extraction for human activity recognition using wearable sensors. In: 2021 International Conference on Networking and Network Applications (NaNA). IEEE, pp 254–259
50. Pan SP, Tsang IW, Kwok JT, Qiang Y (2010) Domain adaptation via transfer component analysis. IEEE Trans Neural Networks 22(2):199–210
51. Long M, Wang J, Ding G, Sun J, Yu PS (2013) Transfer feature learning with joint distribution adaptation. In: Proceedings of the IEEE international conference on computer vision. IEEE, pp 2200–2207
52. Gong B, Shi Y, Sha F, Grauman K (2012) Geodesic flow kernel for unsupervised domain adaptation. In: 2012 IEEE conference on computer vision and pattern recognition. IEEE, pp 2066–2073
53. Chen G, Wang A, Zhao S, Liu L, Chang CY (2018) Latent feature learning for activity recognition using simple sensors in smart homes. Multimed Tools Appl 77:15201–15219