Smart Health 36 (2025) 100574

Contents lists available at ScienceDirect

Smart Health

journal homepage: www.elsevier.com/locate/smhl

Quantum contrastive learning for human activity recognition

Yanhui Ren [a], Di Wang [b], Lingling An [b], Shiwen Mao [c], Xuyu Wang [d,*]

[a] Guangzhou Institute of Technology, Xidian University, Guangzhou, China
[b] School of Computer Science and Technology, Xidian University, Xi'an, China
[c] Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849, USA
[d] Knight Foundation School of Computing and Information Sciences, Florida International University, Miami, FL 33199, USA

ARTICLE INFO

Keywords:
Human activity recognition
Contrastive learning
Quantum machine learning
Quantum neural networks

ABSTRACT

Deep learning techniques have been widely used for human activity recognition (HAR) applications. The major challenge lies in obtaining high-quality, large-scale labeled sensor datasets. However, unlike datasets such as images or text, HAR sensor datasets are non-intuitive and uninterpretable, making manual labeling extremely difficult. Self-supervised learning has emerged to address this problem, which can learn from large-scale unlabeled datasets that are easier to collect. Nevertheless, self-supervised learning has the increased computational cost and the demand for larger deep neural networks. Recently, quantum machine learning has attracted widespread attention due to its powerful computational capability and feature extraction ability. In this paper, we aim to address this classical hardware bottleneck using quantum machine learning techniques. We propose QCLHAR, a quantum contrastive learning framework for HAR, which combines quantum machine learning techniques with contrastive learning to learn better latent representations. We evaluate the feasibility of the proposed framework on six publicly available datasets for HAR. The experimental results demonstrate the effectiveness of the framework for HAR, which can surpass or match the precision of classical contrastive learning with fewer parameters. This validates the effectiveness of our approach and demonstrates the significant potential of quantum technology in addressing the challenges associated with the scarcity of labeled sensory data.

1. Introduction

Human activity recognition (HAR) aims to accurately infer human behaviors based on signals collected from wearable sensor devices. It has a wide range of applications such as augmented reality (AR), virtual reality (VR), smart homes, and smart health. Over the last few years, deep neural networks have been proven to be effective in the HAR domain, achieving high classification accuracy (Wang, Chen, Hao, Peng, & Hu, 2019; Yao et al., 2018). However, these well-performing models typically require a vast amount of labeled data for fully supervised learning. Moreover, with the advancement of deep learning technology, the foundational models have become deeper. This has further intensified the demand for large-scale annotated data. Compared to visual data modalities such as text and images, annotating sensor signal data directly is particularly challenging. Due to its non-intuitive and uninterpretable nature, the process of labeling sensor data is susceptible to human bias, which can lead to annotation ambiguities. Such annotation is both expensive and time-consuming. Therefore, even though models can achieve satisfactory results, their reliance

* Corresponding author.
E-mail addresses: yanhuir@stu.xidian.edu.cn (Y. Ren), wangdi@xidian.edu.cn (D. Wang), an.lingling@gmail.com (L. An), smao@ieee.org (S. Mao), xuywang@fiu.edu (X. Wang).

https://doi.org/10.1016/j.smhl.2025.100574
Received 3 March 2025; Accepted 15 March 2025
Available online 24 March 2025

on large amounts of labeled data makes it difficult to scale. Additionally, a reduction in the number of labeled samples leads to a decreased performance, highlighting the persistent challenges in wearable-based tasks.

Based on the above challenges and the readily available unlabeled sensor datasets, self-supervised learning has emerged, which can generate its own supervisory signals by leveraging the inherent structure of the data and mitigate the need for manual labeling. Numerous solutions have been proposed in fields such as image classification and natural language processing that utilize self-supervised learning to address the issue of limited data labels (Gidaris, Singh, & Komodakis, 2018; Wei, Lim, Zisserman, & Freeman, 2018). Among these, contrastive learning in self-supervised learning has become a particularly promising technique, and the accuracy in downstream tasks sometimes exceeds that of fully supervised learning. For example, Chen, Kornblith, Norouzi, and Hinton (2020) design a simple visual representation contrastive learning framework (SimCLR), which can achieve great performance results on ImageNet. By using different transformed views of the same sample as positive pairs and views from different samples as negative pairs, a feature extractor is trained, which is insensitive to transformations.

Although contrastive learning can utilize unlabeled data for training and address the issue of limited labeled data, achieving performance comparable to supervised learning (Chen et al., 2020), it still faces some challenges. First, training with self-supervised learning on unlabeled data is more difficult, resulting in high computational complexity (Wang, Wang, Wang, Torr, & Lin, 2021). Second, the computational power increases with the amount of data and increases at an overwhelming rate. To capture more complex correlations between augmented views, a large amount of training data is needed, along with increased time cost and larger network requirements (Henaff, 2020). Thus, the development of conventional hardware becomes a significant bottleneck for the purpose of achieving superior performance.

Currently, the emerging quantum technologies, with their powerful computational capabilities, may potentially assist in realizing the demands of large neural networks. Quantum Machine Learning (QML) theoretically has the potential to be faster than classical machine learning algorithms due to its unique characteristics (i.e., quantum entanglement and quantum superposition) that classical machine learning does not have. The principle of quantum superposition allows Quantum Neural Networks (QNNs) to have enhanced parallel processing capabilities, handling larger datasets; with n quantum bits, one can process 2^n binary numbers simultaneously. Early proposed quantum algorithms had fully demonstrated the computational power of quantum computing systems (Grover, 1996; Shor, 1994). As a result, QML emerges as a potential solution to the challenges posed by classical hardware. In this paper, we aim to explore the use of QNNs with their powerful computational abilities to mitigate the hardware issues encountered in self-supervised learning. The existing challenges lie in how to effectively utilize QNNs to alleviate this issue and what kind of quantum circuits to design. It is crucial to ensure that while addressing these challenges, we do not undermine the objectives of self-supervised learning. At the same time, the appropriate design can also save certain resources. However, due to the limited size of quantum circuits, the performance of large-scale quantum models remains challenging to validate. In this paper, our objective is to explore the effectiveness of quantum techniques in contrastive learning for HAR applications.

To mitigate the limitation of classical hardware resources and address the challenge of sensor data being difficult to label, we incorporate quantum circuits into the contrastive learning framework to implement QCLHAR, a novel quantum contrastive learning framework for HAR. Specifically, we design a variational quantum circuit (VQC) to serve as a projection head. The choice of quantum circuits as the projection head is motivated by the potential to capture complex data relationships that classical circuits might overlook, and its capability to effectively represent and handle high-dimensional data spaces. In the first stage, combined with an encoder consisting of a fully convolutional neural network, contrastive learning training is leveraged together to acquire feature information. In the second stage, we utilize the learned parameter information to extract features and train a classifier by supervised learning. By leveraging this hybrid classical–quantum model, we address the challenge of limited labeled data in the sensor domain and mitigate the classical bottleneck with fewer parameters. In both the classical space and quantum space, different augmented views of the same data sample are pulled together, while those of different classes are pushed apart. Experiments reveal that the quantum contrastive learning can outperform classical models on unseen downstream task data while reducing the number of parameters.

The main contributions of this paper are summarized as follows.

- We propose QCLHAR, a novel quantum contrastive learning framework for HAR. This approach effectively combines quantum advantages, thus reducing the model's parameters and computational complexity. To the best of our knowledge, this is the first quantum contrastive learning framework for HAR. This approach would motivate subsequent research. Additionally, our code is publicly available at anonymous.4open.science/r/AF62.
- We design a variational quantum circuit as a quantum projection head to enhance data compression and information representation by leveraging quantum properties. This enables contrastive learning to extract more complex features that classical methods may overlook. Additionally, we develop importance weighting for sampling from imbalanced datasets.
- We evaluate QCLHAR on six HAR datasets, demonstrating the feasibility of the proposed model. The superior performance compared to classical model has been validated with a decreased number of parameters, thereby reducing the complexity of the model.

The rest of the paper is organized as follows. Section 2 introduces the proposed quantum contrastive learning framework. In Section 3, we focus on the experimental setup. In Section 4, we conduct extensive experiments to evaluate the performance of QCLHAR on six datasets. Section 5, we discuss in detail various quantum projection head schemes and analyze the influence of different quantum depths. Finally, we conclude this paper in Section 6.

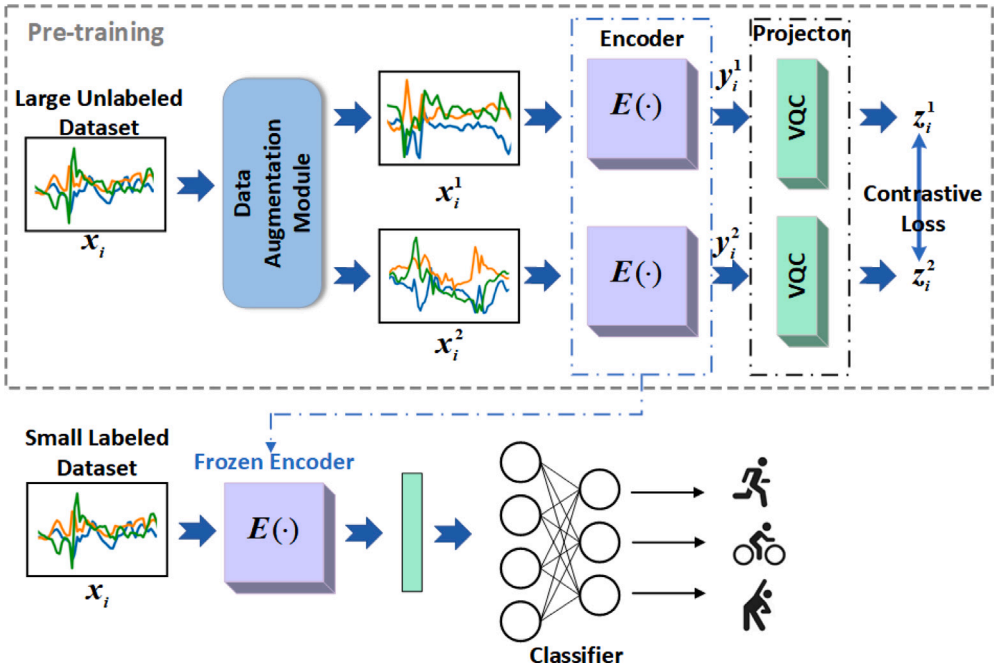

Fig. 1. Overview of the proposed approach QCLHAR. Different data augmentations are applied to the same unlabeled data instance x_i, resulting in two augmented data samples x_i^1 and x_i^2. A classical encoder network $E(\cdot)$ and a quantum projection head VQC are trained to perform contrastive learning. The encoder parameters that are learned through this process are subsequently utilized to train a classifier on labeled data for downstream classification tasks.

2. Quantum contrastive learning framework

In this section, we introduce QCLHAR, our quantum contrastive learning framework for HAR. First, we provide an overview of the proposed method. Next, we explain the details of the quantum projection head. Finally, we offer a detailed description of the network architecture and its specific implementation.

2.1. Overview

We introduce the quantum contrastive learning into HAR in wearable devices. The overall objective is to leverage easily collected unlabeled data for self-supervised pre-training, aiming to learn effective sensor data representations for downstream HAR tasks. The detailed workflow is illustrated in Fig. 1 and comprises two stages: *(i) pre-training*, where unlabeled data are utilized for contrastive learning, generating its own supervisory signal. This training phase primarily consists of four components: the data augmentation module, the encoder, the quantum projection head, and the contrastive loss function. *(ii) fine-tuning*, where labeled data are employed to train a classifier for activity recognition.

During the pre-training process, we use a large volume of unlabeled sensor data $\{x_i\}_{i=1}^N$, where N represents the total number of samples, as input for contrastive learning. First, the input undergoes data augmentation, resulting in two augmented samples, x_i^1 and x_i^2, to enhance the robustness and generalization capability of the model. Second, for each input sample x_i, our objective is to train an encoder $E(\cdot)$ based on a classical neural network, aiming to extract meaningful high-dimensional feature representations $y_i = E(x_i)$, from the raw data. To further optimize the framework of contrastive learning, we have designed a quantum projection head $P(\cdot)$. Its function is to map the features extracted by the encoder to a unified latent space, which can be denoted as $z_i = P(y_i)$. Finally, to ensure that the distance between the latent representations of positive sample pairs is as minimal as possible, while maximizing the distance from negative sample pairs, we employ the normalized temperature-scaled cross entropy loss (NT-Xent) as the contrastive loss function. For the latent representations (z_i^1, z_i^2) of a positive sample pair, the loss function can be expressed as:

$$L_i = -\log \frac{exp(sim(z_i^1, z_i^2)/\tau)}{exp(sim(z_i^1, z_i^2)/\tau) + \sum_{k=1}^{2B} 1_{[k\neq i]} exp(sim(z_i^1, z_k)/\tau)}, \tag{1}$$

where B is the given mini-batch, and τ is the temperature parameter used to adjust the contrast ratio between positive and negative samples. $1_{[k\neq i]} \in \{0, 1\}$ is an indicator function

$$1_{[k\neq i]} = \begin{cases} 1 & \text{for } k \neq i, \\ 0 & \text{for } k = i. \end{cases} \tag{2}$$

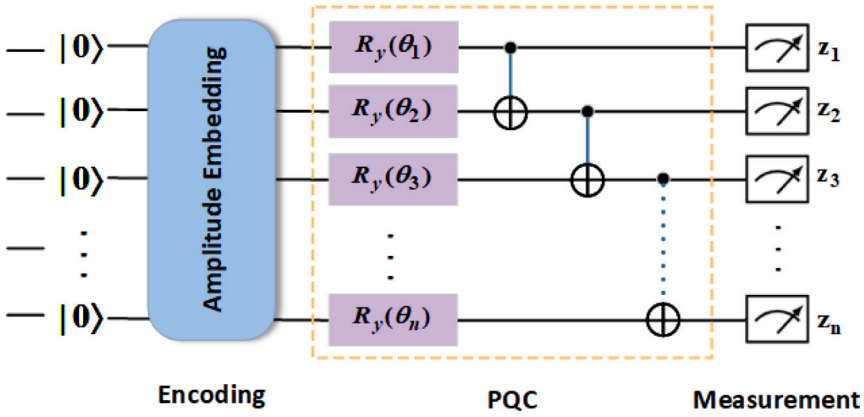

Fig. 2. The structure of the proposed quantum projection head. The circuits in the dash square correspond to the learnable variational quantum circuit layer with depth D.

We use cosine similarity to measure the similarity between two augmented views. For two latent representations z_i^1 and z_i^2, the similarity is calculated as:

$$sim(z_i^1, z_i^2) = \frac{\langle z_i^1 | z_i^2 \rangle}{\left\| |z_i^1\rangle \right\|_2 \left\| |z_i^2\rangle \right\|_2}, \tag{3}$$

where $\langle z_i^1 | z_i^2 \rangle$ denotes the quantum state inner product. Using Eq. (1) to calculate the loss for all positive sample pairs, and the average value is taken as the final loss function.

In downstream supervised learning tasks, we freeze the encoder and discard the projection head. Using the frozen learning weights, we extract features from the labeled training data $\{x_i, Y\}_{i=1}^N$ and train a specialized classification network $f(\cdot)$. This network maps features to specific class labels. The performance obtained directly indicates the quality of the representations learned during the pre-training stage.

2.2. Quantum projection head for contrastive learning

In contrastive learning, the presence of the projection head is important. It maps the original features obtained from the encoder network to a feature space that is more beneficial to contrastive learning, thereby enhancing the model's ability to discriminate between positive and negative samples. In SimCLR (Chen et al., 2020), it has been demonstrated that applying contrastive loss on the mapped feature space z_i is more effective than on the original features y_i. The introduction of the projection head helps to highlight invariant features from latent representations, further amplifying the network's ability to recognize different augmented views of the same sample. However, in traditional contrastive learning, the projection head is primarily composed of linear layers, which often requires a large number of parameters and computational burden. To address this problem, we attempt to explore the possibility of using variational quantum circuits(VQC) as the projection head. The design idea of this quantum projection head is based on the principle of quantum mechanics, aiming to leverage quantum properties to realize dimensionality reduction and feature extraction techniques for high-dimensional data. By utilizing quantum characteristics such as superposition and entanglement, the quantum projection head can represent and process high-dimensional and complex data more efficiently, preserving more information while compressing the data.

The proposed quantum projection head framework is shown in Fig. 2, which consists of three parts: *(i) quantum encoding*, *(ii) parameterized quantum circuit (PQC)*, and *(iii) measurement*. Algorithm 2 describes the execution process of the quantum projection head we proposed. To preserve as much feature information as possible, we employ amplitude encoding to convert classical IMU data into a form that can be input into a quantum system. The core concept is to map classical data onto the amplitudes of qubits, leveraging the superposition property of quantum states to represent different classical information. In amplitude encoding, a quantum bit $|\psi\rangle$ can be represented as a superposition of $|0\rangle$ and $|1\rangle$ states: $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where α and β are complex numbers representing the amplitudes. The probabilities of measuring the qubit in states $|0\rangle$ and $|1\rangle$ are given by $|\alpha|^2$ and $|\beta|^2$ respectively, satisfying the normalization condition $|\alpha|^2 + |\beta|^2 = 1$. By adjusting the amplitudes α and β, classical information can be encoded into the amplitudes of quantum states. The quantum state of the input data can be represented as $|v\rangle = \otimes_{j=1}^n \left|v_j\right\rangle = |v_1\rangle \otimes |v_2\rangle \otimes \cdots \otimes |v_n\rangle$, where n is the number of quantum states. The encoding map prepares each quantum bit into a balanced superposition of $|0\rangle$ and $|1\rangle$.

After obtaining the quantum state, we employ a PQC to learn the linear transformations between quantum states. Although more quantum gate operations can lead to a more complex and powerful model, they simultaneously introduce greater noise and errors. Considering that the primary goal of the projection head is to reduce the dimensionality of the features extracted by the

encoder while retaining as much feature information as possible, a highly complex model structure is unnecessary (as demonstrated by experiments in Section 5.3). Using amplitude encoding, we can map high-dimensional features to low-dimensional features while preserving maximum feature information. We then choose a simple circuit structure with rotation gates and controlled-NOT (CNOT) gates to demonstrate the potential of quantum computing in HAR tasks and to validate the performance improvements and advantages by using quantum computing.

Initially, an R_y rotation gate, denoted as $R_y(\theta_{j,d}), j \in [1,n], d = [1,D]$, is applied to each quantum bit for feature mapping, allowing the quantum bit to rotate from its ground state to a specific quantum state, where θ represents a trainable angle parameter, and D denotes the depth of the PQC. The quantum state's transformation can be adjusted according to the learnable parameter θ. The initial state of a qubit can be represented as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. After passing through the rotation gate $R_y(\theta) = \begin{pmatrix} cos(\theta/2) & -sin(\theta/2) \\ sin(\theta/2) & cos(\theta/2) \end{pmatrix}$, the qubit performs matrix multiplication, resulting in the rotated qubit state as follows: $|\psi'\rangle = (cos(\theta/2)\alpha - sin(\theta/2)\beta)|0\rangle + (sin(\theta/2)\alpha + cos(\theta/2)\beta)|1\rangle$. This interaction between the weight and input data in quantum computing is similar to that in classical machine learning. Through this operation, the input data is mapped to a more complex feature space that is easier to classify. Subsequently, we employ CNOT gates to enhance the entanglement between quantum states. One quantum bit serves as the control bit and the other as the target bit. When the control bit is set to one, the target bit is flipped. This entanglement operation captures complex correlations between input data features, which is a challenge for classical computers.

After passing through several PQC layers, we use the Pauli-Z operator to measure the expected values of each quantum bit. The Pauli-Z operator is a significant operator in quantum mechanics used to measure the Z-direction spin of a qubit (i.e., the states 0 and 1 in quantum computation), which can be expressed as $\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$. In quantum computation, for a system composed of multiple qubits, the Pauli-Z operator is employed to measure the state information of individual qubits. For the other qubits, their states remain unchanged. They are equivalent to the identity matrix under the action of the Pauli-Z operator and are not be affected. Specifically, for a given quantum state $|v\rangle$, the expected observation is denoted as $\left|z_j\right\rangle = \left\langle v|z_j|v\right\rangle$, where z_j represents the Pauli-Z operator acting on the jth quantum bit, and the identity operator I is applied to the other quantum bits. By measuring the expected value of the Pauli-Z operator for each qubit, we can obtain the state information for each qubit. These information $\left|z_1\right\rangle, \left|z_2\right\rangle, \ldots, \left|z_n\right\rangle$ are used to compute the contrastive loss.

2.3. Network architecture and implementation

In the previous section, we provide a detailed explanation of the implementation of our quantum projection head. In this section, we describe the integration of contrastive learning with the projection head for the training of QCLHAR model. Our comprehensive training process is summarized in Algorithm 1.

Algorithm 1 Quantum Contrastive Learning Pre-training

Input: Unlabeled Dataset $\{x_i\}_{i=1}^{N}$, $B, n, \tau, E(\cdot;\phi), P(\cdot;\theta)$;

1: **for** each mini-batch $\{x_i\}_{i=1}^{B}$ **do**
2: **for** $i \in \{1, \ldots, B\}$ **do**
3: $x_i^1, x_i^2 \leftarrow \text{aug1}(x_i), \text{aug2}(x_i)$; ▷ Two augmentations
4: $y_i^1, y_i^2 \leftarrow E(x_i^1;\phi), E(x_i^2;\phi)$; ▷ Encoder
5: **for** each representation y_i^1 and y_i^2 **do**
6: Use Algorithm 2 to obtain the vector representation of quantum projection z_i^1 and z_i^2;
7: **end for**
8: **end for**
9: **for** $i \in \{1, \ldots, B\}$ and $k \in \{1, \ldots, B\}$ **do**
10: $sim(z_i^1, z_i^2) \leftarrow \frac{\langle z_i^1 | z_i^2 \rangle}{\||z_i^1\rangle\|_2 \||z_i^2\rangle\|_2}$; ▷ Similarity
11: $L_i \leftarrow -\log \frac{\exp(sim(z_i^1, z_i^2)/\tau)}{\exp(sim(z_i^1, z_i^2)/\tau) + \sum_{k=1}^{2B} 1_{[k \neq i]} \exp(sim(z_i^1, z_k)/\tau)}$;
12: **end for**
13: $L = \sum_{i=1}^{B} L_i$;
14: Update networks to minimize L;
15: **end for**
return: encoder network $E(\cdot;\phi)$ and freeze, throw away $P(\cdot;\theta)$;

For each sample in every mini-batch, we execute the training steps from lines 2 to 8 to generate a pair of vector representations. These representations are subsequently used to compute the contrastive loss. Each data point undergoes the data augmentation methods in line 3 to obtain two augmented views of the input signal. Considering that the primary emphasis of this paper is not on data augmentation, and it is time-consuming to implement quantum circuits, we only employ two data augmentation techniques for our experiments. The first is the resampling data augmentation technique proposed by Wang et al. (2022), which simulates real

activity data by changing the sampling frequency to maximize the coverage of the sampling space. This involves two specific steps: upsampling and downsampling. Upsampling is the process of fitting new values along the sensor data time axis using interpolation; downsampling involves filtering values through random or periodic sampling to restore the original sample length. The second is the "negated" method described by Saeed, Ozcelebi, and Lukkien (2019), which results in a vertical flip or mirror of the input signal. Following data augmentation, we implement a fully convolutional neural network as the encoder $E(\cdot)$, aiming to extract meaningful feature representations from the augmented views. This encoder consists of four 1D convolutional layers, with channel counts of 32, 64, 128, and 256, respectively. The kernel size for each convolutional layer is 8 with a stride of 1. Following each convolutional layer, we integrate batch normalization, the non-linearity ReLU activation function, and max pooling. Additionally, to prevent overfitting, we introduce a dropout strategy with p = 0.35 after the first convolutional layer. It should be noted that the decision to set the final channel count to 256 is to better facilitate the execution of the quantum projection head operation in Algorithm 2.

Algorithm 2 Quantum Projection Head

Input: Representations y_i^1 and y_i^2 where $i \in \{1, ..., B\}$, D, θ;
1: **for** each representation **do**
2: Use AmplitudeEmbedding to map y_i^1 and y_i^2 onto the 8 qubits;
3: **for** each y_i^1 and y_i^2 **do**
4: **for** d in D **do**
5: **for** j from 0 to $n-1$ **do**
6: Apply $R_y(\theta_{j,d})$ gate on qubit j;
7: **end for**
8: **for** $j = 0$ to $n-2$ **do**
9: Apply CNOT gate to increase entanglement between adjacent qubits;
10: **end for**
11: **end for**
12: Measure the expected value of Pauli-Z on each qubit;
13: **end for**
14: **end for**
return: Quantum projected vectors: $|z_1\rangle, |z_2\rangle ..., |z_n\rangle$;

For each 256-dimensional classical feature vector, we implement the amplitude encoding strategy in line 2 of Algorithm 2, mapping it onto 8 qubits to form its quantum state representation. For each quantum state representation, we execute the parameterized quantum circuit from lines 4 to 11. A R_y rotation gate is applied to each qubit for feature mapping, and CNOT gates are used to increase entanglement between adjacent qubits. These quantum gate operations are repeated for D layers. After completing the circuit operations, the Pauli-Z operator is performed on each qubit as described in line 12 to obtain the expected measurement values. Ultimately, all generated quantum projection vectors are returned for use in the subsequent computational steps of Algorithm 1.

We utilize the acquired feature vector representations for the loss computation of each positive sample pair in lines 9–12. Then, the average contrastive loss across all samples in the mini-batch is computed as the overall loss, which is minimized to update the network parameters.

Following pre-training, we transfer the learned encoder weights to the activity recognition network for learning the final classification task. Fig. 1 depicts this proposed framework. The prediction classifier network $f(\cdot)$ is a linear layer specifically designed for classification. At this stage, we optimize its parameters using the cross-entropy loss function.

3. Experiment setup

In the previous section, we have introduced our QCLHAR framework. This framework is designed to address the problem of limited labeled sensory data and the bottleneck of classical hardware. In this section, we provide the used dataset as well as the specific data processing and experimental settings.

3.1. Datasets

Our evaluation primarily focuses on HAR datasets with motion sensors (3-axis accelerometer and 3-axis gyroscope). In addition, we use USC-HAD dataset as an example of an activity recognition scenario that does not use smartphones. We consider six publicly available HAR datasets (i.e., UCI-HAR Anguita et al., 2013, SHAR Micucci, Mobilio, & Napoletano, 2017, HHAR Stisen et al., 2015, Motionsense Malekzadeh, Clegg, Cavallaro, & Haddadi, 2018, USC-HAD Zhang & Sawchuk, 2012 and MobiAct Chatzaki, Pediaditis, Vavoulas, & Tsiknakis, 2017).

3.2. Data preparation and system implementation

In the training process of machine learning and deep learning, the imbalance of data distribution has a significant impact on the training results of the model. To address this issue, we adopt *a weighted random sampling strategy* to adjust the probability of each category sample being chosen. For each sample, we set the weight to be proportional to the inverse of the number of samples in each category, ensuring that categories with fewer samples receive higher weights. We apply preprocessing on the signals such as normalization and sliding window segmentation on all datasets. For the datasets UCI-HAR, SHAR, HHAR, MotionSense, and USC-HAD, we segment their signals into sliding window sizes of 128, 151, 100, 400, and 250 time steps, respectively, with a 50% overlap. Then, we randomly divide the datasets for training, validation, and testing, with split proportions of 64%, 16%, and 20%, respectively. Finally, we remove the labels from these datasets and generate our own self-supervised signals through the process described in Section 2.

Furthermore, due to the domains of 10 out of 30 participants in the SHAR dataset contain incomplete classes, we disregard the data from these 10 participants. For the HHAR dataset, due to the large sample size and the time-consuming nature of quantum circuits, we only use data collected from phone devices in this paper and down-sample the readings to 50 Hz to reduce computational load.

For the VQC, we set the number of quantum bits to 8 and encode the 256-dimensional classical data features into quantum states through amplitude encoding. Since the projection head does not require a very complex structure, and too deep circuit would introduce hardware costs and a lot of noise, we set the depth of the PQC to 1. In the quantum projection head circuit we designed, only 8 learnable parameters are needed. We implement the VQC using Pennylane as a torch layer, while the rest of the model is implemented using PyTorch.

Unless specified differently, the temperature value τ in the contrastive loss function defaults to 0.1, and the batch size is 128. We update the parameters by using the Adam optimizer with a cosine anneal decay schedule. The learning rates for the first and second phases are set to 3e−3 and 1e−1, respectively. The SHAR dataset for the first stage is set to 1e−2. The model is trained for a total of 120 epochs, and we choose the model with the lowest loss during the validation process in the pre-training phase as the best model for fine-tuning in the second phase.

We evaluate our model using test accuracy and F1-score. The test accuracy represents the percentage of the number of correctly classified samples over the total number of tested samples, which is calculated by:

$$Test\ Accuracy = \frac{N_{correct}}{N_{total}}. \tag{4}$$

The F1-score is the harmonic mean of precision and recall:

$$F_1 = 2\frac{Precision \times Recall}{Precision + Recall}. \tag{5}$$

4. Performance evaluation

In this section, we conduct numerous experiments to evaluate the performance of the proposed framework and verify its effectiveness.

4.1. Comparison against baseline algorithms

In this subsection, we evaluate our quantum contrastive learning technique with quantum projection head against other classical contrastive learning methods and a fully-supervised approach.

Our primary comparison baseline is the classical contrastive learning method SimCLR (Chen et al., 2020). Based on SimCLR, we implement our QCLHAR framework. Apart from the projection head, the rest is consistent with SimCLR. Comparing our proposed method with SimCLR helps readers to better understand the performance of our method and provides a meaningful benchmark. By evaluating the extent of improvement of our proposed method relative to SimCLR, we can better reveal the effectiveness of our improvements. We also report results from other classical contrastive learning frameworks, including BYOL (Grill et al., 2020), SimSiam (Chen & He, 2021), NNCLR (Dwibedi, Aytar, Tompson, Sermanet, & Zisserman, 2021), and TS-TCC (Eldele et al., 2021). In addition, we evaluate a supervised learning method and the quantum self-supervised learning approach QSSL (Jaderberg et al., 2022), as baselines.

For all the self-supervised learning methods in our study, we use the same data augmentation strategies: resample and negated. Except for QSSL, the encoders utilized in our experiments are based on the same four-layer fully convolutional neural network. Different contrastive learning frameworks have various projection heads. However, the primary components of the projection heads in the baselines we employed are linear layers. For example, SimCLR's projection head is a MLP with one hidden layer. After the hidden layer, there is a non-linearity activation unit ReLU. Similar to SimCLR, BYOL uses two linear layers as the projection head. The difference is that BYOL adds batch normalization after each linear layer. In supervised learning, we use labeled data to train both the feature extractor and the classifier simultaneously, optimizing the model using the cross-entropy loss. Additionally, we introduce QSSL, which is the first to combine quantum machine learning with self-supervised learning, as another baseline. The authors of QSSL designed the encoder that replacing the final layer of ResNet network with a quantum circuit. Therefore, we implement their idea by replacing the last convolutional layer with a quantum circuit, serving as our adaptation of the QSSL encoder.

Table 1
On UCI-HAR, SHAR, HHAR, MotionSense, USC-HAD and MobiAct datasets, we compare the proposed QCLHAR framework to the classical contrastive learning framework and supervised method for performance comparison. The F1-score (%) is used as the primary evaluation metric.

Datasets	UCI-HAR	SHAR	HHAR	MotionSense	USC-HAD	MobiAct
Supervised	96.12	94.64	96.13	99.02	91.95	99.79
BYOL	91.60	**87.40**	92.54	94.34	77.47	96.84
SimSiam	54.95	67.13	48.74	85.22	55.60	92.68
NNCLR	88.93	84.60	93.21	93.82	88.51	96.37
TS-TCC	93.20	86.12	93.75	97.59	90.04	98.08
QSSL	80.05	54.66	52.38	51.43	61.66	75.66
SimCLR	94.08	83.26	92.22	93.67	88.05	96.80
SimCLR(MLP)	93.79	85.94	94.29	94.95	87.55	97.28
QCLHAR	**94.13**	86.18	**94.83**	**98.19**	**91.66**	**99.07**

Table 2
Activity categories corresponding to the numbers on the axes of the confusion matrix.

Datasets	Activity classes corresponding to the numbers
UCI-HAR	0: walking, 1: walking_upstairs, 2: walking_downstairs, 3: sitting, 4: standing, 5: lying
SHAR	0: standing up from sitting, 1: standing up from laying, 2: walking, 3: running, 4: walking upstairs, 5: jumping, 6: walking downstairs, 7: lying down to standing, 8: sitting down, 9–16: eight types of falls
HHAR	0: biking, 1: sitting, 2: standing, 3: walking, 4: walking upstairs, 5: walking downstairs
MotionSense	0: sitting, 1: standing, 2: walking, 3: walking upstairs, 4: walking downstairs, 5: jogging
USC-HAD	0: walking forward, 1: walking left, 2: walking right, 3: walking upstairs, 4: walking downstairs, 5: running forward, 6: jumping, 7: sitting, 8: standing, 9: sleeping, 10: elevator up, 11: elevator down
MobiAct	0: standing, 1: walking, 2: jogging, 3: jumping, 4: stairs up, 5: Stairs down, 6: stand to sit, 7: sitting on chair, 8: sit to stand, 9: car step-in, 10: car step-out

Table 1 shows the results by our proposed and other training algorithms on the six target datasets (UCI-HAR, SHAR, HHAR, MotionSense, USC-HAD and MobiAct). The primary evaluation metric employed is the F1-score. As report in this table, our proposed method outperforms all self-supervised learning approaches in five out of six datasets. Compared to the primary baseline SimCLR, we observe significant improvements in some of the datasets, including SHAR, MotionSense, USC-HAD and MobiAct. In particular, our proposed framework achieves a 99.07% F1-score on the large-scale activity recognition dataset MobiAct. For the SHAR dataset, our method does not surpass the classical contrastive learning framework BYOL. However, in discussion, we demonstrate that introducing QML as a projection head in other schemes will outperform BYOL.

In particular, since the SimCLR framework achieves better performance with an MLP consisting of 2–3 linear layers, avoiding the use of a linear layer + ReLU projection structure, we conduct relevant experiments to further validate our framework. The experimental results show that SimCLR with a two-layer MLP projection head outperforms the SimCLR framework with a linear + ReLU projection on certain datasets, such as SHAR, HHAR, MotionSense, and MobiAct. In contrast, our proposed QCLHAR framework consistently demonstrates superior performance across all datasets. Specifically, on the MotionSense and USCHAD datasets, our framework improves the performance by 3.24% and 4.11%, respectively, compared to SimCLR with the MLP projection head. This indicates that our approach has advantages in handling HAR datasets and further validates the effectiveness and superiority of the QCLHAR framework.

Compared to supervised learning, our proposed model shows slightly lower performance. We believe QML is still in its infancy. Therefore, constrained by the limited number of qubits in quantum hardware and the complexity of simulating quantum circuits on classical computers, QML has not yet fully surpassed all classical machine learning tasks in terms of accuracy and scalability. The main focus of the current research direction is to demonstrate the potential of QML for existing and emerging applications. Moreover, from the Table 1, we can see that there is already a trend towards the performance of supervised learning on some datasets, such as MotionSense, USC-HAD and MobiAct.

4.2. Comparison the quality of activity recognition

We adopt the experimental scheme designed by Tang et al. (2021) to compare our models. On the HAR datasets, we present the delta of the confusion matrices obtained by subtracting the confusion matrices of the traditional SimCLR from that of the proposed QCLHAR, thereby reflecting their differences. Positive values on the main diagonal indicate an increase in the number of samples correctly classified, suggesting that the proposed model performs better. Negative values off the diagonal indicate a reduction in the number of samples misclassified. Therefore, green cells represent that our model surpasses the conventional SimCLR framework, while red cells indicate an increase in model confusion.

From Fig. 3, it is evident that QCLHAR has more green cells in the MotionSense, HHAR, USC-HAD, and MobiAct datasets, indicating improved performance over SimCLR for most activities within these datasets. Especially in MotionSense and MobiAct, we can observe positive improvements across all categories. Specifically, in the MotionSense dataset, the quantum method demonstrates advantages in activities numbered 2 to 5. In the HHAR dataset, the quantum method exhibits significant superiority in the activities of walking upstairs and downstairs. In the USC-HAD dataset, the quantum method also shows obvious advantages in walking upstairs

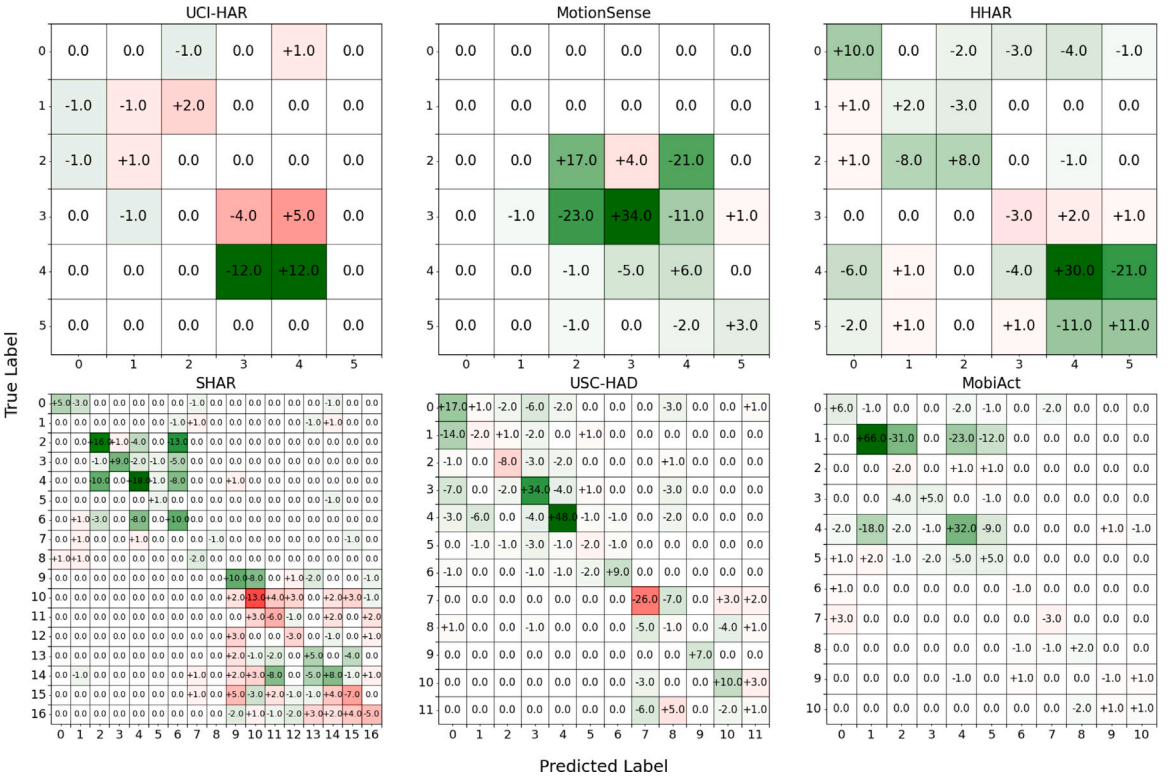

Fig. 3. Comparison the quality of activity recognition. Delta of confusion matrices between QCLHAR and SimCLR on six different datasets. The green cells represent indicate an increase in the number of samples correctly classified. The red cells represent a decrease in performance. The detailed activity categories corresponding to the numbers on the axes of the confusion matrix can be found in Table 2.

and downstairs activities numbered 3 and 4, but it performs less effectively than the classical method in the sitting activity. For the other two datasets SHAR and UCI-HAR, although there is not a significant enhancement, the performance is comparable to the conventional model, and this is achieved with fewer parameters. In the UCI-HAR dataset, compared to classical method, the quantum method performs better in activities numbered 3 and 4, specifically sitting and standing. It increases the number of correctly classified samples and reduces the number of misclassified samples. Moreover, for the 9–16 categories that do not perform well in the SHAR dataset. We can deduce from Table 2 that these represent eight types of fall detection, and we consider that such a quantum framework may be less effective than classical approaches in some cases, such as fall detection.

4.3. Evaluation with different projection heads

As illustrated in Fig. 4, we compare the results using three different architectures for the projection head. The term "w/o projector" denotes the training process of SimCLR without a projection head, relying solely on the feature mapping obtained from the encoder to compute the contrastive loss. The distinction between SimCLR and QCLHAR lies in their projection heads. While SimCLR employs a classical non-linear projection head, QCLHAR utilizes the quantum projection head we designed. It is worth noting that, due to memory constraints, the batch size for the MotionSense dataset is set to 64.

The figure clearly illustrates the importance of a projection head. When testing on unseen datasets, employing a projection head, either non-linear or quantum, significantly boosts performance. Specifically, on the SHAR dataset, the use of a non-linear projection head results in a direct improvement of 10.34%, while our proposed quantum projection head results in a 13.26% improvement. Furthermore, our proposed quantum projection head outperforms both the non-linear projection head and the approach without any projection head. This indicates that our quantum projection head can learn better feature representations in contrastive learning, thereby enhancing the performance of downstream tasks.

4.4. The performance of different amounts of unlabeled data

To evaluate the performance of the QCLHAR model in learning from unlabeled data, we remove the labels from the UCI-HAR, SHAR, HHAR, MotionSense, USCHAD, and MobiAct datasets. We then train the model using different proportions (from 20% to 100%) of these unlabeled pre-training data and compare it with the traditional SimCLR model. As shown in Fig. 5, the results show that QCLHAR consistently outperforms SimCLR across all proportions of pre-training data. This is particularly notable on

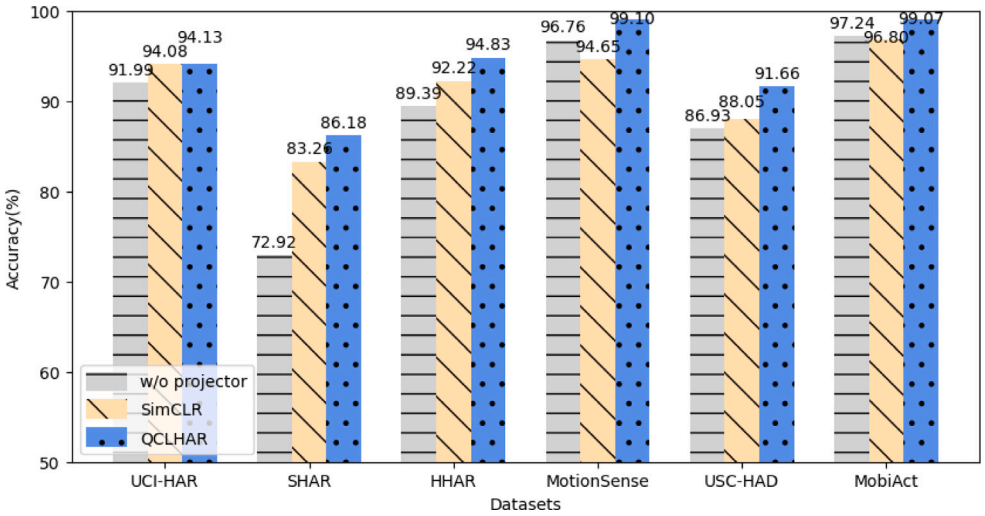

Fig. 4. Study of classification performance using different projection heads. We utilize a no projection head, a classical two-layer nonlinear projection head, and the proposed quantum projection head for pre-training. Subsequently, an evaluation of activity classification is conducted on six HAR datasets. It is observed that QCLHAR demonstrates a significant advantage compared to the other methods.

the MotionSense, USC-HAD and Mobiact datasets, suggesting that the quantum projection head may be more effective in feature extraction compared to the classical nonlinear head. Furthermore, QCLHAR demonstrates the capacity to maintain high accuracy even with limited pre-training data, highlighting its efficient data utilization. These findings emphasis the significance of pre-training data in improving model performance and suggest that QCLHAR is able to leverage data more effectively, thereby offering a considerable advantage in contrastive learning scenarios over conventional approaches.

4.5. The performance of different amounts of labeled data

We then compare the accuracy performance with different amounts of labeled data, where we fix the total amount of pre-training data. As illustrated in Fig. 6, both frameworks exhibit improved classification accuracy as the amount of labeled data increases. Moreover, across all proportions of labeled data, QCLHAR consistently achieves higher accuracy than SimCLR. Notably, when the amount of labeled data is relatively limited (i.e., from 20% to 40%), the performance gap between QCLHAR and SimCLR is more obvious. Particularly when utilizing only 20% of the data for fine-tuning, there is nearly a 10 percentage point difference in their performance on the USC-HAD dataset. This shows that the QCLHAR framework has stronger robustness and efficiency under limited labeled data, making it more advantageous in practical applications.

4.6. Visualization of the salient regions using saliency maps

The previous experiments have proved the effectiveness of our QCLHAR for activity classification tasks. However, it still cannot outperform the fully supervised learning. To verify whether the proposed model can have salient regions similar to supervised learning, we propose to use saliency maps to compare QCLHAR and the fully supervised model.

We randomly select test samples from the MotionSense dataset for visualizing saliency maps. The absolute value of the gradient is considered the measure of saliency, representing the influence of the input data on the model's prediction decision. The larger the absolute value of the gradient, the more important to the model's prediction.

Fig. 7 illustrates the saliency maps generated by the two networks for the same test sample from the MotionSense dataset. We show the raw input data of the three-axis accelerometer in the top pane. The middle and bottom panes show the saliency maps of the input data produced by QCLHAR and the fully supervised learning, respectively. The darker color indicates that region contributes the most to the model's predictions. From the figure, we can observe that QCLHAR and the fully supervised model indicate similar patterns in color intensity. This suggests that QCLHAR and the fully supervised learning model largely focus on similar regions for prediction, and also proves that QCLHAR can generate meaningful representations for activity classification tasks.

4.7. Comparative analysis of model parameters

On the MotionSense dataset, we compare the performance differences between the traditional contrastive learning SimCLR, the quantum contrastive learning framework "L+Q" with a hybrid classical–quantum projection head, and the proposed QCLHAR. As shown in Table 3, models containing quantum circuit consistently outperform the traditional SimCLR model. Specifically, the hybrid scheme achieves an F1-score of 95.78%, while the pure quantum model QCLHAR reaches an impressive 98.19%. This clearly

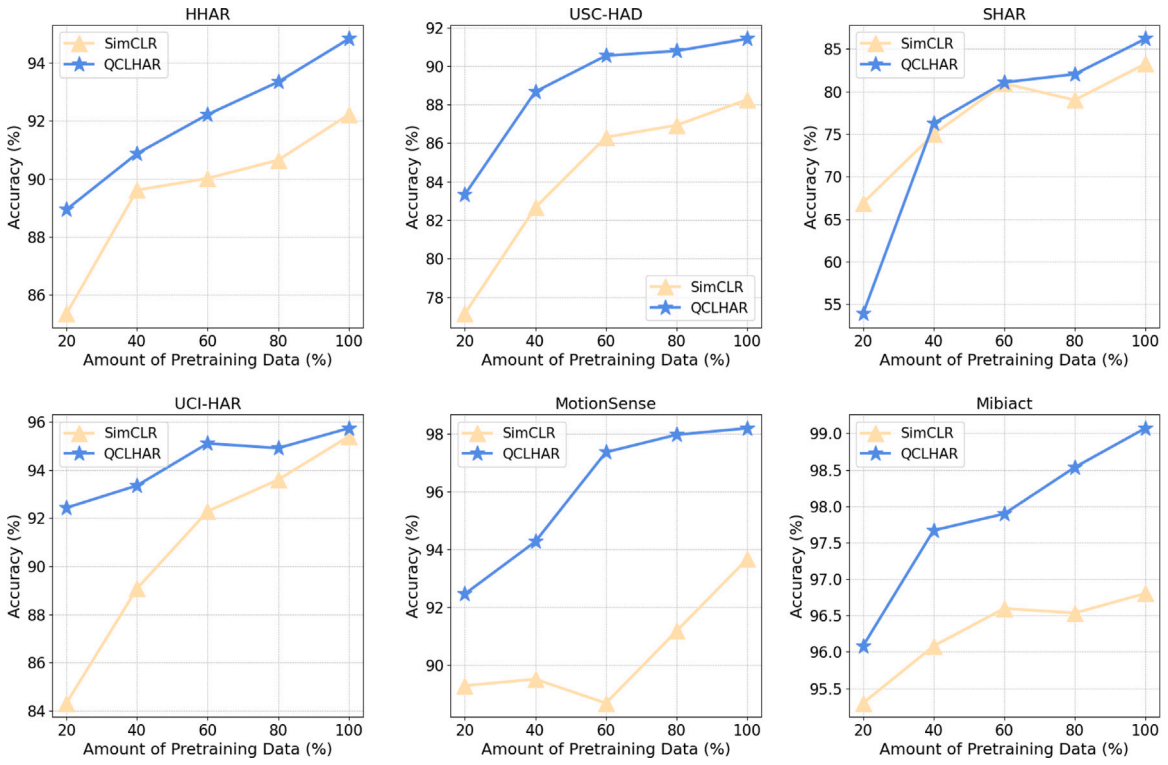

Fig. 5. Assessing the classification performance of QCLHAR and SimCLR with different amounts of pretraining data on six datasets. We randomly divide the training data into subsets of 20%, 40%, 60%, 80%, and 100% for training. The yellow curve (baseline) represents the performance when using a classical nonlinear head for self-supervised learning, while the blue curve depicts the performance variation when training with the proposed quantum projection head. We find that QCLHAR can achieve higher performance with less training data in most cases.

Table 3
The comparison of activity recognition accuracy and the quantity of parameters used by each model on MotionSense dataset. The encoders are identical, the difference in the number of parameters is only in the projection head.

Model	F1-score(↑)	Parameters(↓)	Training time	Inference time
SimCLR	93.6652	C_{Enc}+182.28 MB+3.38 MB	0.17	0.21
QCLHAR(L+Q)	95.7768	C_{Enc}+182.28 MB+8(qubits)	0.15	0.16
QCLHAR	98.1900	C_{Enc}+8(qubits)	0.16	0.15

Table 4
The comparison of activity recognition accuracy and the quantity of parameters used by each model on USCHAD dataset. The encoders are identical, the difference in the number of parameters is only in the projection head.

Model	F1-score(↑)	Parameters(↓)	Training time	Inference time
SimCLR	88.0498	C_{Enc}+182.28 MB+3.38 MB	0.31	0.35
QCLHAR(L+Q)	89.6266	C_{Enc}+182.28 MB+8(qubits)	0.25	0.28
QCLHAR	91.6598	C_{Enc}+8(qubits)	0.25	0.25

indicates that a pure quantum circuit in the projection head can achieve superior performance. Even more striking is that as the F1-score increases, the number of parameters significantly decreases. QCLHAR only requires an additional 8 parameters apart from the encoder parameters, which is significantly fewer compared to other models. Similar results can also be found on other datasets. Additionally, we compare the average classifier training time per epoch and the inference time for the final classification test across these models. As shown in the Table 3, after pretraining with the quantum projection head, QCLHAR has lower average classifier training and inference time compared to the traditional SimCLR framework. This suggests that using quantum methods can not only enhance model performance but also offer advantages in terms of parameter count and computational resource requirements, potentially leading to a faster training speed and a lower computational demand.

To further validate our model's ability to improve performance while reducing parameter count and decreasing training and inference time, we conduct corresponding experiments on the USCHAD dataset. As shown in Table 4, we observe similar conclusions to those from the MotionSense dataset. Specifically, using the QCLHAR model, only 8 additional parameters are needed besides the encoder, with an average classifier training time of 0.25 s and a classification test inference time of 0.25 s as well. In comparison,

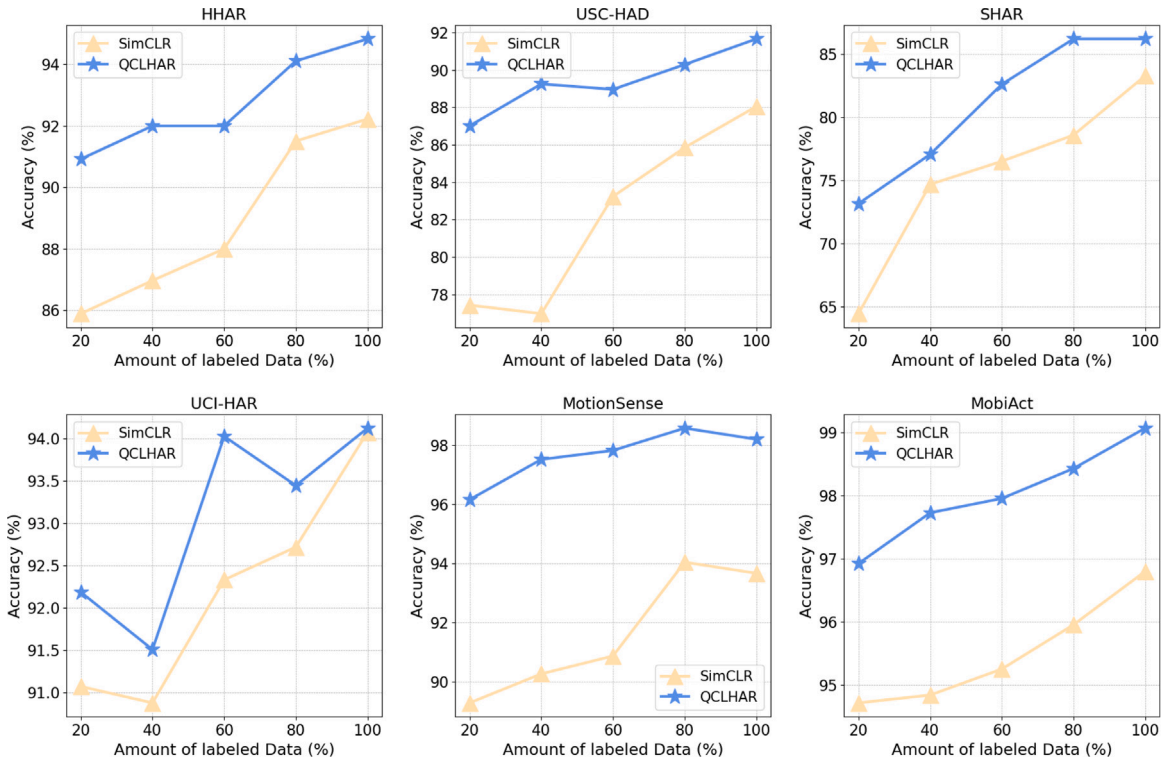

Fig. 6. Assessing the classification performance of QCLHAR and SimCLR with different percentages of labeled data on six datasets. We find that QCLHAR performs better with less fine-tuning data.

the classical SimCLR model requires 0.31 s and 0.35 s, respectively. These results demonstrate that our proposed QCLHAR model can significantly enhance performance while reducing parameter count, training time, and inference time.

5. Discussion

In this section, we first design a series of experiments to evaluate different quantum projection head schemes, aiming to better explore their effectiveness in contrastive learning and determine the optimal scheme. Subsequently, we analyze the influence of different quantum circuit depths, exploring whether deeper circuits lead to enhanced results. Finally, we perform a sensitivity analysis of QCLHAR under various system settings to measure its robustness and adaptability and compare the parameter quantities used by different models.

5.1. The performance of different quantum projection head schemes on MotionSense

To further explore the capabilities of QML in contrastive learning, we systematically compare different projection head design schemes.

5.1.1. Comparative analysis of hybrid and pure quantum projector

As shown in Fig. 8, we compare four projection head design schemes: the traditional non-linear projection head (SimCLR), our proposed quantum projection head (QCLHAR), a hybrid design that combines classical linear layers with a quantum circuit (L+Q), and another hybrid design incorporating the ReLU activation function (L+Q(ReLU)). This comparison is designed to provide a continuous perspective on the performance transition from purely classical to purely quantum schemes. It explores the value of employing quantum techniques in contrastive learning and integrating classical networks with quantum networks. Due to the fact that the ReLU activation function is prevalent in deep learning, and considering that the traditional SimCLR design enhanced its performance by incorporating ReLU into its projection head, we specifically include the L+Q(ReLU) scheme for comparison to observe its effects and role in a hybrid design.

From the experimental results, we can clearly observe that the performance of QCLHAR is ahead of other designs with an F1 score of 98.19%, which verifies the obvious advantage of the pure quantum circuit projection head design in contrastive learning. The hybrid design L+Q and L+Q(ReLU) achieve F1 scores of 95.78% and 93.97%, respectively, demonstrating the transitional performance between pure classical and pure quantum design. In particular, the performance of L+Q(ReLU) slightly declines

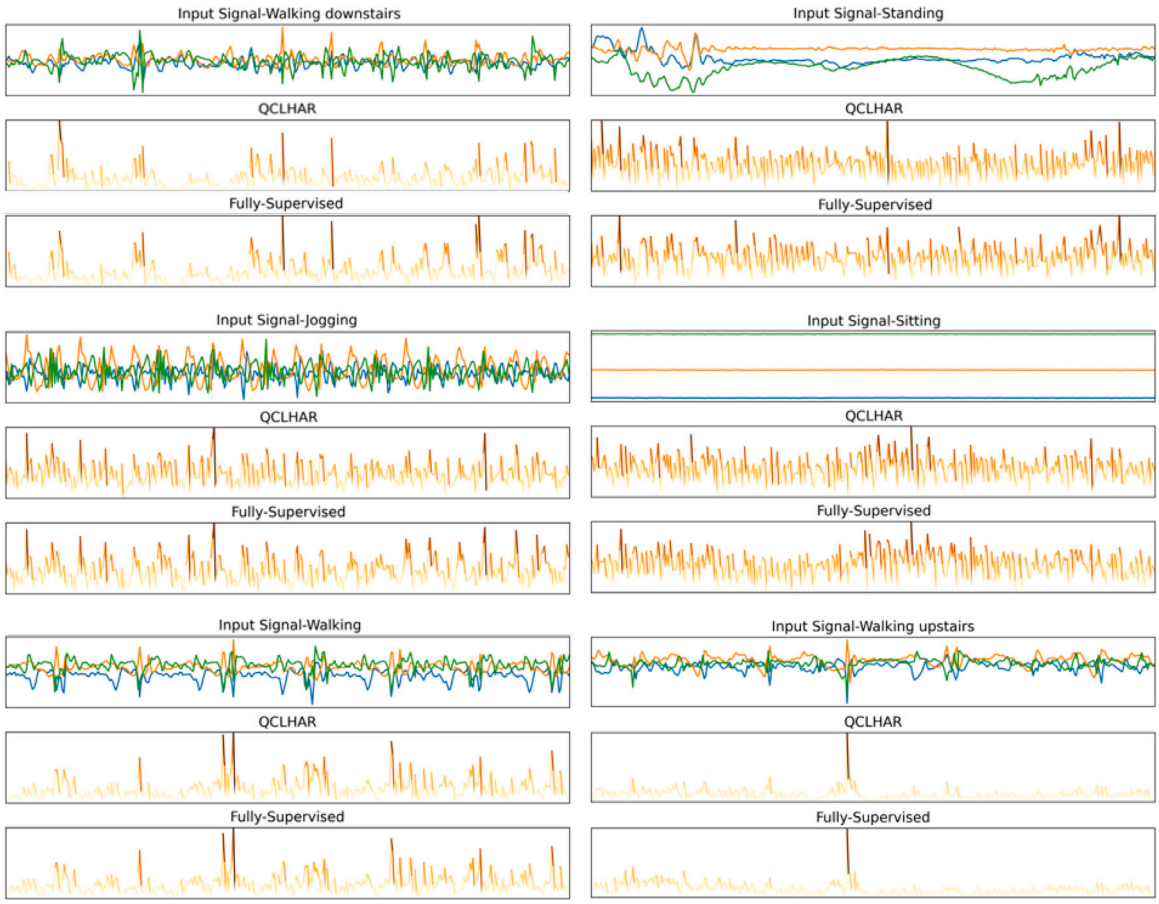

Fig. 7. Saliency map of a randomly selected sample from the MotionSense dataset. The top pane represents the raw input signal, and the middle and bottom panes show the amplitude of the input signal under the QCLHAR and the fully supervised learning networks, respectively. The depth of the color indicates the regions that have a substantial impact on the model's prediction. Our proposed QCLHAR and the fully supervised learning show similar regions, indicating that they focus on similar areas for prediction.

compared to L+Q, suggesting that the ReLU activation function might not be suitable for the hybrid classical–quantum projection head design and may destroy the extracted feature information. In addition, the classical SimCLR projection head scores 93.67% in this experiment, which is close to but slightly lower than the hybrid design schemes L+Q. This implies that even introducing partial quantum computation into the projection head can lead to performance improvement. This further validates the critical importance of the projection head design in contrastive learning and the potential value of quantum computing in this field.

5.1.2. Comparative analysis of multiple quantum circuits

In classical contrastive learning, the design of the projection head mostly adopts a two-layer linear mapping. Such a strategy has been extensively employed in the field. Inspired by this, when introducing the quantum projection head into contrastive learning, we pose a natural question whether it is feasible to design two quantum circuits to imitate the structure of these two linear layers and explore their effects on contrastive learning. To evaluate the efficacy of such dual quantum circuits in emulating the classical bilinear mapping and potentially enhancing performance, we design five distinct schemes. These schemes incorporate two quantum circuits serving as the projection head, as shown in Fig. 9. We examine different encoding methods, such as amplitude encoding and angle encoding, and consider the inclusion of a ReLU activation function between the circuits when designing this quantum projection head. The details are as follows:

- **Two Amp:** This scheme is entirely based on amplitude encoding and consists of two amplitude-encoded quantum circuits. The first quantum circuit uses 8 qubits to encode 256-dimensional classical data, while the second quantum circuit employs 3 qubits to encode the output of the first circuit.
- **Two Amp+ReLU:** On the basis of "Two Amp" scheme, a ReLU activation function is added after the second quantum circuit. Since amplitude encoding requires data normalization before encoding into quantum states, and due to the presence of gradients in two continuous amplitude-encoded circuits, manual normalization cannot be performed after the output of the first quantum circuit. Therefore, we do not design a scheme to add ReLU between the two amplitude-encoded circuits.

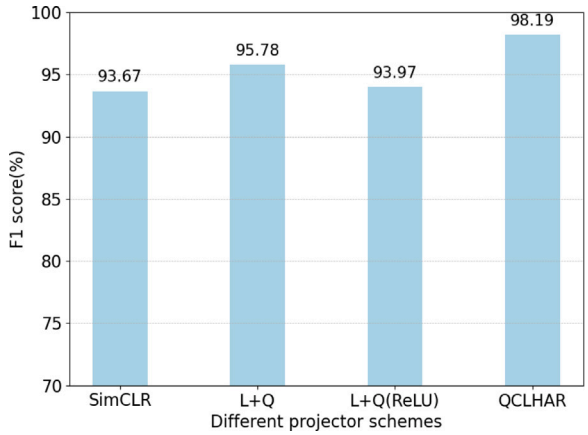

Fig. 8. The performance of different projection head schemes on MotionSense. We evaluate the performance of SimCLR, the hybrid classical–quantum projection head scheme, and QCLHAR on the MotionSense dataset. The results report a significant advantage of quantum technology.

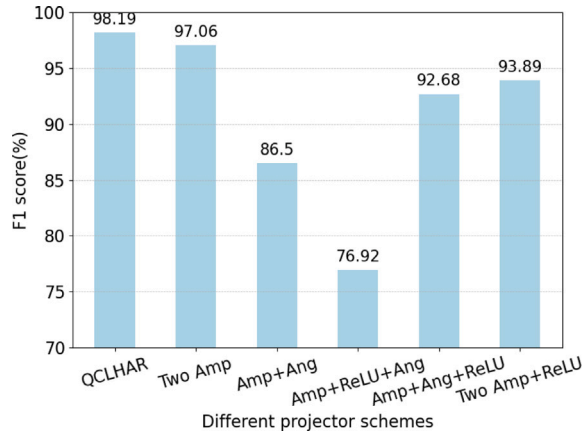

Fig. 9. The performance of multiple quantum circuit projection head schemes on MotionSense. We design different two-quantum circuit structures for the projection head scheme and evaluate the performance on MotionSense dataset.

- **Amp+Ang:** This scheme utilizes a hybrid encoding approach. The first quantum circuit employs amplitude encoding for 256-dimensional classical data, while the second quantum circuit employs 8 qubits for embedding data into quantum states through angle encoding.
- **Amp+ReLU+Ang:** Building on the "Amp+Ang" scheme, a ReLU activation function is incorporated between the two quantum circuits.
- **Amp+Ang+ReLU:** Similar to the "Two Amp+ReLU" scheme, this scheme adds a ReLU at the end of the hybrid encoding scheme.

From the experimental results present in Fig. 9, it is observed that a single quantum circuit (QCLHAR) as the projection head performs the best. The "Two Amp" scheme, entirely based on amplitude encoding, follows closely with an F1 score of 97.06%. This indicates that the scheme based solely on amplitude encoding can effectively retain certain features of the input data. However, when introducing ReLU after this circuit scheme (Two Amp+ReLU), the performance declined. This indicates that ReLU is not suitable to be incorporated into this projection head design. When combining amplitude encoding with angle encoding (Amp+Ang), the performance drops significantly. We speculate that this might be due to the necessity to convert data into angles before angle encoding. Transitioning directly from one quantum circuit to another becomes infeasible due to gradient effects, leading to reduced performance. Furthermore, the introduction of a ReLU activation function between the two quantum circuits further diminishes performance, as evidenced by the "Amp+ReLU+Ang" scheme scoring only 76.92%. This suggests that ReLU might cause information loss or feature distortion from the original circuit, resulting in performance degradation. Additionally, we also experiment with introducing ReLU at the end of the hybrid encoding (Amp+Ang+ReLU), which improves the performance to 92.68%. However, although introducing ReLU after a quantum circuit can bring about certain performance improvements in certain scenarios, these improvements remain limited compared to a single quantum circuit (QCLHAR). This also implies that fine-tuning the final output result using ReLU, rather than the intermediate encoding result, might be more beneficial for performance enhancement. It also

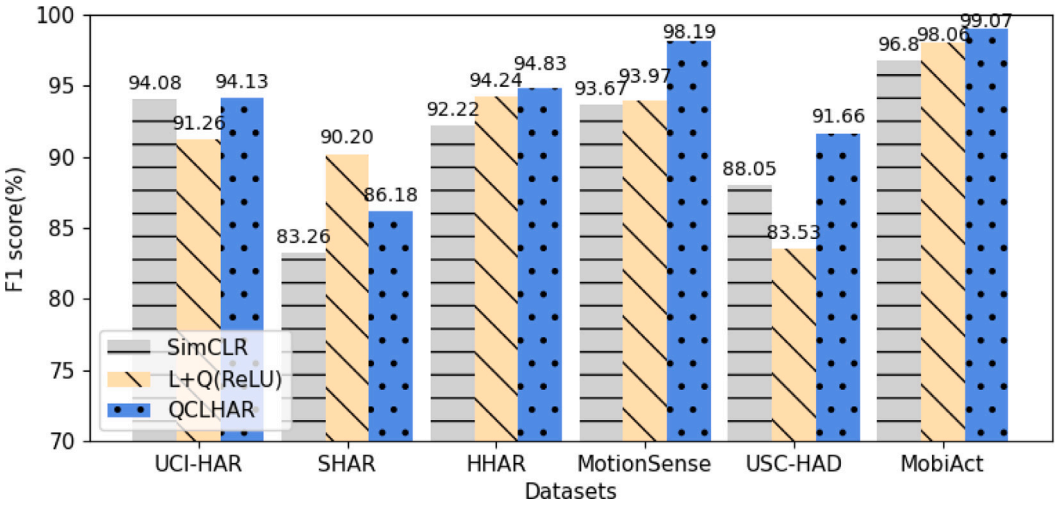

Fig. 10. The performance of hybrid(L+Q) projector and QCLHAR on six datasets.

indicates that the position of ReLU has a significant impact on its effectiveness. In general, while ReLU is a common activation function in classical contrastive learning, its use in quantum computing is not ideal. It might adversely affect quantum states, leading to feature information loss. These results suggest that designing two quantum circuits similar to classical projection heads is not effective. Moreover, when designing circuits, careful consideration should be given to the encoding method and the use of non-linear activation functions to ensure that the designed quantum circuit can effectively accomplish the contrastive learning task.

5.2. Comparative analysis of Hybrid(L+Q) and pure quantum projector

When comparing the performance of the classical projection head (SimCLR), the hybrid classical–quantum projection head (L+Q(ReLU)), and our proposed pure quantum projection head (QCLHAR), we observe an interesting phenomenon. As shown in Fig. 10, we can see that on the SHAR dataset, the hybrid scheme surprisingly surpasses our QCLHAR with an F1 score of 90.20%. On the HHAR dataset, the hybrid scheme exhibit comparable performance to QCLHAR. However, on other datasets, we observe that the hybrid classical–quantum projection head design scheme consistently performs worse compared to our proposed method. Additionally, it even exhibits poorer performance than the classical SimCLR model. This suggests that different datasets might be suitable for different types of projection head designs. However, it is noteworthy that our proposed approach achieves superior performance on all datasets (except SHAR), reaching a remarkable 99% F1-score on the MotionSense and MobiAct datasets. Moreover, on the SHAR dataset, the superior performance is achieved due to the integration of quantum circuits. This indicates that introducing quantum circuits into the projection head seems to be able to achieve performance beyond classical schemes, demonstrating the enormous potential of QML in the field of machine learning.

5.3. Comparison of different depths of VQC network on SHAR

In QML, the number of layers or the depth of the quantum circuit is a crucial factor affecting its representational capability. Theoretically, deeper circuits can achieve more complex transformations, thereby offering better performance. However, due to noise and other factors in quantum computation, deeper circuits might not necessarily bring the expected benefits. Therefore, we conduct this experiment to explore the impact of using parameterized quantum circuit (PQC) with different depths as projection heads. From Fig. 11, we can observe that the F1 score obtained using the traditional SimCLR strategy is 83.26%. When we employ a PQC with a depth of 1 as the projection head, there is a significant performance boost, with the F1 score reaching 86.18%. However, as the circuit depth further increases, the performance starts to decline. This suggests that while using a quantum circuit as a projection head can enhance performance, excessive increase in circuit depth might not be beneficial. We believe this is because the task of the projection head does not require too complex circuit structure. The circuits that are too deep might introduce greater noise and additional computational complexity, making training challenging and reducing performance. Therefore, when designing quantum circuits, it is essential to thoroughly consider the network's requirements, design an appropriate circuit structure and depth, and find the optimal depth to achieve the best performance.

5.4. The impact of NISQ-era quantum computing error on the results

In the noisy intermediate-scale quantum (NISQ) era, quantum error is an important challenge that may significantly affect the performance of quantum computers. However, deploying contrastive learning networks onto real quantum circuits is a highly

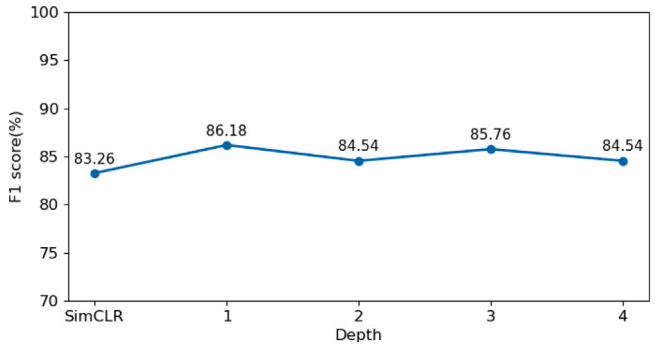

Fig. 11. The performance of different PQC depths and SimCLR on SHAR.

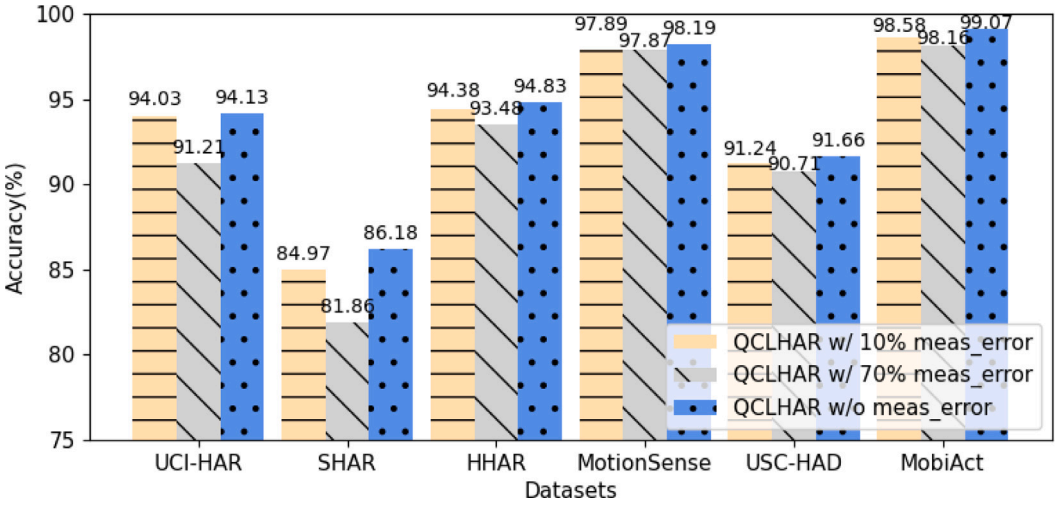

Fig. 12. The performance of with and without measurement errors.

complex and time-consuming task. Uploading classical data to quantum circuits requires a significant amount of time and is concurrently constrained by the availability of quantum bits. Additionally, it is difficult to obtain and operate quantum devices. In light of these challenges, we are indeed confronted with constraints imposed by finite resources, rendering us incapable of conducting deployments and experiments on real quantum hardware. Nevertheless, we remain committed to utilizing quantum simulators to simulate the behavior of quantum circuits as closely as possible in simulated environments, striving to evaluate the performance of our methods accurately.

Specifically, we introduce random measurement errors on the quantum simulator to simulate the effects of measurement errors. This is achieved by applying random rotation gates RX with random numbers conforming to a normal distribution on each qubit. We quantify the degree of added errors by applying such random rotation gates to the qubits before each quantum gate operation with a probability of 10% or 70%. The rotation angles for these gates follow a normal distribution. A 10% measurement error is used to simulate the low-level noise in NISQ devices, while a 70% measurement error is used to simulate high-level measurement noise. Through this method, the measurement results of quantum bits are affected by random rotation gates, effectively simulating actual noise in quantum computation.

We compare the performance results with and without introducing measurement errors on six datasets. The experimental results, as shown in Fig. 12, indicate that introducing a 10% random measurement errors have a certain impact on our framework, resulting in some degradation of our model's performance, especially on the SHAR dataset. However, on other datasets, the introduction of measurement errors does not significantly degrade our model's performance, demonstrating relative stability. When introducing a 70% random error, the performance on the UCI-HAR and SHAR datasets decreases significantly. For instance, on the SHAR dataset, the performance decreases from 86.18% to 81.86% with a 70% measurement error. However, on the other four datasets, there is no notable decline in model performance. Overall, as measurement error increases, there is a decline in performance on certain datasets. However, the model maintains relatively stable performance under high-error conditions on most datasets, indicating its strong robustness to measurement error.

By simulating reasonable errors on the quantum simulator, we are able to evaluate our method's performance in a more realistic quantum computing environment. Although we have not directly conducted experiments on actual quantum hardware, this

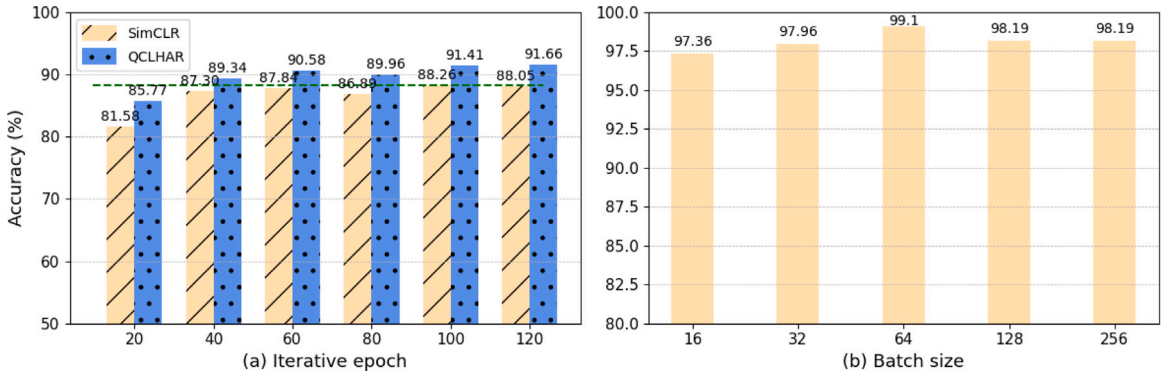

Fig. 13. Accuracy of QCLHAR under different settings. (a) Evaluate the performance variation of SimCLR and QCLHAR across different epochs; (b) Study the impact of performance when using different batch sizes $\in \{16, 32, 64, 128, 256\}$.

approximation method still provides some understanding of the impact of error rates and offers valuable insights for our research. Furthermore, it can also provide a useful reference point for future experiments on real quantum hardware.

5.5. Micro-benchmark performance

5.5.1. Different iterative epochs

In this experiment, we study the impact of the iterative epoch on accuracy performance. The results are shown in (a) of Fig. 13. We observe that QCLHAR outperforms SimCLR at every iterative training process. Moreover, QCLHAR demonstrates significant superiority in fewer epoch training, and its convergence speed is notably faster than SimCLR. Specifically, SimCLR requires 100 epochs to reach its best performance of 88.26%, while QCLHAR only needs 40 epochs to surpass SimCLR's best performance. This suggests that QCLHAR can achieve high-quality feature representations earlier in practical applications. As the number of iterations increases, although the growth trend of QCLHAR's accuracy begins to slow down, it still continues to rise. This implies that there is potential for further performance improvement if the number of iterations is increased again. From these observations, we can infer that when using QCLHAR, we can employ early stopping strategies to save training resources and time while maintaining commendable performance.

5.5.2. Different batch sizes in QCLHAR

We further investigate the performance of QCLHAR with different batch sizes. In previous contrastive learning works (Chen et al., 2020), they often require a large batch size to ensure good performance. This is because larger batches could provide more negative samples to the model, thereby enhancing the discriminative ability of the model. However, as shown in (b) of Fig. 13, QCLHAR still achieves good performance even with small batches (i.e., 64 or 128). Thus, QCLHAR does not rely on large batches of data as traditional contrastive learning methods do. As the batch size increases, the accuracy slightly declines and stabilizes, maintaining a level of around 98.19%. This may be due to the increased diversity of negative samples, making convergence more challenging.

6. Conclusion

In this paper, we propose a quantum contrastive learning framework (i.e., QCLHAR) for HAR, which is a novel self-supervised learning paradigm integrating quantum technology. By leveraging readily available unlabeled data, this framework addresses the challenge of limited labeled data and alleviates the classical hardware bottleneck. Specifically, we design a variational quantum circuit to serve as the projection head of the contrastive learning framework, which uses the unique properties of quantum technology to efficiently achieve data compression and feature information representation. On several publicly available HAR datasets, we demonstrate the effectiveness of our designed quantum projection head for HAR tasks. Our approach surpasses classical contrastive learning models with fewer parameters, showing its superiority over the classical SimCLR framework. Additionally, through a series of detailed experiments on various projection head schemes, we show that quantum technology has great potential. With the advancement of quantum technology and the increase in the number of qubits, we believe there will be a considerable enhancement in the HAR task performance. Although the application domain of this paper is focused on HAR tasks based on inertial sensors, the introduced quantum technology and the designed framework are general and they can be applied to other domains (e.g., wireless sensing) in the future.

CRediT authorship contribution statement

Yanhui Ren: Investigation, Methodology, Validation, Visualization, Writing – original draft, Writing – review & editing. **Di Wang:** Supervision. **Lingling An:** Supervision. **Shiwen Mao:** Writing – review & editing. **Xuyu Wang:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Anguita, D., Ghio, A., Oneto, L., Parra, X., Reyes-Ortiz, J. L., et al. (2013). A public domain dataset for human activity recognition using smartphones. *vol. 3*, In *Esann* (p. 3).

Chatzaki, C., Pediaditis, M., Vavoulas, G., & Tsiknakis, M. (2017). Human daily activity and fall recognition using a smartphone's acceleration sensor. In *Information and communication technologies for ageing well and e-health: second international conference, ICT4AWE 2016, rome, Italy, April 21-22, 2016, revised selected papers 2* (pp. 100–118). Springer.

Chen, X., & He, K. (2021). Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 15750–15758).

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning* (pp. 1597–1607). PMLR.

Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., & Zisserman, A. (2021). With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 9588–9597).

Eldele, E., Ragab, M., Chen, Z., Wu, M., Kwoh, C. K., Li, X., et al. (2021). Time-series representation learning via temporal and contextual contrasting. arXiv preprint arXiv:2106.14112.

Gidaris, S., Singh, P., & Komodakis, N. (2018). Unsupervised representation learning by predicting image rotations. arXiv preprint arXiv:1803.07728.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., et al. (2020). Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, *33*, 21271–21284.

Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on theory of computing* (pp. 212–219).

Henaff, O. (2020). Data-efficient image recognition with contrastive predictive coding. In *International conference on machine learning* (pp. 4182–4192). PMLR.

Jaderberg, B., Anderson, L. W., Xie, W., Albanie, S., Kiffner, M., & Jaksch, D. (2022). Quantum self-supervised learning. *Quantum Science and Technology*, *7*(3), Article 035005.

Malekzadeh, M., Clegg, R. G., Cavallaro, A., & Haddadi, H. (2018). Protecting sensory data against sensitive inferences. In *Proceedings of the 1st workshop on privacy by design in distributed systems* (pp. 1–6).

Micucci, D., Mobilio, M., & Napoletano, P. (2017). Unimib shar: A dataset for human activity recognition using acceleration data from smartphones. *Applied Sciences*, *7*(10), 1101.

Saeed, A., Ozcelebi, T., & Lukkien, J. (2019). Multi-task self-supervised learning for human activity detection. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, *3*(2), 1–30.

Shor, P. W. (1994). Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science* (pp. 124–134). Ieee.

Stisen, A., Blunck, H., Bhattacharya, S., Prentow, T. S., Kjæ rgaard, M. B., Dey, A., et al. (2015). Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM conference on embedded networked sensor systems* (pp. 127–140).

Tang, C. I., Perez-Pozuelo, I., Spathis, D., Brage, S., Wareham, N., & Mascolo, C. (2021). Selfhar: Improving human activity recognition through self-training with unlabeled data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, *5*(1), 1–30.

Wang, J., Chen, Y., Hao, S., Peng, X., & Hu, L. (2019). Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, *119*, 3–11.

Wang, G., Wang, K., Wang, G., Torr, P. H., & Lin, L. (2021). Solving inefficiency of self-supervised representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 9505–9515).

Wang, J., Zhu, T., Gan, J., Chen, L. L., Ning, H., & Wan, Y. (2022). Sensor data augmentation by resampling in contrastive learning for human activity recognition. *IEEE Sensors Journal*, *22*(23), 22994–23008.

Wei, D., Lim, J. J., Zisserman, A., & Freeman, W. T. (2018). Learning and using the arrow of time. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8052–8060).

Yao, S., Zhao, Y., Shao, H., Zhang, C., Zhang, A., Hu, S., et al. (2018). Sensegan: Enabling deep learning for internet of things with a semi-supervised framework. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, *2*(3), 1–21.

Zhang, M., & Sawchuk, A. A. (2012). USC-HAD: A daily activity dataset for ubiquitous activity recognition using wearable sensors. In *Proceedings of the 2012 ACM conference on ubiquitous computing* (pp. 1036–1043).