Sensors and Actuators A 331 (2021) 112900

Contents lists available at ScienceDirect

Sensors and Actuators A: Physical

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

# A portable sitting posture monitoring system based on a pressure sensor array and machine learning

Xu Ran, Cong Wang, Yao Xiao, Xuliang Gao, Zhiyuan Zhu, Bin Chen*

*Chongqing Key Laboratory of Non-linear Circuit and Intelligent Information Processing, College of Electronic and Information Engineering, Southwest University, Chongqing, 400715, China*

ARTICLE INFO

*Article history:*
Received 20 January 2021
Received in revised form 16 May 2021
Accepted 6 June 2021
Available online 9 June 2021

*Keywords:*
Sitting posture
Sensor array
Machine learning
Vibration feedback

ABSTRACT

Poor sitting posture is one of the main inducements that lead to a series of skeletal muscle diseases. Sitting posture monitoring system can remind the user to maintain the correct sitting posture to prevent the harm of poor sitting posture to the body. In this paper, we proposed a portable sitting posture monitoring system to recognize the user's sitting posture and feedback the results in real time. A pressure sensor array is used to collect sitting postures related information, while the collected data can be displayed on a computer. The proposed system was designed to recognize seven types of sitting postures, including sitting upright, leaning forward, leaning backward, leaning left, leaning right, cross left leg, and cross right leg. Seven machine learning algorithms were implemented for comparation. The results showed that a five-layer Artificial Neural Network could achieve the highest accuracy of 97.07 %. To enhance system performance and reduce hardware cost, we further optimized the size of the sensor array. An $11 \times 13$ sensor array combined with Random Forest algorithm realized the optimal balance between classification accuracy (96.26 %) and hardware resource consumption. The final system prediction time is 19 us on the Raspberry Pi, which could satisfy the practical application requirement on the embedded platform.



## 1. Introduction

The advances in information technologies and the development of labor saving devices have engineered sedentary activity into the modern lifestyle [1]. It is reported that children and adults in the United States spent approximately 55 % of their waking hours, or 7.7 h a day, in sedentary behaviors; senior teenagers (16–19 years old) and adults (60–85 years old) spend nearly 60 % of their time, or more than 8 h a day, in sedentary behavior [2]. Many adults are in occupations that require prolonged sitting time. A study found that white collar occupations being five times more likely to spent sitting more than 7.5 h per day compared to manual workers [3]. Long sitting time in daily are associated with a greater risk of all-cause mortality [4]. Recent researches have approved that poor sitting postures have been considered in association with a number of spinal musculoskeletal disorders, including structural deformity of the spine and back pain [5–8]. In various common sitting positions, slouched sitting has been suggested to cause viscoelastic creep, which may impair trunk muscle activity and proprioception, and heighten the risk of low back pain [9]. Furthermore, sitting for more than 30−80 min may increase the risk of neck and low back pain, and sitting for more than 4 h can increase musculoskeletal discomfort in various parts of the body [10]. In addition to musculoskeletal and metabolic health risks, long periods of sedentary activity may have a negative impact on the mental state and will weaken the ability to solve creative problems [11]. Hence, it is particularly important to develop a sitting posture monitoring system that can evaluate the posture of the seated person in real time and correct the sitting posture to avoid the health hazards of sedentary behaviors include by the poor sitting posture.

There already have studies on sitting posture monitoring system in recent years. They can be roughly divided into two categories from the aspect of sensors: vision sensor and pressure sensor. The tremendous progress of image processing technology has enabled visual sensors to be applied in the field of sitting posture monitoring. Paliyawan et al. used the real-time skeleton data stream captured by a single Kinect camera to perform data mining and classification to detect whether office workers have been sitting for a long time [12]. Mallare et al. presented a method of obtaining sitting posture parameters and assess it through the use of Computer Vision to be used as an assistance for physical therapists in their sitting posture assessment and correction [13]. Mu et al. extracted the user's sitting contour features based on pattern matching and compares them with standard contour features to detect sitting posture [14]. Bei et al. applied Astra 3D sensors for data collection

* Corresponding author.
*E-mail address:* chenbin121@swu.edu.cn (B. Chen).

https://doi.org/10.1016/j.sna.2021.112900
0924-4247/

to extract relative topological features and local edge features for sitting posture classification [15].

Compared with visual sensors, pressure sensors obtain data more directly and are not affected by the intensity of ambient light. The piezoelectric effect has been proposed for more than 150 years, and it has been widely used in commercial and non-commercial sensor applications [16,17]. Pressure sensors based sitting posture monitoring systems are usually implemented by installing pressure sensors on chair surface and back [18–22]. However, sensors fixed on the seat reduces the portability of the system and has the problem of inconvenient installation, which limits the large-scale use of the sitting posture detection system. Most sitting posture monitoring systems use Arduino to collect data from the pressure sensor, and send the data to the computer for processing via Bluetooth [23–25]. In recent years, due to the tremendous progress of machine learning algorithms, there already have studies using machine learning algorithms for sitting posture classification. Yongxiang et al. obtained the body pressure distribution of different sitting postures through the pressure sensor array, and used the support vector machine algorithm to recognize the human body posture with an accuracy rate of 83.33 % [26]. Hu et al. used six flex sensors, an Analog to Digital Converter (ADC) board and a two-layer Artificial Neural Network on a Spartan-6 Field Programmable Gate Array (FPGA), and the system achieves 97.78 % accuracy with a floating-point evaluation [27]. Furthermore, Ma et al. installed 12 pressure sensors on the back and surface of the chair, through five supervised learning algorithms to achieve a maximum accuracy of 99.48 % in five sitting positions [28].

In recent years, researches have proposed sitting posture monitoring systems based on various sensors; however, they are rarely commercialized due to poor portability or user experience. Most sitting posture monitoring systems have fixed the pressure sensor on the seat, and the sensor needs to be reinstalled when the seat is replaced. In addition, many sitting posture classification algorithms run on the computer and users need to view the sitting classification results via computer, which greatly limits the practicability of the system. A few studies have transplanted classification algorithms to embedded systems to improve the portability of the system. In this work, we designed a portable sitting posture monitoring system. A high-resolution pressure sensor array is one of the key components in the system. The data acquisition card collects the voltage of each sensing point on the sensing array and performs ADC, and then transmits the sitting posture related data to the Raspberry Pi via Universal Serial Bus (USB). The sitting posture data is processed on the Raspberry Pi with a series of machine learning algorithm. RF algorithm combined with $11 \times 13$ sensor array realized the optimal balance between classification accuracy and hardware resource consumption. For the user to maintain the correct sitting posture, the classification results are vibrated back to the user. This study has three main contributions: (1) The data is processed in real time on the Raspberry Pi and the system feeds back the classification results by vibrating the flat motor. (2) For the sake of data display, storage and analysis, we have developed corresponding software, through which the sitting posture related data can be visualized. (3) A series of machine learning algorithms and the size of sensor array are optimized. The proposed system could realize classification accuracy of 96.26 % with 19 us prediction time.

## 2. Sitting posture monitoring system design

### 2.1. Sitting pressure sensing

Our smart cushion is equipped with a commercially available pressure sensor (IMM00014, I-MOTION) for pressure measurement. The pressure sensor is formed by printing conductive electrodes and force-sensitive materials on a flexible polyester substrate, and then encapsulating the upper and lower substrates face-to-face. The resistance of force-sensitive material decreases with increasing pressure. For multiple sensor nodes, collinear output can be adopted, because collinear can effectively save the number of output signal lines. For example, the array pressure sensor we use is a collinear output. As shown in Fig. 1, the top layer outputs 44 signals, and the bottom layer outputs 52 signals. The 96-way electrodes are connected to the data acquisition card through the Flexible Printed Circuit board (FPC) interface. There are a total of 2288 ($44 \times 52$) sensing points on the sensor.

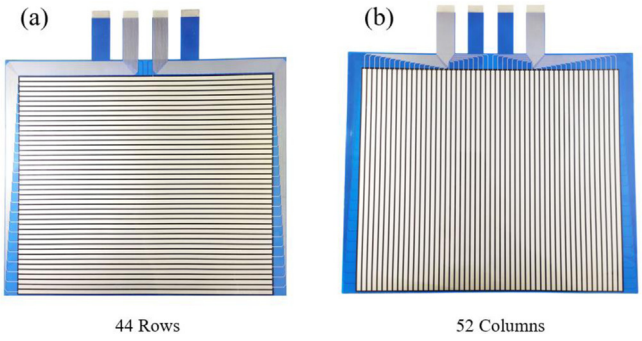



**Fig. 1.** Pressure sensor array (a) top layer and (b) bottom layer.

The advantages of this sensing system are summarized as follows: First, the sensor has a high resolution (2288 sensing points) and a large sensing area (305 mm $\times$ 364 mm). Second, the electrode is led out through the FPC interface (1 mm pitch), and the lead wire in length is 70 mm. Third, the weight range of each sensing point is 1 kg, the minimum induction pressure is 20 g, and the response time is less than 10 us. The time to collect a frame of data with the data acquisition card (IMMFPDB, I-MOTION) is 30 ms. The data acquisition card performs ADC on the data of each sensing point and outputs 8-bit digital signals proportional to pressure. The hysteresis error of a sensing point is 10 %, and the parasitic capacitance is 15 pF. The working temperature of the sensor is from -20 °C to 60 °C. The sensor does not generate electromagnetic radiation, has no electrostatic induction and complies with RoHS requirements.

### 2.2. Data transmission

In our system, the pressure data is transmitted to the computer or other embedded platforms for processing by the data acquisition card through USB communication. According to the communication protocol, we developed a sitting posture analysis software with LabVIEW 2018 on the computer so that the data can be visualized and stored. Since the interface call is provided to Python in LabVIEW 2018, the data can be classified by the algorithm written by Python and the classification result is returned to the software as shown in Fig. 2. For improve the portability and practicability of the system, the Linux driver of the data acquisition card was developed. Therefore, the sitting posture data can be transmitted to the embedded platform for processing by the data acquisition card in real time.

### 2.3. Haptic function design

The feedback part is composed of four flat motors and a motor drive circuit. The diameter of the vibration motor is 10 mm and the height is 3 mm. The vibration motor is embedded in a sponge with a thickness of 20 mm and the same size as the sensor array (335 mm $\times$ 405 mm). Then the sponge with the motor embedded and the pressure sensor are encapsulated by fabric as shown in

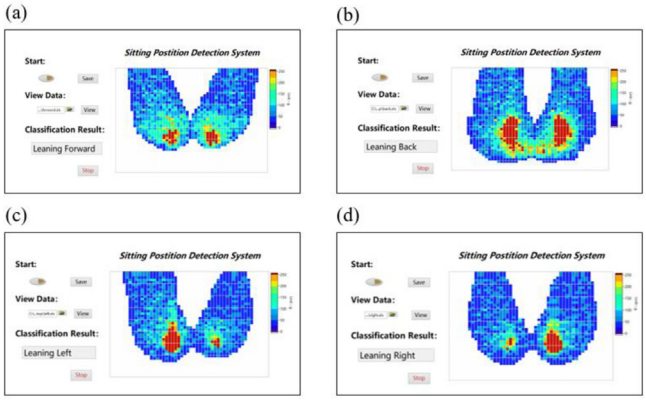



**Fig. 2.** The dynamic pressure distribution map (a) leaning forward, (b) leaning backward, (c) leaning left, and (d) leaning right.

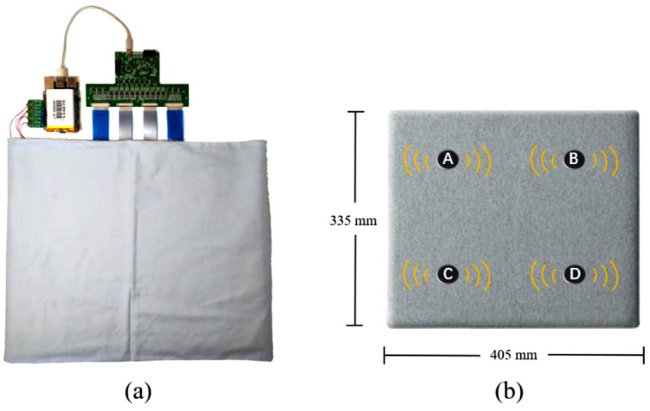



**Fig. 3.** Smart cushion package (a) The hardware composition of the smart cushion; (b) vibration motor locations.

Fig. 3(a). In a typical sitting position, the shoulder, height width and hip ranges from 375 to 505 mm, 505–646 mm and 310–405 mm, respectively [29]. The human sitting position covers 80 % of a seat area amounting to 350 × 300 mm [22]. The sensor used in the smart cushion has a width of 405 mm and a height of 335 mm, which can obtain complete sitting posture data. The positions of the four flat motors are shown in Fig. 3(b) and represented by the letters A, B, C and D. The horizontal spacing between motors is 200 mm and the vertical spacing is 160 mm. This was determined from pilot studies that asked participants to distinguish between points of vibrotactile feedback in the smart cushion.

The objective of the smart cushion system is to use vibrotactile feedback as a medium for encouraging upright sitting posture and reducing bad postures [29]. The feedback process is shown in Fig. 4. In stage one, the user's sitting posture is sensed and classified by the smart cushion. In stage two, the user begins to slouch, the smart cushion system detects this as a poor sitting posture based on sitting posture classification algorithm. In stage three, the user receives vibration feedback from two motors C and D based on the posture detected in stage two. This vibration pulses until the user makes corrections to their posture. Finally, in stage four the user returns to the upright position and the vibration stops. In the case where the user is in an upright position or is not sitting down, no vibration is applied. The power supply voltage of the vibration motor is 5 V. The vibration pulse is generated by the Raspberry Pi 4B, the duty cycle of the pulse is 70 % and the period is 4 s. Each sitting posture corresponds to the motor vibration in different positions, as shown in Table 1.

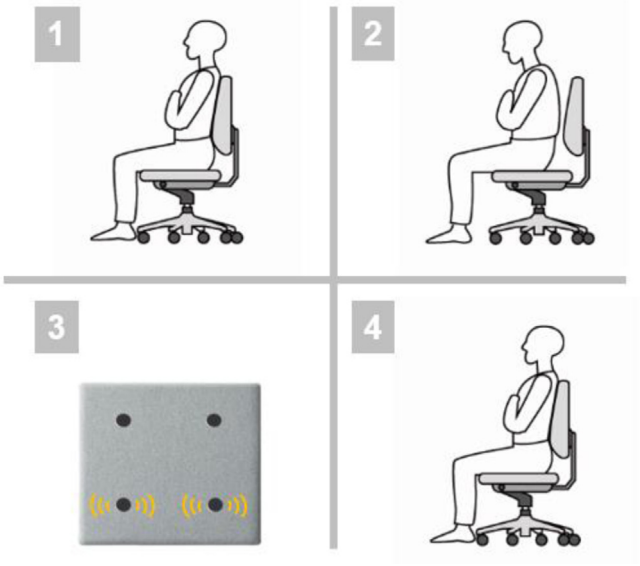



**Fig. 4.** System feedback loop: (1) sensing sitting posture, (2) detecting poor sitting posture, (3) vibration feedback to remind users, (4) user corrects their posture and vibration stops.

**Table 1**
Posture Classification and Haptic Feedback.

| No. | Posture | Vibration Pattern | Samples of Posture |
|---|---|---|---|
| 1 | Sitting Upright | No Activation | 3000 |
| 2 | Leaning Forward | C and D | 3000 |
| 3 | Leaning Backward | A and B | 3000 |
| 4 | Leaning Left | A and C | 3000 |
| 5 | Leaning Right | B and D | 3000 |
| 6 | Cross Left Leg | A, B and C | 3000 |
| 7 | Cross Right Leg | A, B and D | 3000 |

### 2.4. Architecture of the proposed system

The system architecture of the sitting posture monitoring system is as shown in Fig. 5. This system is mainly composed of three parts: sensing control unit, portable embedded system unit and data visualization unit. The sensing control unit collects the user's sitting posture related pressure data, and then transmits the data to the back-end for processing via USB. The portable embedded system unit classifies the sitting posture data and feeds back the classification results through motor vibration. The data visualization unit is designed to facilitate sitting posture data display, storage and analysis.

## 3. Sitting posture evaluation model

### 3.1. Posture definition and dataset collection

The proposed sitting posture monitoring system can recognize seven sitting postures, as shown in Fig. 6. The correct sitting posture should be that the back is naturally straight, the chest is open, and the shoulders are flat as shown in Fig. 6(a). After reading, writing, and sorting materials for a long time, you will lean forward without knowing it, as shown in Fig. 6(b). This posture seems to feel more comfortable. In fact, the pressure on the waist is greater than when sitting upright, which can overwhelm the waist muscles, causing pain and cramps. Fig. 6(c) shows a sitting posture with the body leaning back and the lower back hanging in the air. The lumbar spine is in a kyphotic state without support. The ligaments are in a relaxed state, and the spine is prone to sequence deformation. As shown in Fig. 6(d–e), sitting with the body tilted to the left or right

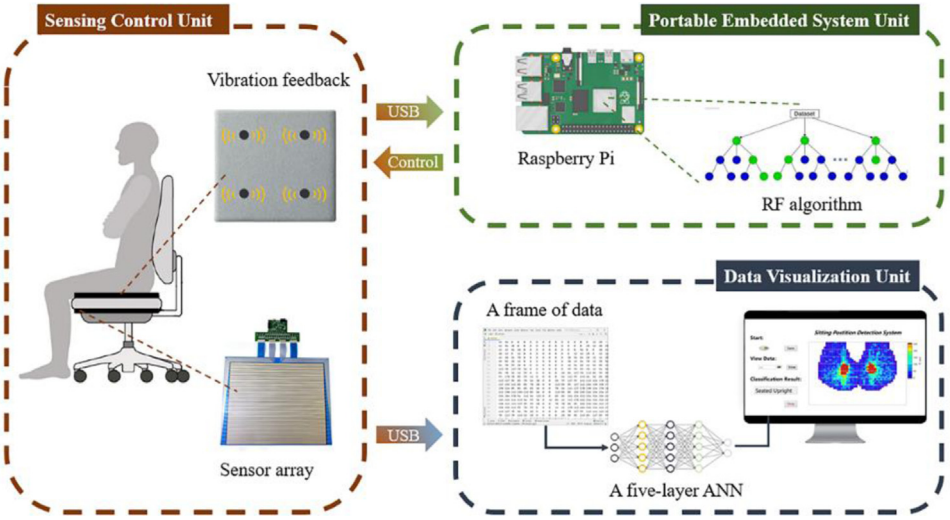



**Fig. 5.** Architecture of the proposed system.

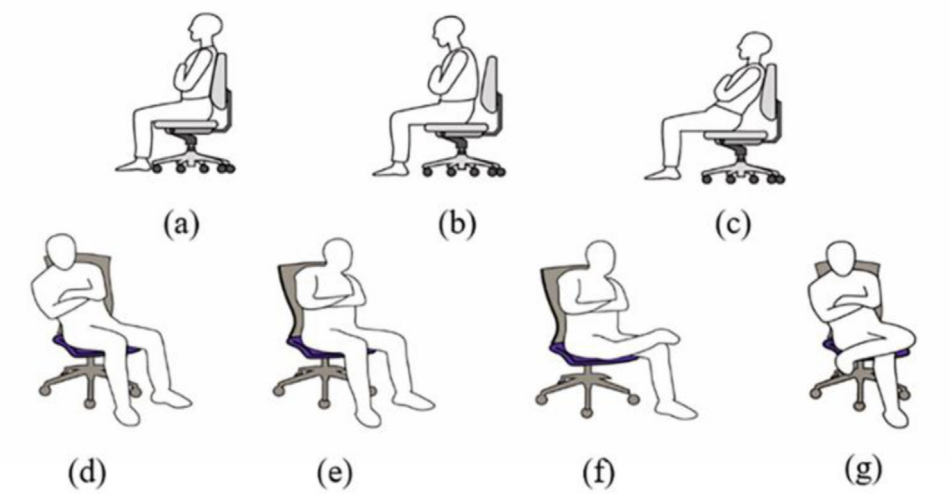



**Fig. 6.** Schematic graph of seven sitting positions (a) sitting upright, (b) leaning forward, (c) leaning backward, (d) leaning left, (e) leaning right, (f) cross left leg, and (g) cross right leg.

can cause the spine to deform easily due to the change of the body's center of gravity. Sitting with your legs crossed can temporarily relax the lower limbs and plantar muscles as shown in Fig. 6(f–g), but habitually the sitting posture for a long time will cause uneven pressure distribution in the intervertebral disc, as well as strain and strain on the waist muscles. The above selection of sitting posture category is from the suggestion of 2 doctors.

Applying machine learning algorithms to recognize sitting posture requires a large number of data to train the model. Sitting posture data is collected from 100 young subjects (63 males, age mean ± standard deviation: 26.81 ± 3.66 years, range: 22–43 years, weight mean ± standard deviation: 62.49 ± 12.5 kg, range: 40–88 kg and height mean ± standard deviation: 167.14 ± 7.21 cm, range: 154–183 cm). The selected subjects have full-time office work, and they have to sit in chairs for long periods of time, but they did not have musculoskeletal pathology or muscle injury. This experiment is conducted in an office environment. We instructed each subject to sit on the smart cushion to complete the above seven types of sitting postures. Subjects were required to maintain each type of sitting posture for more than 10 s. The sensing array was composed of 2248 sensing points, and the voltage values of all sensing points were collected as a frame of data. The sampling frequency of the data acquisition card is 33 Hz. The data was transferred to the computer via a USB cable. We randomly sample 30 frames of data for each sitting posture of the subjects as a data set of data (21,000 frames in total).

### 3.2. Raw data processing

Since the classification of sitting posture mainly depends on the relative pressure distribution on the surface of the cushion, each frame of data is normalized by the following equation:

$$f_k' = \frac{f_k - f_{min}}{f_{max} - f_{min}} \tag{1}$$

where $f_{min}$ and $f_{max}$ represent the minimum and maximum of a frame of data, respectively. $f_k$ represents data of the $k$th sensing point, and $f_k'$ denotes the normalized data.

We visualize the normalized data as shown in Fig. 7(a). It can be seen from the figure that even though the sensor is calibrated before use, the sensor still records some noise. These noises recorded by the sensor arising from flicker noise, mechanical and stretching stimuli, temperature changes and calibration errors [30]. In addition, there are many missing values on the sitting pressure map. This is because the large number of sensing points on the sensor makes the sensing array very dense and the uneven distribution

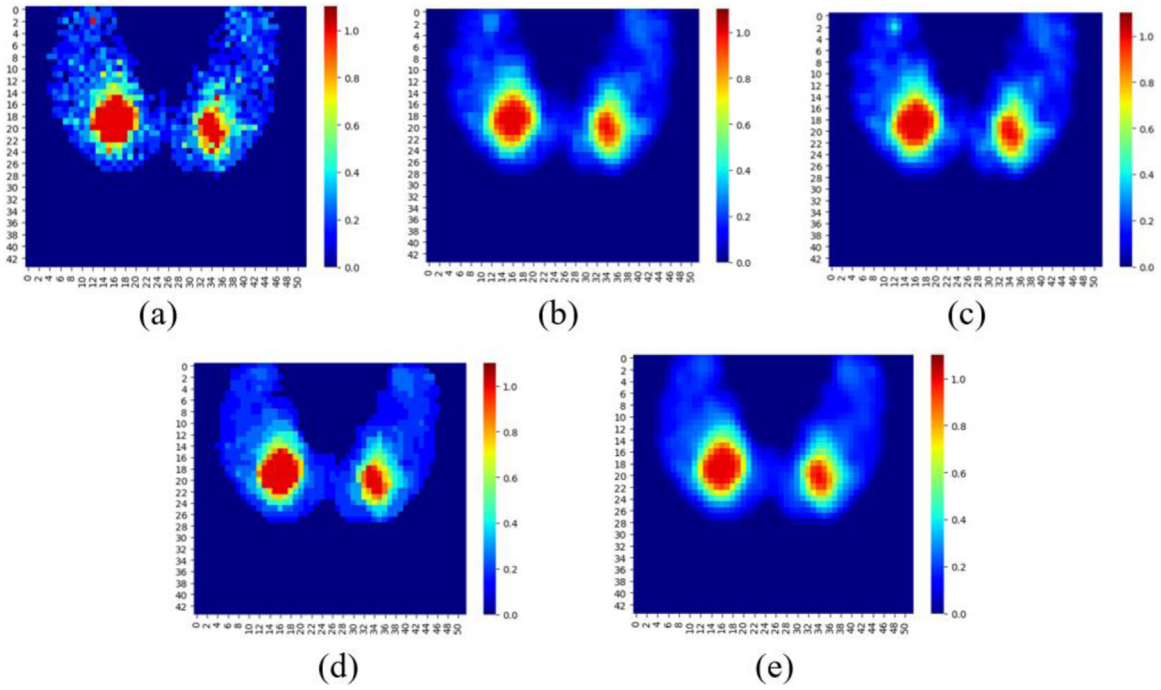



**Fig. 7.** Sitting image filtering (a) Original sitting image, (b) Mean filtered image, (c) Gaussian filtered image, (d) Median filtered image, and (e) Median-Gaussian filtered image.

of pressure exerted on the sensor, which makes individual sensing points unable to receive force. The performance of a model trained with a lot of noisy data will be affected. Therefore, it is necessary to remove noise before the data is used.

We compared the noise reduction effects of mean filter, median filter and Gaussian filter. The specific effect is shown in Fig. 7. The larger the size of the filter kernel, the more blurred the original image will be. For maintain more details of the original image, the kernel size is set to $3 \times 3$. Although mean filtering and Gaussian filtering can compensate for missing values, spike noise cannot be effectively filtered out. For example, the spike noise in the upper left corner of Fig. 7(a) is not filtered out by median filter and Gaussian filter, especially the Gaussian filter is more obvious as shown in Fig. 7(c). In comparison, median filtering has a good performance not only to compensate for missing values but also to filter out spike noise, while it does not smooth the edges of the image as shown in Fig. 7(d). Therefore, we first perform median filtering on the data to filter out noise, and then perform Gaussian filtering to smooth the image.

### 3.3. Classification algorithm description

Machine learning algorithms have already been applied to classify sitting posture [31–34]. Since different classifiers have different performance on the same data set, seven machine learning algorithms are applied for the recognition of sitting postures. Specifically, we compared k-Nearest Neighbor, Logistic Regression, Decision Tree, Support Vector Machine, Naive Bayes, Random Forest, and Artificial Neural Network, respectively, and the related principle is as follows:

*k-Nearest Neighbors algorithm (KNN)* is a simple supervised learning method used for classification and regression. The basic idea of KNN is that if most of the $k$ most similar instances of an instance in the feature space belong to a certain category, the instance also belongs to this category [34]. The parameter $k$ plays the role of a capacity control. The optimal choice of the value $k$ is highly data-dependent. In general, a larger $k$ suppresses the effects of noise, but makes the classification boundaries less distinct. We set the value of $k$ as 3, 5, and 7 respectively and perform 10-fold cross-validation. According to the simulation result, the optimal $k$ value is 3.

*Logistic Regression (LG)* is based on linear regression, which fits the data to a logit function. In the process of LG model learning, maximum likelihood estimation is used to estimate model parameters [35]. In scikit-learn, the LogisticRegressionCV function implements LG with builtin cross-validation to find out the optimal regular parameter $c$. The parameter $cv$ of this cross-validation is set to 5 based on experience.

*Decision Tree (DT)* is a tree structure algorithm, which predicts the value of target variable by learning decision rules inferred from data features [36]. We use the information gain to measure the quality of a split. Since our dataset is relatively small, there the maximum depth of the subtree was not considered.

*Naive Bayes (NB)* is a supervised learning algorithm based on Bayes' theorem, which is a probabilistic classifier based on the assumption of conditional independence among the predictive attributes given the class [37]. In the experiment, we selected a multinomial NB classifier suitable for classification with discrete features.

*Support Vector Machine (SVM)* is a supervised machine learning algorithm used for classification or regression, which maps data to a high-dimensional feature space and finds a hyperplane to create a boundary between data types [38]. We built a linear kernel SVM with lower computational overhead. The parameter $c$ was set as 0.1, 1 and 10 to perform 10-fold cross-validation and the optimized value is 1.

*Random Forest (RF)* is an algorithm that integrates multiple trees through the idea of ensemble learning, and assigns the category with the most votes as the final output [39]. We set the number of trees in the forest to 10, 25, 50, 70 and 100 respectively. After 10-fold cross-validation, it is found that the saturation point is appeared at about 25 trees.

*Artificial Neural Network (ANN)* is widely used in pattern recognition, signal processing and automatic control. The back propagation

**Table 2**
Machine learning algorithm parameter configuration.

| No. | Classifier | Parameters |
|---|---|---|
| 1 | KNN | *k = 3* |
| 2 | LG | *cv = 5* |
| 3 | DT | *Criterion = 'entropy'* |
| 4 | NB | *default* |
| 5 | SVM | *C = 1, kernel='linear'* |
| 6 | RF | *n_estimators = 25* |
| 7 | ANN | *batch size = 128, epoch = 30* |

**Table 3**
Performance of classifiers.

| Classifier | Accuracy | Precision | Recall | F1-score | Model Build Time (s) |
|---|---|---|---|---|---|
| DT | 94.14 % | 0.942 | 0.942 | 0.941 | 17.95 |
| RF | 96.32 % | 0.964 | 0.963 | 0.964 | 4.63 |
| Bayes | 53.57 % | 0.578 | 0.535 | 0.543 | 0.08 |
| ANN | 97.07 % | 0.970 | 0.971 | 0.971 | 13.17 |
| KNN | 96.59 % | 0.966 | 0.966 | 0.965 | 1.80 |
| LG | 95.97 % | 0.959 | 0.960 | 0.960 | 546.20 |
| SVM | 96.49 % | 0.965 | 0.964 | 0.965 | 72.56 |

(BP) neural network plays an important role in practical applications with its simple structure and direct realization [40]. We built a five-layer BP neural network as the classifier in which we set the feed batch size as 128 and the epoch as 30.

The parameter settings of all classifiers are summarized in Table 2. Each classifier was evaluated using the same dataset. The data set has seven label categories as shown in Table 1. 70 % of the data set is used as the training set to train the classification model, and the remaining 30 % is used as the test set to evaluate the model.

### 3.4. Model performance evaluation

After the model was trained, the performance needs to be analyzed, and the model with the best performance was selected according to the evaluation parameters for sitting posture classification. These models were evaluated using four indicators: accuracy, precision, recall and F-measure, calculated as follows:

$$Accuracy = \frac{Tp + Tn}{Tp + Tn + Fp + Fn} \tag{2}$$

$$Precision = \frac{Tp}{Tp + Fp} \tag{3}$$

$$Recall = \frac{Tp}{Tp + Fn} \tag{4}$$

$$F1\,score = 2 \times \frac{precision \times recall}{precision + recall} \tag{5}$$

where *Tp* (true positives) represents samples that are correctly classified in positive examples and *Tn* (true negatives) represents samples that are correctly classified in negative examples. *Fn* (false negatives) means that the positive sample was incorrectly classified as negative and *Fp* (false positives) means that the negative sample was incorrectly classified as positive.

We used 70 % of the data as the training set, and the remaining 30 % as the test set. Each algorithm is run ten times, and the ten classification results are averaged. The performance of these seven models is summarized in Table 3. It can be seen that the accuracy of ANN is the highest (97.07 %), followed by the KNN accuracy of 96.59 %. KNN requires less time to build the model, but it requires high computation load for classification. Since we only care about the classification accuracy of the sitting posture analysis software without considering the hardware resource consumption of the model, the five-layer ANN model is deployed in the sitting posture analysis software.

**Table 4**
The performance of the seven algorithms on seven sitting postures.

| Posture | Accuracy | | | | | | |
|---|---|---|---|---|---|---|---|
| | DT | RF | Bayes | ANN | KNN | LG | SVM |
| Sitting Upright | 0.896 | 0.928 | 0.272 | 0.937 | 0.924 | 0.919 | 0.924 |
| Leaning Forward | 0.913 | 0.941 | 0.437 | 0.964 | 0.947 | 0.943 | 0.947 |
| Leaning Backward | 0.917 | 0.937 | 0.437 | 0.968 | 0.938 | 0.934 | 0.938 |
| Leaning Left | 0.956 | 0.977 | 0.733 | 0.976 | 0.978 | 0.974 | 0.978 |
| Leaning Right | 0.939 | 0.968 | 0.634 | 0.955 | 0.976 | 0.960 | 0.976 |
| Cross Left Leg | 0.984 | 0.995 | 0.823 | 0.993 | 0.996 | 0.993 | 0.996 |
| Cross Right Leg | 0.986 | 0.999 | 0.715 | 0.999 | 0.997 | 0.992 | 0.997 |

The accuracy of each sitting posture is shown in Table 4. We can see that the cross left and right leg have the highest recognition accuracy, and the accuracy of many algorithms have reached more than 99 %. The reason is mainly attributed to that lifting the thigh will cause the original side of the force to lose pressure, which is quite different from other sitting images. Algorithms also have good classification accuracy for the two types of sitting postures, leaning left and leaning right, and most of the algorithms have an accuracy of more than 96 %. The accuracy of the algorithms for sitting upright, leaning forward and leaning backward is relatively low. Most algorithms have an accuracy of about 92 % for sitting upright, and an accuracy of about 94 % for leaning forward and leaning backward, which could be explained by that the two sitting postures only move the center of gravity forward or backward compared to sitting upright, making the difference between these three types of sitting postures less obvious.

## 4. System performance optimization

### 4.1. Sensor array size versus algorithm accuracy

The system cost is one of the significant factors to be considered for large-scale commercialization of products. Although the $44 \times 52$ sensor array in the system can reflect the user's sitting posture very well, the application of large-scale sensor arrays means that hardware costs will inevitably increase. To reduce the hardware cost and system power consumption, we optimized the size of the sensor array.

By collecting data in every other row and every other column as shown in Fig. 8, the original $44 \times 52$ sensor array was reduced to $22 \times 26$, $11 \times 13$, $6 \times 7$ and $3 \times 4$, respectively. A $3 \times 4$ size image of a sitting posture is shown in Fig. 8(e). In the sensor array size of $3 \times 4$, the sensor points are too scattered and the surrounding sensor points have not collected pressure data, so that comprehensive sitting posture information cannot be obtained. When the size of the sensor array increased to $6 \times 7$, a substantially complete sitting posture image is obtained as shown in Fig. 8(f). This is because the $6 \times 7$ size sensor array adds 30 sensing points compared to the $3 \times 4$ size to get more pressure data. When the sensor array continues to expand, the edge details of the sitting image can be much clearly reflected as shown in Fig. 8(g–h). We use ten-fold cross-validation to test the accuracy of the algorithm. The relationship between the size of the sensor array versus the algorithm accuracy is shown in Fig. 9. It can be seen that the accuracy increases as the sensor array increases. After the sensor array size was expanded to $11 \times 13$, the accuracy of these algorithms began to saturate. With small array sizes of $3 \times 4$ and $6 \times 7$, RF, DT and KNN show good performance. When the sensor array size is expanded from $3 \times 4$ to $6 \times 7$, the accuracy of all classifiers is greatly improved. When the sensor array size is $6 \times 7$, the accuracy of KNN is 95.64 %, and RF is 95.75 %. When the sensor array expanded to $11 \times 13$, the best per-

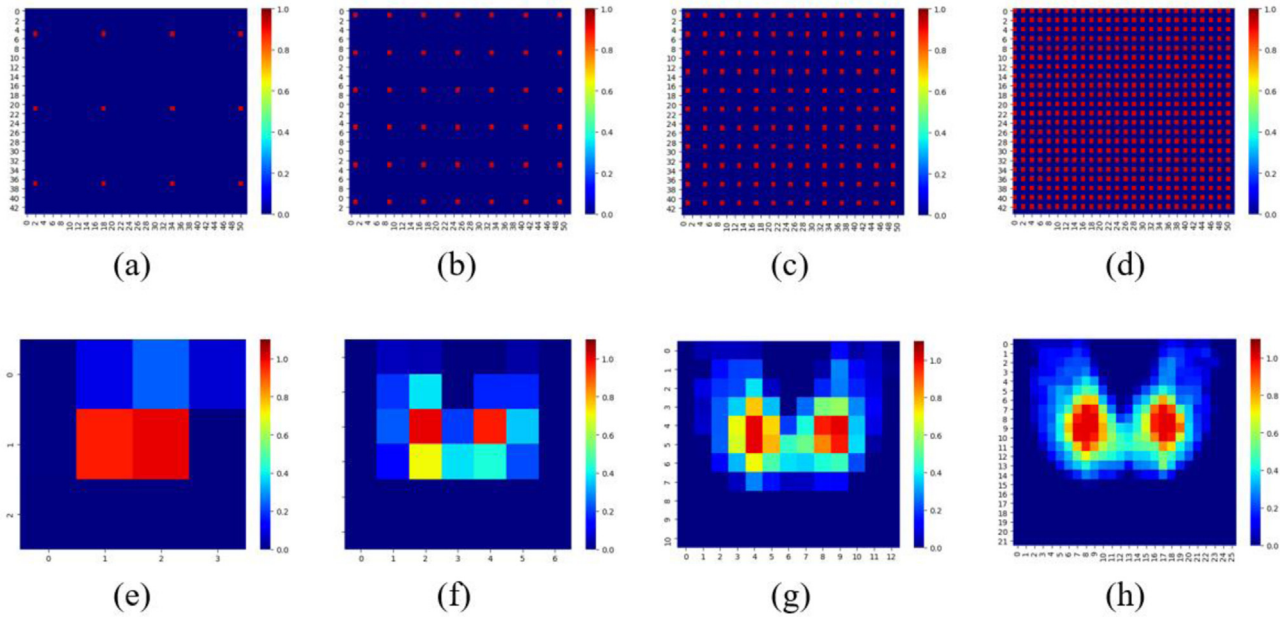



**Fig. 8.** Sensor array size and the related images (a,e) 3 × 4, (b,f) 6 × 7, (c,g) 11 × 13, (d,h) 22 × 26.

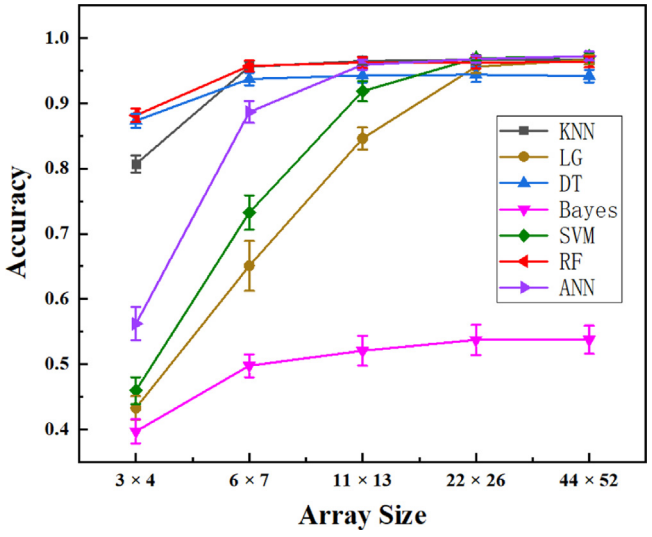



**Fig. 9.** Relationship between array size and accuracy.

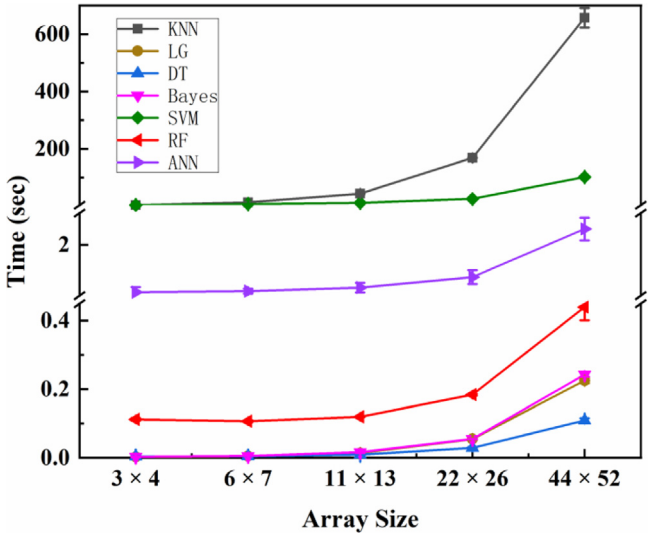



**Fig. 10.** Relationship between array size and prediction time.

formance is KNN with an accuracy of 96.5 %, followed by RF with an accuracy of 96.26 %. The accuracy growth rate of most algorithms began to slow down. When the array continues to expand to 22 × 26 and 44 × 52, the best performing algorithms are SVM and ANN, respectively, with accuracy of 96.93 % and 97.25 %. On the whole, RF and KNN perform well when the array size related to 11 × 13, 6 × 7 or 3 × 4. In contrast, SVM and ANN have higher accuracy when the size of sensor array is large such as 44 × 52 and 22 × 26.

### 4.2. Sensor array size versus hardware resource consumption

The sitting posture monitoring system not only has to work in computer, but also in the embedded systems of Raspberry Pi. Therefore, the consumption of hardware resources running algorithms should be considered. Since the algorithm model only needs to be trained once, we do not consider the modeling time and merely focus on the prediction time. We use 6300 frames test data to test the prediction time of each model on the Raspberry Pi, and each model is tested ten times.

The relationship between the size of sensor array and prediction time is shown in Fig. 10. The prediction time increases as the array size increases, and the prediction time of different algorithms varies significantly. The prediction time using DT, RF, LG and NB is much shorter. Among them, the prediction time using DT is 0.109 s in array sizes of 44 × 52. The prediction time using ANN, KNN and SVM is relatively long, especially SVM and KNN. When the array size increase to 44 × 52, the prediction time of SVM and KNN is 101.84 s and 657.28 s, respectively. Therefore, they are not suitable for the real-time prediction on the embedded platforms. ANN and SVM show higher accuracy, 97.25 % and 97.17 % respectively. However, their accuracy decreases sharply when the sensor array become small. When the size of sensor array is 6 × 7, the accuracy of RF is 95.75 %, ANN is 88.65 %, and SVM is 73.25 %. From the perspective of the consumption of hardware resources and prediction accuracy, DT takes the shortest time and the consumption of

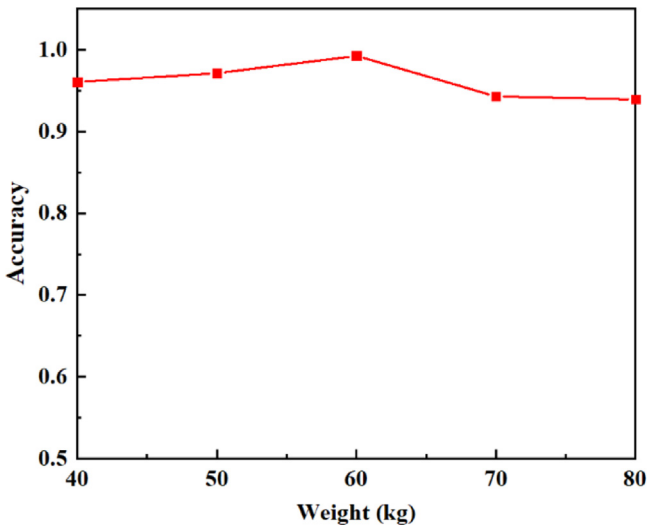



**Fig. 11.** Relationship between prediction accuracy and weights.

hardware resources is the least but the highest accuracy of DT is 94.29 %. When the array size is $6 \times 7$ and $11 \times 13$, the prediction time of RF is 0.1017s and 0.1199s, with time difference 0.0182 s. The corresponding accuracy of the two is 95.75 % and 96.26 %, with a difference of 0.51 %. Moreover, the accuracy of most algorithms only begins to saturate after the array size is $11 \times 13$, as shown in Fig. 9. Hence, the $11 \times 13$ sensor array combined with RF algorithm were selected as the optimal combination. The accuracy of the optimal combination is 96.26 %, and the prediction time on Raspberry Pi system is as short as 19 us for one frame.

### *4.3. Generalization performance of optimal combination*

Sitting posture classification is based on the relative pressure distribution on the sensor array. The size ($44 \times 52$) of each frame of the sitting data set (21,000 frames in total) is reduced to $11 \times 13$. The data set is then used to train a classification model based on the RF algorithm. To test the performance of the optimal combination on pressure data of different weights, we recruited 5 groups (two persons per group) of volunteers with different weights (40 kg, 50 kg, 60 kg, 70 kg, and 80 kg). These volunteers are all graduate students. They spend more than 8 h sitting in the laboratory every day, but they did not have musculoskeletal pathology or muscle injury. Each participant was extracted 280 frames of sitting posture data as a test set. This data is used as a test set to evaluate the model. As shown in Fig. 11, the classification accuracy of different weight data is above 90 %. The accuracy of the optimal combination is the highest (99.28 %) on the sitting posture data with a weight of 60 kg. The accuracy is relatively low (93.92 %) on data with a weight of 80 kg, which is mainly due to the fact that the data set contains less data above 80 kg. Although the same number of samples were recorded for each different posture, the weight distribution of the data set was not balanced. Nevertheless, the results show that overweight or underweight do not take obvious effect on the recognition accuracy. The average accuracy of the optimal combination on the data of different weights is 96.14 %.

## 5. Conclusions

In summary, we designed a portable and practical sitting posture monitoring system. A high-resolution pressure sensor array was used for data collection. The sitting posture data was classified on the Raspberry Pi through seven algorithms including LG, KNN, SVM, NB, DT, RF and ANN. According to the classification results, the vibration feedback could remind the user to maintain the correct sitting posture. A sitting posture analysis software was developed for the real-time data observation, storage and analysis. Considering from the hardware cost aspect, we optimized the sensor array based on algorithm accuracy and hardware resource consumption. As a result, an optimal combination with an accuracy of 96.26 % and prediction time 19 us is realized using RF algorithm based on a $11 \times 13$ sensor array, which is applicable in the practical situation. The portable sitting posture monitoring system proposed in this paper has the characteristics of high accuracy, low power consumption and good human-computer interaction. In our future work, more compact design will be considered to improve the portability. More kinds of wearable sensors will be considered to enrich the function and application field.

## Author statement

Xu Ran: Writing- Reviewing and Editing, Software.
Cong Wang: machine learning algorithm.
Yao Xiao: Data collection.
Xuliang Gao: Data pre-processing.
Ziyuan Zhu: Supervision
Bin Chen: Validation, Methodology.

## 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.

## Acknowledgments

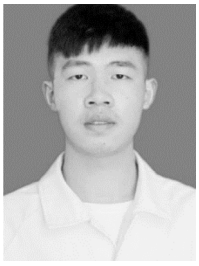
This study was supported by the National Nature Science Foundation of China [No. 61801400] and JSPS KAKENHI [No. JP18F18392].

## References

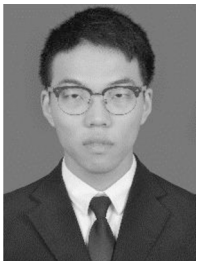
[1] B.H. Hansen, E. Kolle, S.M. Dyrstad, I. Holme, S.A. Anderssen, Accelerometer-determined physical activity in adults and older people, Med. Sci. Sports Exerc. 44 (2012) 266–272, http://dx.doi.org/10.1249/MSS.0b013e31822cb354.
[2] C.E. Matthews, K.Y. Chen, P.S. Freedson, M.S. Buchowski, B.M. Beech, R.R. Pate, R.P. Troiano, Amount of time spent in sedentary behaviors in the United States, 2003-2004, Am. J. Epidemiol. 167 (2008) 875–881, http://dx.doi.org/10.1093/aje/kwm390.
[3] A. Loyen, H.P. van der Ploeg, A. Bauman, J. Brug, J. Lakerveld, European sitting championship: prevalence and correlates of self-reported sitting time in the 28 European Union member states, PLoS One 11 (2016), e0149320, http://dx.doi.org/10.1371/journal.pone.0149320.
[4] J.Y. Chau, A.C. Grunseit, T. Chey, E. Stamatakis, W.J. Brown, C.E. Matthews, A.E. Bauman, H.P. van der Ploeg, Daily sitting time and all-cause mortality: a meta-analysis, PLoS One 8 (2013), e80000, http://dx.doi.org/10.1371/journal.pone.0080000.
[5] A. Biswas, P.I. Oh, G.E. Faulkner, R.R. Bajaj, M.A. Silver, M.S. Mitchell, D.A. Alter, Sedentary time and its association with risk for disease incidence, mortality, and hospitalization in adults: a systematic review and meta-analysis, Ann. Intern. Med. 162 (2015) 123, http://dx.doi.org/10.7326/M14-1651.
[6] W.Y. Wong, M.S. Wong, Smart garment for trunk posture monitoring: a preliminary study, Scoliosis 3 (2008) 7, http://dx.doi.org/10.1186/1748-7161-3-7.
[7] Syazwan A. Azhar, A. Anita, H. Azizan, M. Shaharuddin, Muhaimin Hanafiah, A. Nizar, Baharudin M.R. Juliana, Poor sitting posture and a heavy schoolbag as contributors to musculoskeletal pain in children: an ergonomic school education intervention program, J. Pain Res. (2011) 287, http://dx.doi.org/10.2147/JPR.S22281.
[8] B. Jia, M.A. Nussbaum, Influences of continuous sitting and psychosocial stress on low back kinematics, kinetics, discomfort, and localized muscle fatigue during unsupported sitting activities, Ergonomics 61 (2018) 1671–1684, http://dx.doi.org/10.1080/00140139.2018.1497815.
[9] A.Y.L. Wong, T.P.M. Chan, A.W.M. Chau, H. Tung Cheung, K.C.K. Kwan, A.K.H. Lam, P.Y.C. Wong, D. De Carvalho, Do different sitting postures affect spinal biomechanics of asymptomatic individuals? Gait Posture 67 (2019) 230–235, http://dx.doi.org/10.1016/j.gaitpost.2018.10.028.

[10] P. Waongenngarm, A.J. van der Beek, N. Akkarakittichoke, P. Janwantanakul, Perceived musculoskeletal discomfort and its association with postural shifts during 4-h prolonged sitting in office workers, Appl. Ergon. 89 (2020) 103225, http://dx.doi.org/10.1016/j.apergo.2020.103225.
[11] R. Baker, P. Coenen, E. Howie, A. Williamson, L. Straker, The short term musculoskeletal and cognitive effects of prolonged sitting during office computer work, Int. J. Environ. Res. Public Health 15 (2018) 1678, http://dx.doi.org/10.3390/ijerph15081678.
[12] P. Paliyawan, C. Nukoolkit, P. Mongkolnam, Prolonged sitting detection for office workers syndrome prevention using kinect, in: 2014 11th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol. ECTI-CON, IEEE, Nakhon Ratchasima, Thailand, 2014, pp. 1–6, http://dx.doi.org/10.1109/ECTICon.2014.6839785.
[13] J.C.T. Mallare, D.F.G. Pineda, G.M. Trinidad, R.D. Serafica, J.B.K. Villanueva, A.R. Dela Cruz, R.R.P. Vicerra, K.K.D. Serrano, E.A. Roxas, Sitting posture assessment using computer vision, in: 2017 IEEE 9th Int. Conf. Humanoid Nanotechnol. Inf. Technol. Commun. Control Environ. Manag. HNICEM, IEEE, Manila, Philippines, 2017, pp. 1–5, http://dx.doi.org/10.1109/HNICEM.2017.8269473.
[14] Lan Mu, Ke Li, Chunhong Wu, A sitting posture surveillance system based on image processing technology, in: 2010 2nd Int. Conf. Comput. Eng. Technol., IEEE, Chengdu, China, 2010, pp. V1-692–V1-695, http://dx.doi.org/10.1109/ICCET.2010.5485381.
[15] S. Bei, Z. Xing, L. Taocheng, L. Qin, Sitting posture detection using adaptively fused 3D features, in: 2017 IEEE 2nd Inf. Technol. Netw. Electron. Autom. Control Conf. ITNEC, IEEE, Chengdu, 2017, pp. 1073–1077, http://dx.doi.org/10.1109/ITNEC.2017.8284904.
[16] A.S. Fiorillo, C.D. Critello, S.A. Pullano, Theory, technology and applications of piezoresistive sensors: a review, Sens. Actuators Phys. 281 (2018) 156–175, http://dx.doi.org/10.1016/j.sna.2018.07.006.
[17] L. Shang, C. Liu, M. Watanabe, B. Chen, K. Hayashi, LSPR sensor array based on molecularly imprinted sol-gels for pattern recognition of volatile organic acids, Sens. Actuators B Chem. 249 (2017) 14–21, http://dx.doi.org/10.1016/j.snb.2017.04.048.
[18] M. Kim, H. Kim, J. Park, K.-K. Jee, J.A. Lim, M.-C. Park, Real-time sitting posture correction system based on highly durable and washable electronic textile pressure sensors, Sens. Actuators Phys. 269 (2018) 394–400, http://dx.doi.org/10.1016/j.sna.2017.11.054.
[19] T. Fu, A. Macleod, IntelliChair: an approach for activity detection and prediction via posture analysis, in: 2014 Int. Conf. Intell. Environ., IEEE, China, 2014, pp. 211–213, http://dx.doi.org/10.1109/IE.2014.39.
[20] Hu Yu, A. Stoelting, Yi-Tao Wang, Zou Yi, M. Sarrafzadeh, Providing a cushion for wireless healthcare application development, Ieee Potentials 29 (2010) 19–23, http://dx.doi.org/10.1109/MPOT.2009.934698.
[21] R. Barba, Á.P. de Madrid, J.G. Boticario, Development of an inexpensive sensor network for recognition of sitting posture, Int. J. Distrib. Sens. Netw. 11 (2015) 969237, http://dx.doi.org/10.1155/2015/969237.
[22] A.R. Anwary, D. Cetinkaya, M. Vassallo, H. Bouchachia, Smart-Cover: A real time sitting posture monitoring system, Sens. Actuators Phys. 317 (2021) 112451, http://dx.doi.org/10.1016/j.sna.2020.112451.
[23] S. Ahn, Y. Jeong, D. Kim, H. Kim, Development of the non-wearable system with FSR sensors for correction of sitting position, in: 2015 Second Int. Conf. Comput. Technol. Inf. Manag. ICCTIM, IEEE, Johor, Malaysia, 2015, pp. 140–143, http://dx.doi.org/10.1109/ICCTIM.2015.7224608.
[24] J. Ahmad, H. Andersson, J. Siden, Screen-printed piezoresistive sensors for monitoring pressure distribution in wheelchair, IEEE Sens. J. 19 (2019) 2055–2063, http://dx.doi.org/10.1109/JSEN.2018.2885638.
[25] A.R. Anwary, H. Bouchachia, M. Vassallo, Real time visualization of asymmetrical sitting posture, Procedia Comput. Sci. 155 (2019) 153–160, http://dx.doi.org/10.1016/j.procs.2019.08.024.
[26] J. Yongxiang, D. Jingle, D. Sanpeng, Q. Yuming, W. Peng, W. Zijing, Z. Tianjiang, Sitting posture recognition by body pressure distribution and airbag regulation strategy based on seat comfort evaluation, J. Eng. 2019 (2019) 8910–8914, http://dx.doi.org/10.1049/joe.2018.9143.
[27] Q. Hu, X. Tang, W. Tang, A smart chair sitting posture recognition system using flex sensors and FPGA implemented artificial neural network, IEEE Sens. J. 20 (2020) 8007–8016, http://dx.doi.org/10.1109/JSEN.2020.2980207.
[28] C. Ma, W. Li, R. Gravina, G. Fortino, Posture detection based on smart cushion for wheelchair users, Sensors 17 (2017) 719, http://dx.doi.org/10.3390/s17040719.
[29] K. Ishac, K. Suzuki, LifeChair: a conductive fabric sensor-based smart cushion for actively shaping sitting posture, Sensors 18 (2018) 2261, http://dx.doi.org/10.3390/s18072261.
[30] J. Heikenfeld, A. Jajack, J. Rogers, P. Gutruf, L. Tian, T. Pan, R. Li, M. Khine, J. Kim, J. Wang, J. Kim, Wearable sensors: modalities, challenges, and prospects, Lab Chip 18 (2018) 217–248, http://dx.doi.org/10.1039/C7LC00914C.
[31] C. Ma, W. Li, R. Gravina, G. Fortino, Activity recognition and monitoring for smart wheelchair users, in: 2016 IEEE 20th Int. Conf. Comput. Support. Coop. Work Des. CSCWD, IEEE, Nanchang, China, 2016, pp. 664–669, http://dx.doi.org/10.1109/CSCWD.2016.7566068.
[32] Guanqing Liang, Jiannong Cao, Xuefeng Liu, Smart cushion: a practical system for fine-grained sitting posture recognition, in: 2017 IEEE Int. Conf. Pervasive Comput. Commun. Workshop PerCom Workshop, IEEE, Kona, HI, 2017, pp. 419–424, http://dx.doi.org/10.1109/PERCOMW.2017.7917599.
[33] J. Roh, H. Park, K. Lee, J. Hyeong, S. Kim, B. Lee, Sitting posture monitoring system based on a low-cost load cell using machine learning, Sensors 18 (2018) 208, http://dx.doi.org/10.3390/s18010208.
[34] J. Meyer, B. Arnrich, J. Schumm, G. Troster, Design and modeling of a textile pressure sensor for sitting posture classification, IEEE Sens. J. 10 (2010) 1391–1398, http://dx.doi.org/10.1109/JSEN.2009.2037330.
[35] G. Tripepi, K.J. Jager, F.W. Dekker, C. Zoccali, Linear and logistic regression analysis, Kidney Int. 73 (2008) 806–810, http://dx.doi.org/10.1038/sj.ki.5002787.
[36] S.B. Kotsiantis, Decision trees: a recent overview, Artif. Intell. Rev. 39 (2013) 261–283, http://dx.doi.org/10.1007/s10462-011-9272-4.
[37] P. Bermejo, J.A. Gámez, J.M. Puerta, Speeding up incremental wrapper feature subset selection with Naive Bayes classifier, Knowledge Based Syst. 55 (2014) 140–147, http://dx.doi.org/10.1016/j.knosys.2013.10.016.
[38] G. Liu, D. Zhou, H. Xu, C. Mei, Model optimization of SVM for a fermentation soft sensor, Expert Syst. Appl. 37 (2010) 2708–2713, http://dx.doi.org/10.1016/j.eswa.2009.08.008.
[39] J. Ham, Yangchi Chen, M.M. Crawford, J. Ghosh, Investigation of the random forest framework for classification of hyperspectral data, IEEE Trans. Geosci. Remote Sens. 43 (2005) 492–501, http://dx.doi.org/10.1109/TGRS.2004.842481.
[40] A. Prieto, M. Atencia, F. Sandoval, Advances in artificial neural networks and machine learning, Neurocomputing 121 (2013) 1–4, http://dx.doi.org/10.1016/j.neucom.2013.01.008.

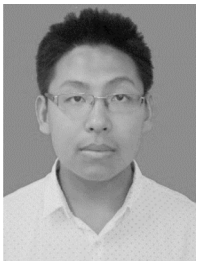
## Biographies



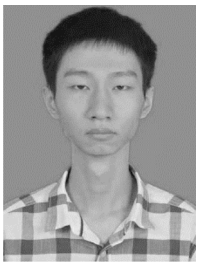
**Xu Ran** received her B.S. degree in the Electronic Information and Electrical Engineering from Chongqing University of Arts and Sciences (China) in 2019. He is now pursuing his master's degree in the Graduate School of Electronic Information Engineering in Southwest University engaging in the research of machine learning and flexible sensing.



**Cong Wang** received his B.S. degree in the Electronic Science and Technology from Southwest University (China) in 2018. He is now pursuing his master's degree in the Graduate School of Electronic Information Engineering in Southwest University engaging in the research of machine learning and deep learning.

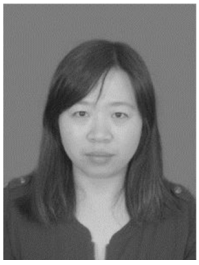


**Yao Xiao** received his B.S. degree in the Automation from China Three Gorges University (China) in 2018. He is now pursuing his master's degree in the Graduate School of Electronic Information Engineering in Southwest University engaging in the research of computer vision and deep learning.



**Xuliang Gao** received his B.S. degree in the Electronic Information Engineering from Chongqing Three Gorges University (China) in 2020. He is now pursuing his master's degree in the Graduate School of Electronic Information Engineering in Southwest University engaging in the research of computer vision and deep learning

**Ziyuan Zhu** is currently a professor at Southwest University. He got his B.S in electronic science and technology (microelectronics technology) at university of electronic science and technology of china, Ph.D. in microelectronics and solid state electronics at Peking University, and he was a visiting scholar of Georgia institute of technology (2013–2014). He worked as an assistant professor at Zhejiang University before joining southwest university in 2019. His research involves various issues in self-powered electronics and micro fabrication.



**Bin Chen** received her Ph. D. degree in the Graduate School of Information Science and Electrical Engineering from Kyushu University (Japan) in 2014. She is now an associate professor at the College of Electronic and Information Engineering at Southwest University, and a JSPS fellow work in Kyushu University engaging in the research of sensors, portable gadget and device development, and machine learning.