Research Article

Pruning Growing Self-Organizing Map Network for Human Physical Activity Identification

Lingfei Mo [ID], Hongjie Yu, and Wenqi Hua

School of Instrument Science and Technology, Southeast University, Nanjing 210096, China

Correspondence should be addressed to Lingfei Mo; lfmo@seu.edu.cn

Human physical activity identification based on wearable sensors is of great significance to human health analysis. A large number of machine learning models have been applied to human physical activity identification and achieved remarkable results. However, most human physical activity identification models can only be trained based on labeled data, and it is difficult to obtain enough labeled data, which leads to weak generalization ability of the model. A Pruning Growing SOM model is proposed in this paper to address the limitations of small-scale labeled dataset, which is unsupervised in the training stage, and then only a small amount of labeled data is used for labeling neurons to reduce dependency on labeled data. In training stage, the inactive neurons in network can be deleted by pruning mechanism, which makes the model more consistent with the data distribution and improves the identification accuracy even on unbalanced dataset, especially for the action categories with poor identification effect. In addition, the pruning mechanism can also speed up the inference of the model by controlling its scale.

1. Introduction

Appropriate amount of exercise can promote human blood circulation, help bone and muscle growth, enhance cardiopulmonary function, and also prevent diabetes, hypertension, and other diseases [1, 2]. Therefore, human activity monitoring is necessary for human health management. However, if the assessment of the amount of exercise only depends on the personal impression, the evaluation results are inevitably too subjective. Therefore, a reliable human physical activity identification system is needed to identify the daily human movements. The development of microelectromechanical sensors, Internet of Things technology, and high-speed low-power communication networks makes it easier to implement the task of human physical activity identification by using wearable sensors to form a human sensor network. The wearable device can perform the all-weather and uninterrupted collection of the activity information of the wearing part without affecting the daily activities of the human body. A human sensor network composed of multiple wearable devices can estimate the pose of different parts of the human body as well as the

heart rate or the respiratory rate of the human body [3], and the data collected by these devices can be transmitted to the cloud storage via the network for data analysis [4]. All-weather data collection means that a large amount of human activity data is produced by wearing sensors, which makes it possible to analyze human daily activity and evaluate human health.

In early research, only one sensor is used to identify human physical activity [5–7]. Then, a multisensor system is introduced into this area [8–10]. Also, deep learning [11–13] and ensemble learning [14, 15] have been introduced into this area. However, these research methods based on vision are easily restricted by many factors, such as object occlusion and light conditions, and are not conducive to the protection of users' privacy. What is more, Graph CNN [16–21] has been widely studied in this field, but GNN is generally for image data, video data, or skeleton data, which has the ability of representation learning, but large computational power and training data is necessary. Compared with these methods, sensors with high computing power, small size, and low cost make it possible for people to interact with these devices as a part of their daily life.

The identification task inevitably needs labels to determine the identification results, but unsupervised learning models without labeled data can be used during model training, which will greatly reduce the proportion of demand for labeled data in the process of training and using models. Therefore, unsupervised learning has been introduced to both feature extraction [22] and activity identification [23–27]. Multiple unsupervised learning algorithms such as k-means, Gaussian mixture model (GMM), hierarchical clustering, and density-based spatial clustering of applications with noise (DBSCAN) have been applied to human physical activity identification. But k-means, GMM, and hierarchical clustering all need to have prior knowledge about the number of the classes of the dataset, GMM can only perform well on dataset with Gaussian distribution, DBSCAN is not suitable to dataset with high dimension or uneven density. Besides, it is difficult to identify with a unified model because different people have different physical activity patterns. However, most of the existing models are offline training, so they cannot make adaptive adjustment to the new input data to realize online learning or incremental learning. This defect makes these models cannot perform well on different individuals. The Self-Organizing Map (SOM) network is an unsupervised learning algorithm proposed by the Finnish mathematician Kohonen in 1990 [28]. It is based on Hebb's learning rules and only updates the connection weights between neurons based on activated neurons. Since the algorithm was proposed, there have been various improvements made by scholars. The improvements are aimed at the network structure [29–31] and the learning mechanism [32–35]. The improvement of parameter initialization is also a research target [36, 37]. These rich SOM network variants show that the SOM network is exceedingly scalable. Compared with other classic unsupervised learning algorithms, SOM only needs little prior knowledge about the dataset, and it can deal with high-dimensional data well. The proposed model can perform well in the field of human physical activity identification when the data distribution is complex; the Growing SOM network (GSOM) can achieve the adaptive adjustment of network structure to obtain the most suitable network structure for specific dataset and make the topology of neural network more consistent with the distribution of dataset. Growing SOM has been widely used in the Internet of Things area, including sensor network communication [38], big data analysis [39], and data mining [40]. Growing SOM and its variants had been utilized widely because of their characteristics [41, 42]. It is suitable for tasks when the researchers have no idea about the data distribution [43, 44]. In addition, the characteristics of Growing SOM enable it to generate new network neurons for new samples to identify them and realize incremental learning without losing the original information.

In this paper, a Pruning Growing SOM (PGSOM) model is proposed based on GSOM. After an additional pruning mechanism is introduced, the redundant neurons will be deleted to reduce inference time and the storage space of the model under the premise of ensuring identification accuracy. The introduction of PGSOM not only reduce the dependence of human physical activity identification on labeled data and the prior knowledge of the exact number of the categories, but also adjust the structure of the model to fit the dataset better. Because the training cost of the model is not high and there are few super parameters, a robust model can be trained offline through grid search.

The organization of this article is as follows: firstly, the importance of this research and the related work of this area will be introduced; then the method used in this article will be explained in detail; after that, there is a simulation experiment and a real-world experiment that helps to show the characteristics of the method, and the results will be analysed; at last, a conclusion is proposed about this article.

2. Materials and Methods

2.1. PA Identification Diagram. The human physical activity identification is shown in Figure 1. In the experiment, the wearable device is used to obtain the activity data of different individuals. The input dataset is ready after feature extraction and feature selection, but different individuals' activity patterns are not consistent, and it is expensive to obtain a large quantity of labeled data in reality. What is more, it is difficult to design appropriate topology for different datasets. PGSOM could learn without supervision and be adjusted adaptively according to the scale and dimension of the dataset to reduce the quantization error. Then, a small amount of labeled data is used to calibrate the neurons in the trained network to obtain a model for the activity identification of different individuals.

2.2. SOM Network. The SOM network structure is shown in Figure 2. It is a feedforward neural network composed of an input layer and an output layer. The number of neurons in the input layer is the same as the dimension of the input sample. The output layer is also called the competition layer. Most of the neurons in the layer are arranged in two dimensions. The input layer and the output layer are fully connected networks with variable weights. The samples are input to the network for training, the neuron with the smallest Euclidean Distance from the input sample is the winner, and the weights of the neurons in the neighborhood of the winner are updated. The labeled information is not needed in training. After unsupervised training, a small number of labeled samples can be input to label the activated neurons.

The calibration algorithm is summarized as follows:

2.3. Growing SOM Network. The disadvantage of SOM is that it requires a preset network topology, and a large number of experiments are needed to find an appropriate network topology to the dataset.

The structure of GSOM starts with four neurons and grows on boundary based on heuristics and input representation. GSOM controls the difficulty of network growth by adjusting hyperparameters. When the cumulative quantization error (CEQ) on an activated neuron reaches the growth threshold (GT) and the winner neuron is a boundary
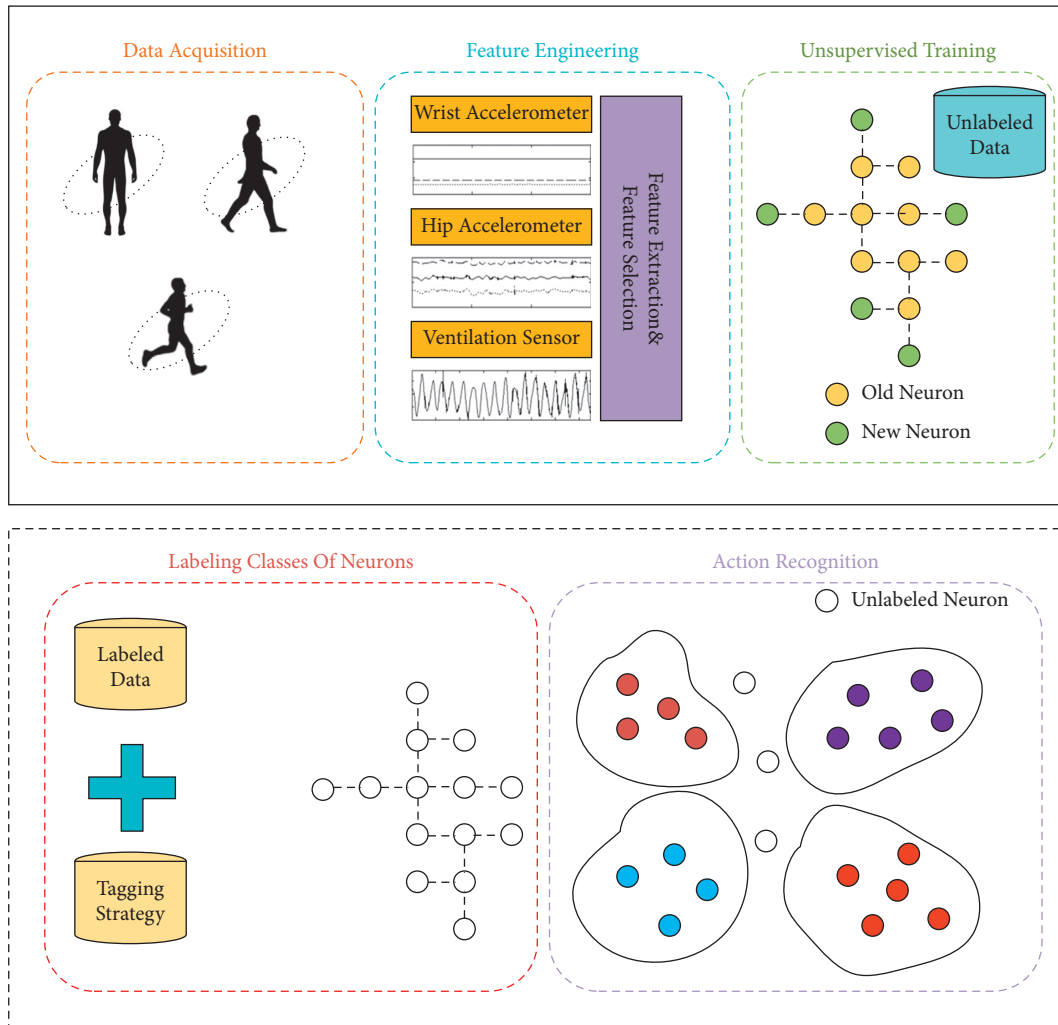
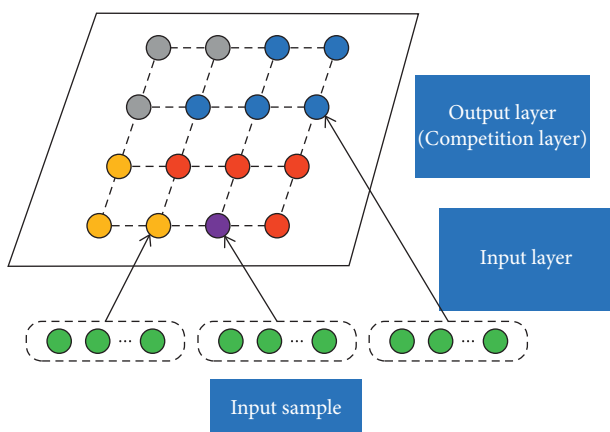FIGURE 1: Human physical activity identification with PGSOM.

FIGURE 2: SOM network structure. Different colours of neurons represent different clusters; each input sample can activate a neuron in output layer.

neuron, it will perform growth operations; GT is a hyperparameter to control the ability about growth. Its CEQ will be halved and distributed to all adjacent neurons
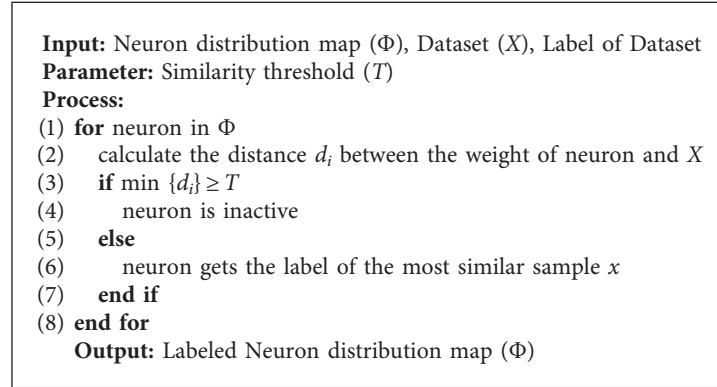
if it is not at the boundary of the network, which is shown in Figure 3.

However, GSOM also faces growth controllability issues. The increasing number of new neurons will contribute to the old neurons less activated because their weight vectors do not correspond to a certain pattern. Therefore, a pruning mechanism needs be introduced to optimize GSOM.

2.4. Pruning Growing SOM Network. Cutting the connection between neurons belonging to different patterns has been proved to be a way to adjust the network structure [45–47]. The key is to find a proper method to decide when a neuron (or the connection between neurons) should be removed.

The Pruning Growing Self-Organizing Map cuts neurons when the samples can activate the neuron being less than a predesigned threshold. When the dataset is unbalanced, samples of one or more categories are significantly less or more than others, the threshold is hard to be designed to make the pruning mechanism able to equally treat neurons represent minority category and majority category.

The TurSOM [48] uses neuron age threshold as the pruning threshold, but due to its mechanism, it prefers to cut

Input: Neuron distribution map (Φ), Dataset (X), Label of Dataset
Parameter: Similarity threshold (T)
Process:
(1) **for** neuron in Φ
(2)   calculate the distance d_i between the weight of neuron and X
(3)   **if** min $\{d_i\} \geq T$
(4)   neuron is inactive
(5)   **else**
(6)   neuron gets the label of the most similar sample x
(7)   **end if**
(8) **end for**
      Output: Labeled Neuron distribution map (Φ)

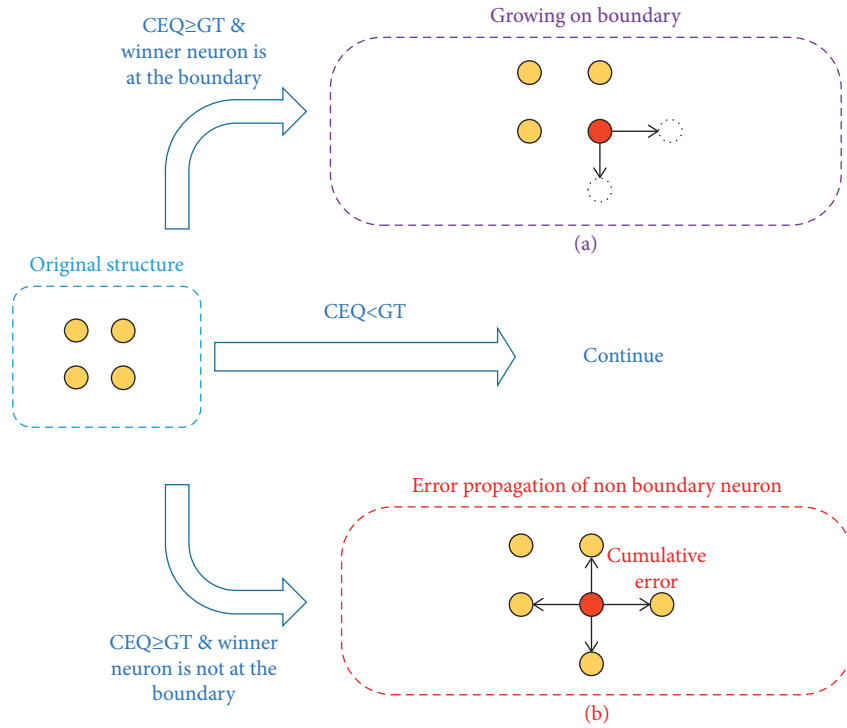ALGORITHM 1: Class Calibration Algorithm.

FIGURE 3: GSOM network structure and related operations. (a) New neurons grow at the dotted line position. (b) The cumulative error of winner neurons is halved and propagated to neurons; red neuron is the winner.

new neurons, not the old neurons. This mechanism will make the newly input knowledge harder to be learned.

Inspired by the SOINN proposed by Furao and Hasegawa [49], a pruning mechanism based on the active degree of the neurons was proposed to improve GSOM. The inactive neurons in the GSOM contain very little useful information and have limited contribution to the performance of network by the theory of SOINN. The number of neurons can be effectively controlled without significantly affecting the accuracy of the model if the mechanism is added.

The algorithm uses neuron's age to describe how much time has passed since the neuron was last activated and it will eliminate neurons that have been inactive for a long time and preserve newborn neurons and the neurons that are often

activated, which makes the model more suitable for learning new information without forgetting the useful information of the past; the pruning operation of PGSOM is shown in Figure 4. After the weight update and growth of the network are completed, all inactive neurons will be deleted.

Once an input sample x is presented to PGSOM, the index of the winner neuron is calculated according to (1) and (2) [28]. Then, the CEQ [29–31] is calculated by (3) and (4). Update the weights of the winner neuron and other neurons in its topological neighborhood according to (4) and (5).

$$d_i = x(t) - w_i^2, \quad i \in \phi(t), \tag{1}$$
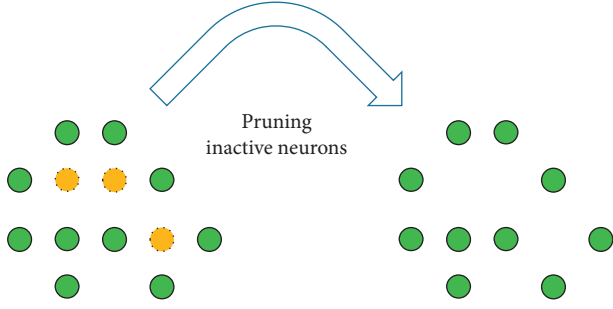
$$\text{winner}(t) = \arg\min\{d_i\}, \tag{2}$$

FIGURE 4: Pruning operation of PGSOM; the yellow neurons are older than age threshold.

$$\text{CEQ}_{\text{winner}(t)} = \text{CEQ}_{\text{winner}(t)} + \eta(t) \times d_i, \tag{3}$$

$$\eta(t+1) = \left(1 - \left(\frac{R}{|\phi(t)|}\right)\right) \times \eta(t), \tag{4}$$

$$w_i(t+1) = w_i(t) + \eta(t) \times \left(w_{\text{winner}(t)}(t) - w_i(t)\right) \times e^{-d_i}, \quad i \in \phi(t). \tag{5}$$

The symbol w is the weight vector of neuron in competition layer; d is Euclidean distance between the weight vector of input sample and the weight vector of competition layer. $\Phi(t)$ is a set of all neurons in competition layer. η is decay factor, which decreases with the increase of iteration, and is used to ensure the training of PGSOM convergence. The network enters the growth stage after updating the weights.

Judging whether the CEQ of the winner neuron exceeds the growth threshold, the growth threshold can be determined by (6), D is the dimension of input sample, and spread factor (SF) is a hyperparameter between 0 and 1 to influence the difficulty of network growth. If the above condition is true and the winner is at the boundary of the network, the new neuron will be used to fill the surrounding of the winner neuron as (7) and (8) [29–31], and the weight of the new neuron is initialized by interpolation. If the winner neuron is not a boundary neuron, the CEQ is halved and the lost error is equally distributed to the neurons around as shown in Figure 3. Finally, the age of new neurons and winner neurons is updated to 0, and ages of the remaining neurons are increased by 1 as (9). At the same time, the neurons whose age exceeded the age threshold are deleted, which can be expressed by (10) and (11).

$$\text{GT} = -D \times \ln(\text{SF}), \tag{6}$$

$$\text{New neuron}(t) = \begin{cases} \text{neurons in remaining neighborhood,} \\ \quad \text{if winner's CEQ} \geq \text{GT and winner is at the boundary,} \\ \{\}, \text{ else,} \end{cases} \tag{7}$$

$$\phi(t) = \phi(t) \bigcup \text{New neuron}(t), \tag{8}$$

$$\text{Age}_i = \begin{cases} 0, & \text{winner or new neuron,} \\ \text{Age}_i + 1, & \text{else,} \end{cases} \tag{9}$$

$$\text{Delete}(t) = \{neu_i | i \in \phi(t) \cap \text{Age}_i > M\}, \tag{10}$$

$$\phi(t+1) = \phi(t) - \text{Delete}(t). \tag{11}$$

The training algorithm is summarized as follows:

3. Results and Discussion

3.1. Dataset Description. A total of 110 volunteers were called for data collection; each volunteer carried 5 to 7 physical activities [50]. The whole statistic characteristics of volunteers are given in Table 1.

The experiment involved a total of 9 common human activities; the label, kind, and the number of activities after feature engineering are shown in Table 2. All physical activity data come from two triaxial acceleration sensors worn on the wrist and hip and a piezo sensor worn on the abdomen, whose sampling frequency is 30 Hz; as shown in Figure 5, multiple raw data are stored in SD card for training model in offline environment. The raw data collected is shown in Figure 6.

A sliding window with length of 500 sampling points [51] is used to divide the raw data into several units. Each unit contains 7 signals combined with 6 signals from 2 triaxial acceleration sensors and one signal from a ventilation sensor; on this basis, we add 2 synthetic acceleration signals. The calculation formula of synthetic acceleration signal is as follows:

$$\text{synthetic signal} = \sqrt[2]{x^2 + y^2 + z^2}. \tag{12}$$

11 features are extracted from each signal, respectively, as shown in Table 3, and the correlation coefficients between the synthetic acceleration signal from hip and piezo signal, the synthetic acceleration signal from wrist and piezo signal are calculated, which forms a 101-dimensional ($11 * 9 + 2 = 101$) dataset.

4. Experiment and Results

One of the most important parameters of model is Age threshold (M), which determines the life cycle of neurons that have not been activated for a long time. The old neurons are more likely to be preserved with M increasing. Another important parameter is SF, which determines the growth threshold (GT) and affects the speed of the growing. 8586 samples are divided into training set and validation set in a ratio of 9 : 1; then, 10-fold cross validation is conducted. In each iteration of training, 4000 samples are randomly selected from 7727 samples for training, and fixed 2000 samples which are selected from current training set for calibration. The 859 samples in the validation set are used to evaluate the performance after each iteration of training. Ratio factor (RF) is the ratio of M to the size of samples in each iteration (batch size); for example, when the size of input in each iteration is 4000 and RF is 1.25, the M of

(i) **Input:** Dataset (X)
   Parameter: Spread factor (SF), Iterations, Age threshold (M), $R = 2.8$, $\eta (0) = 1$
   Process:
(1) **initialize the network as shown in** Figure 3 **and set SF with an interval of (0, 1), all the age of neurons set 0, calculating the GT according to** (1), **set iteration = 0**
(2)   **while iteration < iterations do**
(3)   **for x in random subset from X**
(4)   **present the input sample x (t) to the Neuron distribution map**
(5)   **get the winner by** (1) **and** (2), **update the CEQ of winner by** (3)
(6)   **update the weights of competitive neurons by** (4) **and** (5)
(7)   **if the CEQ of winner ≥ GT as** (6) **and winner is at the boundary**
(8)   **carry out growth operation by** 7 **and** 8 **and** Figure 3
(9)   **end if**
(10)   **update the age of competitive neurons by** (9)
(11)   **prune the network by** (10) **and** (11)
(13)   **end for**
(14)   **iteration = iteration +1**
(15)   **end while**
   Output: Neuron distribution map (Φ)

ALGORITHM 2: Pruning Growing Self-Organizing Map.

TABLE 1: Volunteers' information.

Information	Distribution Category	Number	Time duration of each activity	Percentage
Gender	F	59	53.6%	
	M	51	46.4%	
Age (years)	20–30	30	27.3%	
	30–40	28	25.5%	
	40–50	25	22.7%	
	50–60	27	24.5%	
Mass (kg)	<50	2	1.8%	
	50–60	26	23.7%	
	60–70	34	30.9%	
	70–80	13	11.8%	Each volunteer performs 5–7 kinds of physical activities for 5 minutes and rest for 2 minutes after each exercise to lower heart rate
	80–90	21	19.1%	
	>90	14	12.7%	
Height (cm)	150–160	16	14.5%	
	160–170	43	39.1%	
	170–180	33	30.0%	
	>180	18	16.4%	
BMI (kg/m^2)	<18.5	1	0.9%	
	18.5–25	65	59.1%	
	25–30	30	27.3%	
	>30	14	12.7%	

TABLE 2: Physical activity and Label, and the number of each kind of samples after feature engineering.

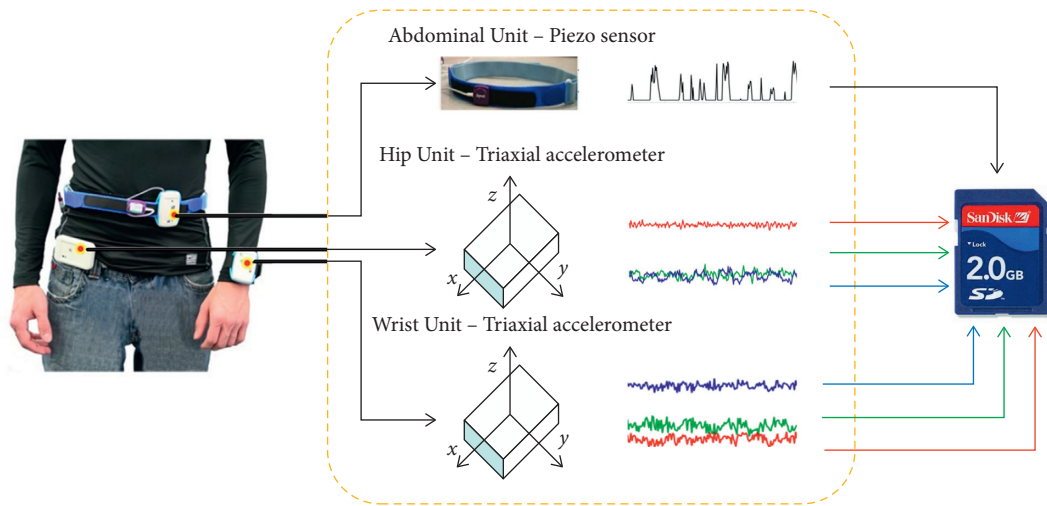| Label and activity		
1-Table working (1635)	2-Housekeeping (807)	3-Moving objects (794)
4-Walking (840)	5-Cycling (1549)	6-Running low speed (1646)
7-Running fast speed (494)	8-Playing tennis (359)	9-Playing basketball (462)

FIGURE 5: Wearing position of sensors and raw data acquisition. The wrist unit and the abdominal unit transmit the sensor data to the hip unit wirelessly via ZigBee [50].
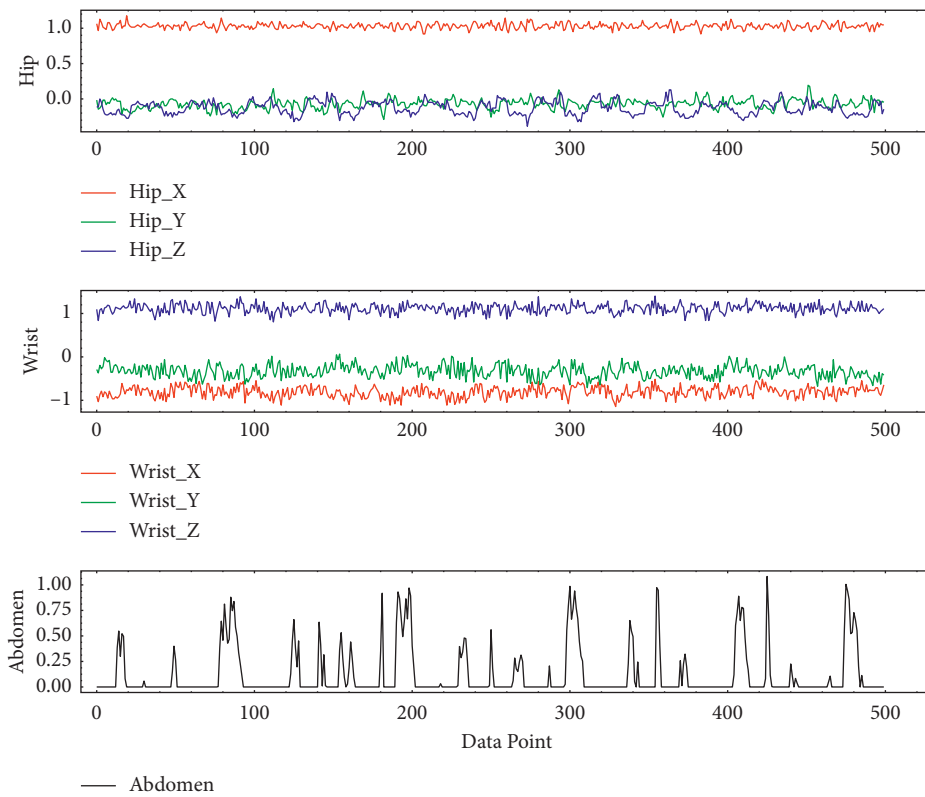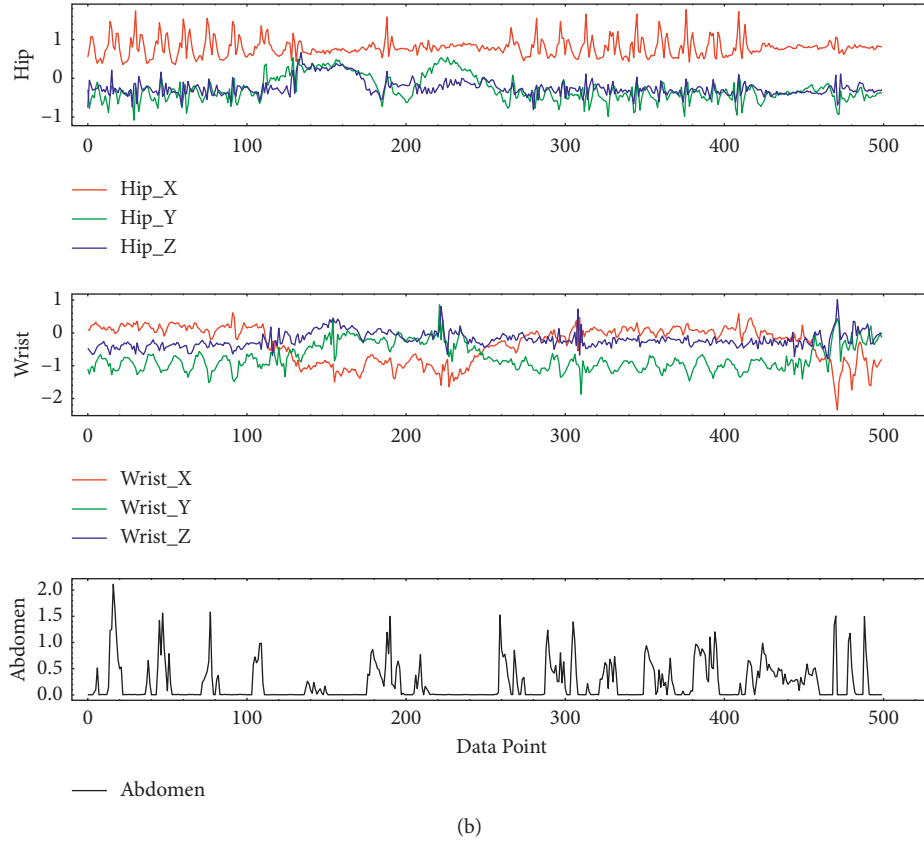

(a)

FIGURE 6: Continued.

(b)

Figure 6: Sensor raw data. (a) Walking. (b) Running (low speed).

Table 3: All the extracted features.

Data sources	Features	Total number
Uniaxial signal and synthetic acceleration signal	Mean, variance, root mean square, kurtosis, skewness, energy [10, 25, 50, 75, and 90] percentile	11 * 9
Between each synthetic acceleration signals and piezo signal	Correlation coefficient	2

neuron is 5000, as (13). First, the ratio factor is set as 1 and different SF used to verify the impact of SF on network size; the result is shown in Figure 7.

$$RF = M \div \text{batch size}. \tag{13}$$

When the growth and pruning of the network reach a balance, the network will converge. Therefore, comparative experiment is carried out on the basis of adjusting the two parameters, recording the accuracy of the PGSOM on the validation set and the number of neurons. The experimental results are shown in Figure 8. The training curve of the model that has high accuracy is shown in Figure 9.

As can be seen from Figure 8, the PGSOM achieves the highest accuracy when SF = 0.8 and RF = 1.5. It can be seen from the training curve in Figure 8 that PGSOM and GSOM models have begun to converge after 45 iterations, and PGSOM can achieve higher accuracy in the validation set. In

order to verify the superiority of PGSOM, we compare PGSOM with SOM and GSOM. Experiments will be conducted on a personal computer with Intel® Core™ i5-10210U CPU, which is quad core, and the frequency is 1.60 GHz. The program used to train the models runs in the Python environment (Python 3.7). Indicators such as the accuracy of model, inference time, and the size of the model are used for comparing the performance of SOM series models. The comparison results are shown in Table 4. The hardware configuration of inference stage is consistent with that of training stage.

In addition, we supplement the comparison results between PGSOM and other unsupervised learning methods such as k-means, GMM, DBSCAN, and hierarchical clustering in Figure 10.

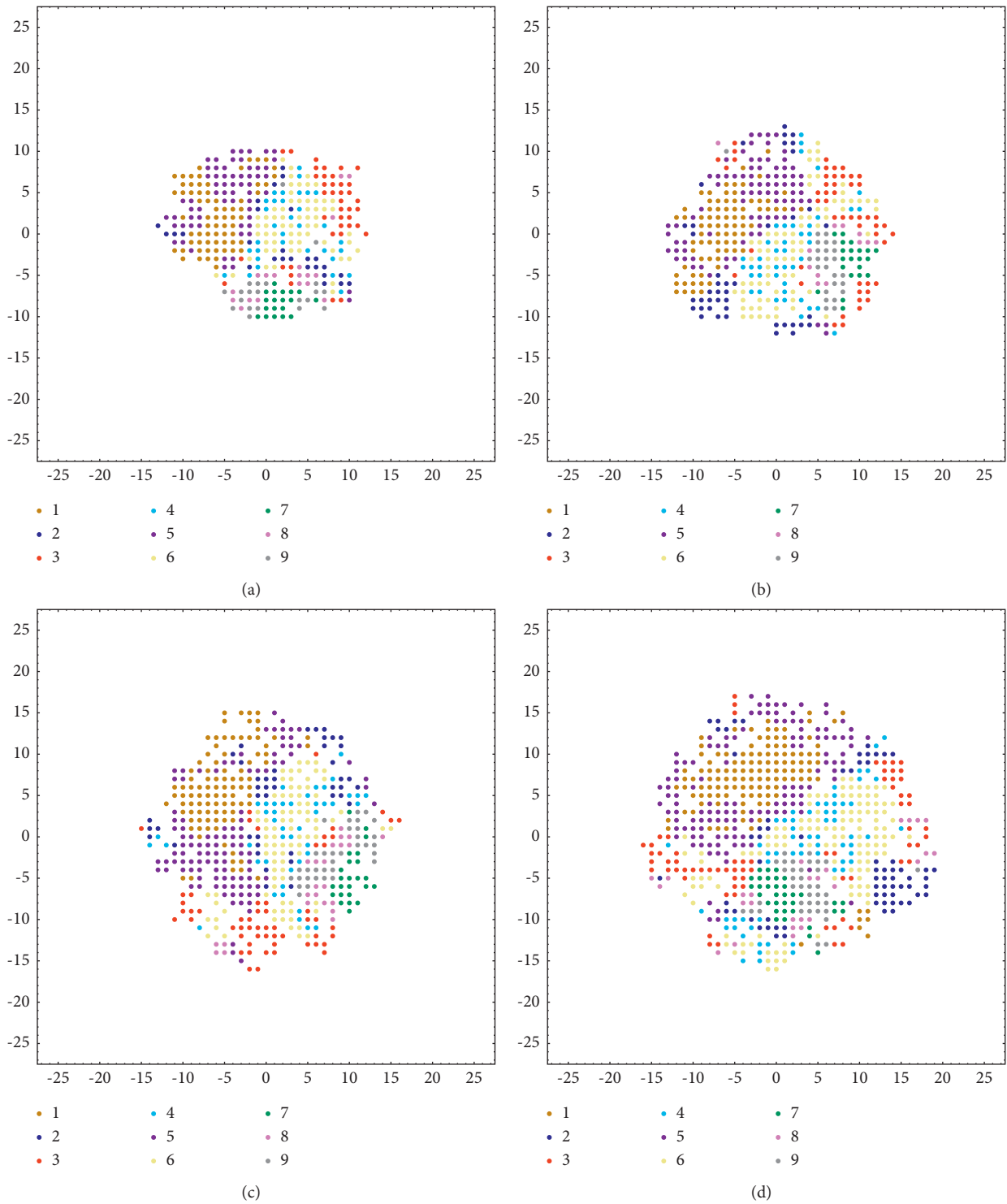Other indicators such as kappa, recall, and F measure are calculated in Table 5.

FIGURE 7: Neuron distribution of PGSOM with different spread factors. (a) SF = 0.1. (b) SF = 0.25. (c) SF = 0.5. (d) SF = 0.75.

5. Discussion

There will be isolated neuronal clusters no matter how to set the hyperparameter of PGSOM from Figure 7, which shows that there are transition regions between neurons representing different patterns, and the neurons in these transition regions are deleted because they cannot represent any pattern, pruning out the fact that these useless neurons will help speed up the training of the PGSOM, and the size of the PGSOM will increase with the increase of SF from (a) to (d) in Figure 7.

Then, different SF and RF are set up to train PGSOM to find the appropriate hyperparameter for the physical activity dataset and the best accuracy appears when SF = 0.8 and
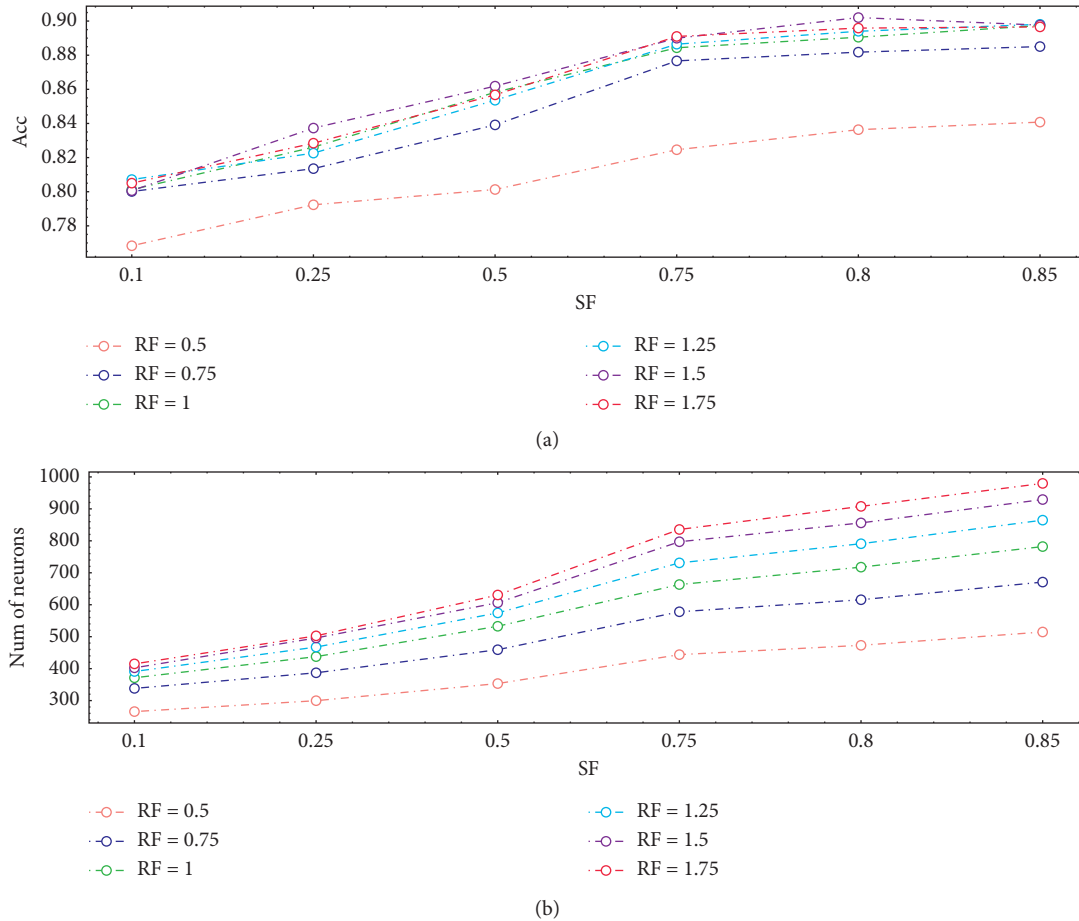
(a)

(b)

FIGURE 8: Results under different SF and RF. (a) Accuracy of model. (b) Number of neurons in the model.

RF = 1.5 from Figure 8. In Figure 9, we draw several training curves of PGSOM and GSOM, which proves that the classification performance of PGSOM is better than GSOM, and each model basically converges when the number of iterations reaches 45. Therefore, in the following experiments, we set the maximum iterations as 50 to ensure the network convergence.

In Table 3, we compare the performance of a series of SOM networks including SOM of 3 structures (20 ∗ 20, 25 ∗ 25, and 30 ∗ 30), GSOM of 3 structures (SF = 0.75, 0.8, and 0.85), and PGSOM of 6 structures (SF, RF = 0.75/1.25, 0.75/1.50, 0.80/1.25, 0.80/1.50, 0.85/1.25, and 0.85/1.50) according to the accuracy of model, inference time, and the size of the model. Generally speaking, networks with more neurons have better identification ability and generalization performance, but it will also increase the cost of training and retain too many redundant neurons to increase the cost of model storage. The top accuracy of SOM is 86.723% when the size of competition layer is 30 ∗ 30, the top accuracy of GSOM is 88.272% when SF is 0.8 and the top accuracy of PGSOM is 90.120% when SF is 0.8 and RF is 1.5. Obviously, the last two models have better performance. The inference time of top PGSOM model is 0.262 ms, which is 50.3% less than the top GSOM model, and the number of neurons used is 856.1, which is 31.7% less than the top GSOM model. It

proves that the introduction of pruning mechanism can effectively delete redundant neurons, improve the structure of the network, and optimize the training speed of the network.

Besides, both GSOM and PGSOM have the ability to grow new neurons when there are samples belonging to new patterns being input. The network will distribute new area to store these new patterns, which means these models have incremental learning ability. In Figure 10, it can be found from the experimental results that k-means algorithm performs worst among all unsupervised learning algorithms because it misidentifies Housekeeping and Playing tennis more frequently than other algorithms. All 6 algorithms have difficulty in distinguishing Walking and Running (low speed). Due to the length of the article, the confusion matrix of each algorithm is not expanded here. But since the activity category number is fixed, k-means, GMM, and hierarchical clustering algorithm cannot generate more clusters to describe these two activities more accurately. As for DBSCAN, the data density is obviously uneven in this dataset, especially those of easy-misidentified activity. Therefore, the performance of DBSCAN is limited. But GSOM and PGSOM can get over these adverse conditions to reach higher accuracy. In addition, because the adaptive adjustment mechanism of the network increases the stability of the
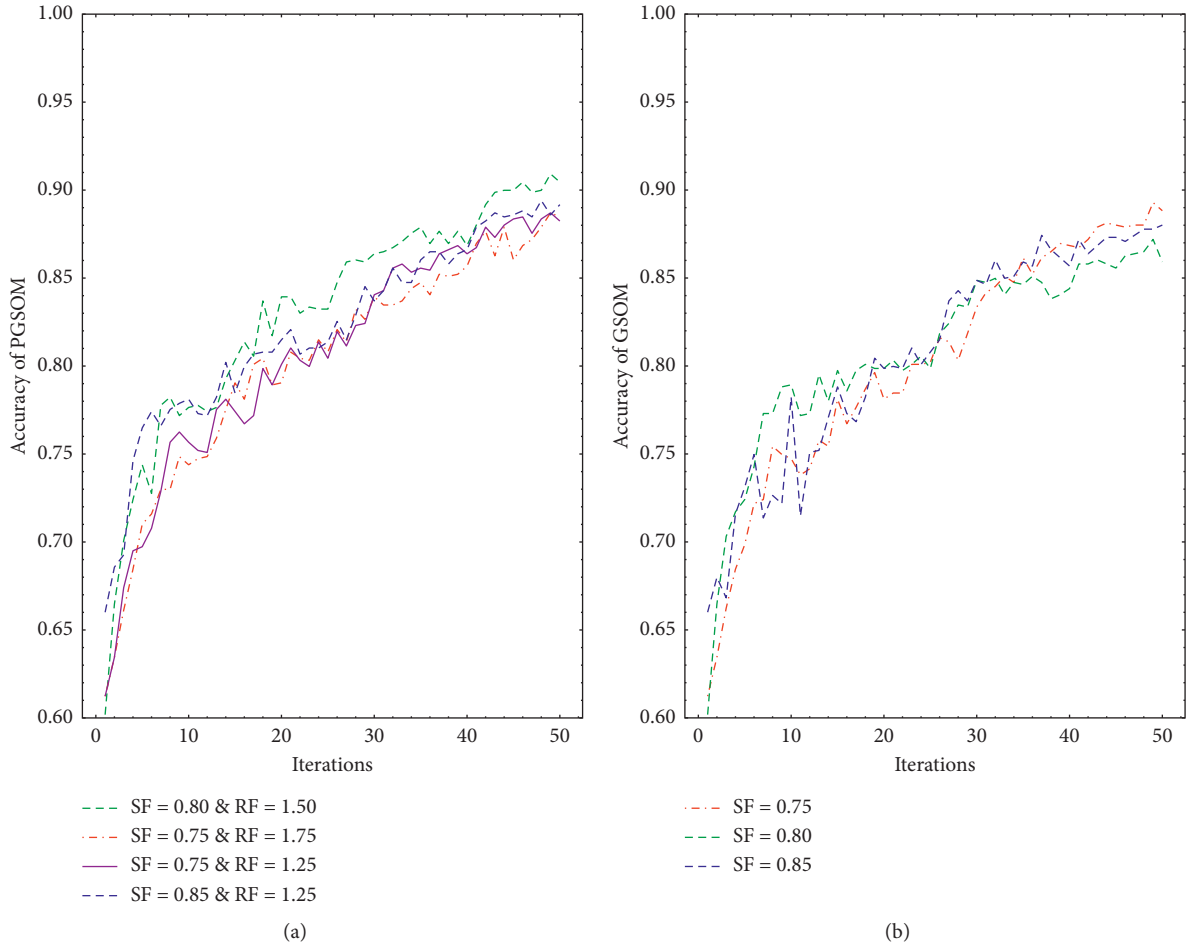
FIGURE 9: Training curve of Some models. (a) Accuracy of PGSOM. (b) Accuracy of GSOM.

TABLE 4: Training performance of different SOM models

Model	Hyperparameter	Accuracy (%)	Inference time (ms)	Number of neurons
	20 ∗ 20	79.269	0.071	400
SOM	25 ∗ 25	84.405	0.110	625
	30 ∗ 30	**87.596**	0.261	900
	SF = 0.75	87.713	0.294	1044
GSOM	SF = 0.80	88.027	0.315	1155
	SF = 0.85	**88.272**	0.527	1254
	SF = 0.75, RF = 1.25	88.644	0.127	731
	SF = 0.75, RF = 1.50	88.994	0.161	797.1
PGSOM (ours)	SF = 0.80, RF = 1.25	89.401	0.265	864
	SF = 0.80, RF = 1.50	**90.120**	0.262	856.1
	SF = 0.85, RF = 1.25	89.809	0.261	864
	SF = 0.85, RF = 1.50	89.762	0.283	929

network, the standard deviation of PGSOM is smaller than other unsupervised learning methods.

We give the confusion matrix of GSOM and PGSOM in Figure 11 to analyze the source of identification error and find that the main error source of both GSOM and PGSOM is the misclassification of Playing tennis as Playing basketball or Moving objects. We hold the view that it is due to the imbalance of the dataset, but the identification accuracy of other activities by PGSOM has reached a high level, which also reflects the robustness of the algorithm to the unbalanced dataset. And the identification effect of PGSOM is almost better than GSOM. We believe that the pruning mechanism removes redundant neurons and reduces the misidentification rate of the model.
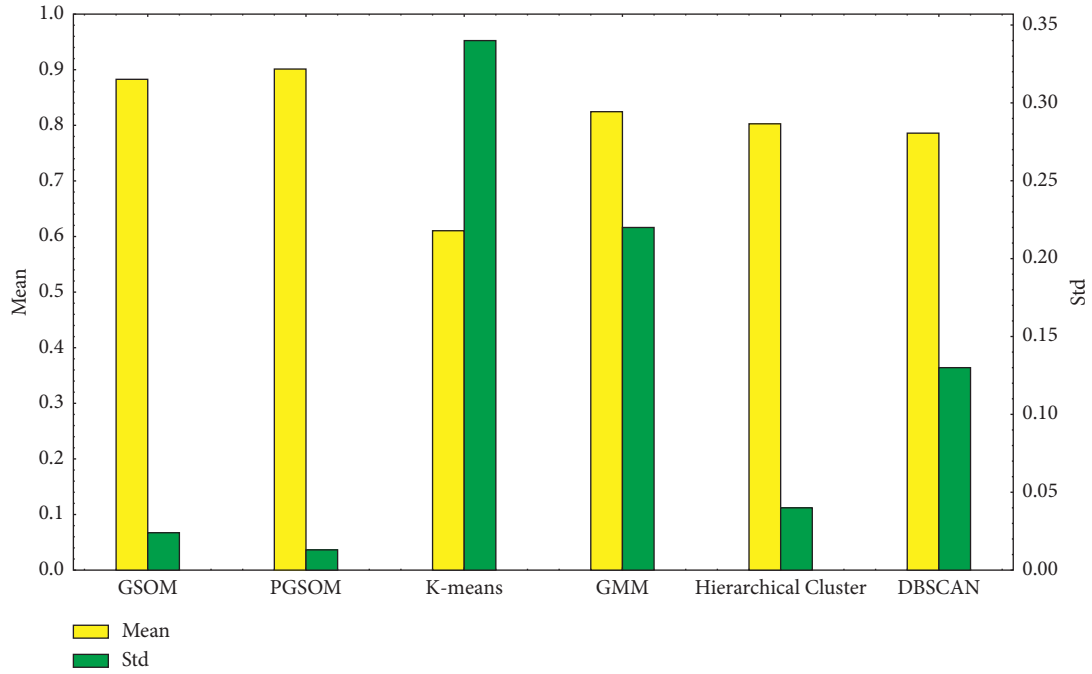
FIGURE 10: Accuracy of unsupervised learning algorithms in proposed human physical activity dataset.

TABLE 5: Performance comparison of unsupervised learning algorithms.

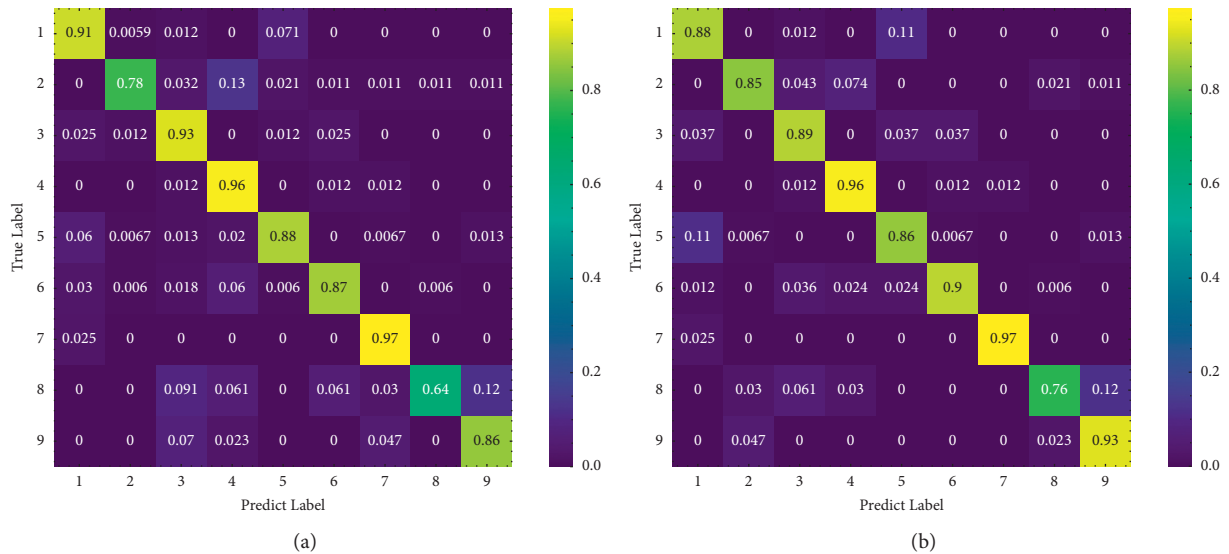| Algorithm	Kappa	F measure	Inference time (ms)
GSOM	0.882	0.896	0.527
PGSOM (ours)	0.897	0.902	0.262
k-means	0.611	0.613	4.101
GMM	0.852	0.873	9.033
Hierarchical cluster	0.802	0.786	4.371
DBSCAN	0.778	0.808	4.205

FIGURE 11: Confusion matrix of model. (a) Confusion matrix of top GSOM model. (b) Confusion matrix of top PGSOM model.

The results of the experiments indicate that the proposed model works well in the human physical activity identification task, and the identification effect of action 2 and action 8 is significantly improved because of the pruning mechanism deleting a large number of ineffective neurons with side effects on identification. In addition, due to the incremental learning ability of PGSOM, this model can adapt itself to the particular user when practically applied, which means that individual differences will not be a serious problem when this model is utilized as a human physical activity classifier if each user uses its own physical activity data to further train this model.

6. Conclusions

PGSOM is proposed to address the scale of GSOM neurons. The adaptive growth of GSOM can adjust the structure of model to the distribution of the dataset with the limitations of little prior knowledge, which makes the application scenarios of the algorithm more abundant and makes it possible for the algorithm to migrate between different individuals. But it also leads to a large number of neurons, which increase the storage and reduce the inference speed and the identification accuracy of the model. PGSOM inherits the advantages of GSOM and adds pruning mechanism, which can delete the inactive neurons with useless information as well as learn the new knowledge. The PGSOM algorithm has been utilized to identify 9 kinds of physical activities and the accuracy on the provided dataset can reach 90.120%. Compared with other unsupervised learning algorithms, the proposed algorithm has competitive effect. Besides, it can obtain higher identification accuracy and reduce the inference time as well as reduce the storage cost of the model.

Data Availability

The human activity data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] S. Tow, J. Gober, and M. R. Nelson, "Adaptive sports, arts, recreation, and community engagement," *Physical Medicine and Rehabilitation Clinics of North America*, vol. 31, no. 1, pp. 143–158, 2019.

[2] K. J. Isern, S. Mecho, R. Pruna et al., "Sports-related lower limb muscle injuries: pattern recognition approach and MRI review," *Insights into Imaging*, vol. 11, no. 108, 2020.

[3] J. Parkka, M. Ermes, P. Korpipaa, J. Mantyjarvi, J. Peltola, and I. Korhonen, "Activity classification using realistic data from wearable sensors," *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 1, pp. 119–128, 2016.

[4] A. Subasi, K. Khateeb, T. Brahimi, and A. Sarirete, "Human activity recognition using machine learning methods in a smart healthcare environment," *Innovation in Health Informatics*, Next Generation Technology Driven Personalized Medicine and Smart Healthcare, pp. 123–144, 2020.

[5] M. Yoshioka, M. Ayabe, T. Yahiro et al., "Long-period accelerometer monitoring shows the role of physical activity in overweight and obesity," *International Journal of Obesity (2005)*, vol. 29, pp. 502–508, 2005.

[6] S. Zhou, Q. Shan, F. Fei, W. J. Li, and J. Y. J. Liou, "Gesture recognition for interactive controllers using MEMS motion sensors," in *Proceedings of the 4th IEEE International Conference on Nano/Micro Engineered and Molecular Systems, IEEE-NEMS*, January 2009.

[7] M. Shoaib, S. Bosch, O. Incel, H. Scholten, and P Havinga, "Fusion of smartphone motion sensors for physical activity recognition," *Sensors*, vol. 14, no. 6, pp. 10146–10176, 2014.

[8] K. Wang, J. He, and L. Zhang, "Attention-based convolutional neural network for weakly labeled human activities' recognition with wearable sensors," *IEEE Sensors Journal*, vol. 19, no. 17, pp. 7598–7604, 2019.

[9] T. Mahmud, S. S. Akash, S. A. Fattah, W.-P. Zhu, and M. O. Ahmad, "Human activity recognition from multi-modal wearable sensor data using deep multi-stage LSTM architecture based on temporal feature aggregation," in *Proceedings of the 2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 249–252, IEEE, Springfield, MA, USA, August 2020.

[10] E. Valarezo, P. Rivera, J. M. Park et al., "Human activity recognition using a single wrist IMU sensor via deep learning convolutional and recurrent neural nets," *UNIKOM Journal of ICT, Design, Engineering and Technological Science1*, vol. 1, pp. 1–5, 2017.

[11] M. Yu, W. Zhang, Q. Zeng, and C. Wang, "Human-object contour for action recognition with attentional multi-modal fusion network," in *Proceedings of the 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*https://ieeexplore.ieee.org/author/38240583600, Okinawa, Japan, February 2019.

[12] M. Yu, "Human action identification by a quality-guided fusion of multi-model feature - sciencedirect," *Future Generation Computer Systems*, vol. 116, pp. 13–21, 2021.

[13] A. Akula, A. K. Shah, and R. Ghosh, "Deep learning approach for human action recognition in infrared images," *Cognitive Systems Research*, vol. 50, pp. 146–154, 2018.

[14] S. Mohammadi, S. G. Majelan, and S. B. Shokouhi, "Ensembles of deep neural networks for action recognition in still images," in *Proceedings of the 2019 9th International Conference On Computer And Knowledge Engineering (ICCKE)*, October, 2019.

[15] X. Yu, Z. Zhang, L. Wu et al., "Deep ensemble learning for human action recognition in still images," *Complexity*, vol. 2020, no. 1, 23 pages, Article ID 9428612, 2020.

[16] B. Wu, Y. Liu, B. Lang, and L. Huang, "DGCNN: disordered graph convolutional neural network based on the Gaussian mixture model," *Neurocomputing*, vol. 321, pp. 346–356, 2017.

[17] S. Fu, W. Liu, D. Tao, Y. Zhou, and L. Nie, "HesGCN: hessian graph convolutional networks for semi-supervised classification - sciencedirect," *Information Sciences*, vol. 514, pp. 484–498, 2020.

[18] H. Shi, Y. Zhang, Z. Zhang et al., "Hypergraph-induced convolutional networks for visual classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 3, pp. 1–10, 2018.

[19] S. Fu, W. Liu, Y. Zhou, and L. Nie, "HpLapGCN: hypergraph P-Laplacian graph convolutional networks," *Neurocomputing*, vol. 362, pp. 166–174, 2019.

[20] N. Khan, U. Chaudhuri, B. Banerjee, and S. Chaudhuri, "Graph convolutional network for multi-label VHR remote sensing scene recognition," *Neurocomputing*, vol. 357, pp. 36–46, 2019.

[21] F. Sichao, L. Weifeng, L. Shuying, and L. Weifeng, "Two-order graph convolutional networks for semi-supervised classification," *IET Image Processing*, vol. 13, no. 14, pp. 2763–2771, 2020.

[22] Y. Kwon, K. Kang, and C. Bae, "Unsupervised learning for human activity recognition using smartphone sensors," *Expert Systems with Applications*, vol. 41, no. 14, pp. 6067–6074, 2014.

[23] T. Hachaj and M. R. Ogiela, "Full body movements recognition - unsupervised learning approach with heuristic R-GDL method," *Digital Signal Processing*, vol. 46, pp. 239–252, 2015.

[24] W.-H. Ong, P. Leon, and T. Koseki, "Investigation of feature extraction for unsupervised learning in human activity detection," *Bulletin of Networking, Computing, Systems, and Software*, vol. 1, 2013.

[25] Y. Yang Song, L. Goncalves, and P. Perona, "Unsupervised learning of human motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 814–827, 2003.

[26] D. Trabelsi, S. Mohammed, Y. Amirat, and L. Oukhellou, "Activity recognition using body mounted sensors: an unsupervised learning based approach," in *Proceedings of the 2012 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, IEEE, Brisbane, Australia, June 2012.

[27] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[28] B. Fritzke, "Growing cell structures-a self-organizing network for unsupervised and supervised learning," *Neural Networks*, vol. 7, no. 9, pp. 1441–1460, 1994.

[29] P. Koikkalainen, "Tree structured self-organizing maps," Elsevier Science BV, Amsterdam, Netherlands, 1999.

[30] R. E. López and A. D. Ramos, "Grid topologies for the self-organizing map," *Neural Networks*, vol. 56, pp. 35–48, 2014.

[31] E. Berglund and J. Sitte, "The parameterless self-organizing map algorithm," *IEEE Transactions on Neural Networks*, vol. 17, no. 2, pp. 305–316, 2006.

[32] E. Berglund, "Improved PLSOM algorithm," *Applied Intelligence*, vol. 32, no. 1, pp. 122–130, 2010.

[33] A. D. Ramos, E. L. Rubio, and E. J. Palomo, "The forbidden region self-organizing map neural network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 1, pp. 201–211, 2019.

[34] P. D'Urso, L. De Giovanni, and R. Massari, "Smoothed self-organizing map for robust clustering," *Information Sciences*, vol. 512, pp. 381–401, 2020.

[35] D. Polani, "On the optimization of self-organizing maps by genetic algorithms," Elsevier Science BV, Amsterdam, Netherlands, 1999.

[36] Y. W. Jing and L. F. Li, "SOM and PSO based alerts clustering in intrusion detection system," *Applied Mechanics and Materials*, Trans Tech Publications Ltd, vol. 401-403, , pp. 1453–1457, 2013.

[37] S. M. Guru, A. Hsu, S. Halgamuge, and S. Fernando, "An extended growing self-organizing map for selection of clusters in sensor networks," *International Journal of Distributed Sensor Networks*, vol. 1, no. 2, pp. 227–243, 2005.

[38] M. Jayaratne, D. Alahakoon, D. D. Silva, and X. Yu, "Apache spark based distributed self-organizing map algorithm for sensor data analysis," in *Proceedings of the Conference of the IEEE Industrial Electronics Society*, November 2017.

[39] T. Ichimura, S. Kamada, and K. Kato, "Analysis of tourist subjective data in smartphone based participatory sensing system by interactive growing hierarchical SOM," *Intelligent Decision Technologies*, Springer, Berlin, Heidelberg, pp. 225–235, 2012.

[40] T. Ichimura and T. Yamaguchi, "A proposal of interactive growing hierarchical SOM," in *Proceedings of the 2011 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 3149–3154, IEEE, Anchorage, AK, USA, October 2011.

[41] R. Amarasiri, D. Alahakoon, and K. Smith, "Applications of the growing self-organizing map in high dimensional data," in *Proceedings of the International Information Technology Conference (IITC)*, Burlingame, CA, USA, June 2004.

[42] T. Kuremoto, T. Otani, M. Obayashi, K. Kobayashi, and S. Mabu, "A hand shape instruction recognition and learning system using growing SOM with asymmetric neighborhood function," *Neurocomputing*, vol. 188, pp. 31–41, 2016.

[43] T. Esaki and T. Hashiyama, "Extracting human players' shogi game strategies from game records using growing SOM," in *Proceedings of the 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pp. 2176–2181, IEEE, Hong Kong, China, June 2008.

[44] M. B. Gorzałczany and F. Rudziński, "Generalized self-organizing maps for automatic determination of the number of clusters and their multiprototypes in cluster analysis," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 7, pp. 2833–2845, 2017.

[45] D. Castro, L. Nunes, and F. J. Von Zuben, "An improving pruning technique with restart for the Kohonen self-organizing feature map,"in Proceedings of the IJCNN'99. InternationalJoint Conference on Neural Networks. Proceedings, vol. 3, IEEE, Washington, DC, USA, July 1999.

[46] L. N. De Castro, F. J. Von Zuben, J. Fernando, and Von Zuben, "A pruning self-organizing algorithm to select centers of radial basis function neural networks," *Artificial Neural Nets and Genetic Algorithms*, Springer, Berlin, Germany, pp. 114–117, 2001.

[47] D. Beaton, I. Valova, and D. MacLean, "Growing mechanisms and cluster identification with TurSOM," in *Proceedings of the 2009 International Joint Conference on Neural Networks*, pp. 280–287, IEEE, Atlanta, GA, USA, June 2009.

[48] D. Beaton, I. Valova, and D. Maclean, "TurSOM: A turing inspired self-organizing map," in *Proceedings of the International Joint Conference On Neural Networks*, June 2009.

[49] S. Furao and O. Hasegawa, "An incremental network for on-line unsupervised classification and topology learning," *Neural Networks*, vol. 19, no. 1, pp. 90–106, 2006.

[50] L. Mo, L. Zeng, S. Liu, and R. X. Gao, "Multi-sensor activity monitoring: combination of models with class-specific voting," *Information*, vol. 10, no. 6, 197 pages, 2019.

[51] G. Okeyo, L. Chen, H. Wang, and R. Sterritt, "Dynamic sensor data segmentation for real-time knowledge-driven activity recognition," *Pervasive and Mobile Computing*, vol. 10, pp. 155–172, 2014.