Journal of Systems Engineering and Electronics
Vol. 33, No. 5, October 2022, pp.1108 – 1122

A situation awareness assessment method based on fuzzy cognitive maps

CHEN Jun[1,2,*], GAO Xudong[1], RONG Jia[3], and GAO Xiaoguang[1]

1. School of Electronics and Information, Northwestern Polytechnical University, Xi’an 710072, China; 2. Chongqing Institute for Brain and Intelligence, Guangyang Bay Laboratory, Chongqing 400064, China; 3. Department of Data Science and AI, Monash University, Clayton VIC3800, Australia

Abstract: The status of an operator’s situation awareness is one of the critical factors that influence the quality of the missions. Thus the measurement method of the situation awareness status is an important topic to research. So far, there are lots of methods designed for the measurement of situation awareness status, but there is no model that can measure it accurately in real-time, so this work is conducted to deal with such a gap. Firstly, collect the relevant physiological data of operators while they are performing a specific mission, simultaneously, measure their status of situation awareness by using the situation awareness global assessment technique (SAGAT), which is known for accuracy but cannot be used in real-time. And then, after the preprocessing of the raw data, use the physiological data as features, the SAGAT’s results as a label to train a fuzzy cognitive map (FCM), which is an explainable and powerful intelligent model. Also, a hybrid learning algorithm of particle swarm optimization (PSO) and gradient descent is proposed for the FCM training. The final results show that the learned FCM can assess the status of situation awareness accurately in real-time, and the proposed hybrid learning algorithm has better efficiency and accuracy.

Keywords: situation awareness (SA), fuzzy cognitive map (FCM), particle swarm optimization (PSO), gradient descent.

DOI: 10.23919/JSEE.2022.000108

1. Introduction

As the age of industrialization and information technology accelerates, more and more complex machines and operating systems are being invented and put into practical use. Operators of systems will be confronted with more complex and rapidly changing systems. In the decision-making process for dynamic systems, it is becoming apparent that situation awareness (SA) becomes a key component in the decision-making process for dynamic systems [1]. As the name implies, SA studies the understanding and knowledge of the situation under current conditions. The concept is most widely used in aviation, and it can even be argued that the concept of SA was discovered and refined from research in the aviation sector [2]. Pilots are often faced with a large amount of information to perceive and process during a flight or mission, requiring accurate perception and understanding of the information. SA theory is also applied in other key areas of cognitive loops, such as air traffic control [3], large systems operations in large manufacturing plants [4], and medical systems [5]. Thus, we will not distinguish between pilots and operators later in this paper.

It is obvious that the status of an operator’s SA is very important during the mission, and thus there exist lots of measurement methods for SA, such as situation present assessment method (SPAM) [6], SA global assessment technique (SAGAT) [7], and physiological process indices [8]. However, till now, there is no method that can measure the status of SA in real-time with high accuracy. The SAGAT is accurate enough but it cannot be applied in real-time, the method of physiological process indices can measure SA in real-time but lose the accuracy. Thus, the paper’s aim is to deal with the gap between the accuracy and real-time of the SA measurement.

As mentioned before, the SAGAT method has a high accuracy, and the physiological process indices method can be applied in real-time. The basics of the physiological indices method are that some physiological indices like eye movement, blood oxygen saturation, variance of the heart rate, etc, can reflect a human’s psychological status like worries, happiness, fatigue, and so on [9]. Thus, we propose a hypothesis that the physiological indices can also predict the status of SA. The methods in

Manuscript received October 19, 2021.

*Corresponding author.

This work was supported by the National Natural Science Foundation of China (61305133) and the Aeronautical Science Foundation of China grant number 2020Z023053002.

the literature failed in accuracy because they are not calibrated by the accurate methods.

Thus, this paper tries to use the result of the SAGAT, which is proved accurate already, to calibrate the SA assessment model which uses psychological indices. Firstly, an experiment is conducted to get the data, which includes the SAGAT results and the relevant physiological indices at the same time. Secondly, we train a fuzzy cognitive map (FCM) [10] using the supervised learning method, the physiological indices are features and the SAGAT results are labels. After the training process, the FCM is the final model of SA assessment.

The reason for choosing the FCM as the model is its interpretability and ease to use [11]. FCM is a new kind of intelligent model, it looks like a neural network, but it supports the circle causality [12]. In most cases where the status of the operator's SA needs to be assessed, such as a pilot is controlling an aircraft landing, a surgeon is performing a very important operation, we need to know why the SA's status is good or bad, so the popular models like deep neural network can not be used since it is a 'black box' and cannot be explained. The FCM's structure and reasoning process are very similar to the neural network [13], so it has enough knowledge representation and process ability to assess the SA status, it is more powerful than many of the classic pattern recognition methods like random forest and decision tree, etc.

The only obstacle is that the FCM does not have an efficient learning algorithm [14]. This paper tackles such a problem by using a hybrid method of particle swarm optimization (PSO) and gradient descent. Using such a hybrid learning algorithm, both efficiency and learning accuracy are improved.

The innovations of this paper are as follows:

(i) Build an FCM-based model to assess the SA status of the operators, such a model is explainable and can measure the SA status accurately in real-time.

(ii) Using the hybrid algorithm of PSO and gradient descent to train an FCM, exploit the PSO's global optimization capability and the efficiency of the gradient descent simultaneously.

The rest of this paper is organized as follows: Section 2 clarifies the definition of SA and review the measurement methods in literature; Section 3 shows the data collection method and the basics of FCM, and describe the learning algorithms; Section 4 shows the data preprocessing; Section 5 shows the learning results and does the analysis; Section 6 concludes the whole paper's work.

2. SA

This section shows the relevant concepts of SA and an overview of the measurement methods of SA.

2.1 Concepts about SA

The concept of SA was introduced by the US military in the 1980s and went through a series of debates on the definition of SA and whether it could be defined in the 1980s to 1990s. Now the academic community basically agrees with the three-level model definition of Endsley, the chief scientist of the US Air Force: perception of elements in a given space-time, understanding of the meaning of elements, and prediction of the state of elements over a small period of time in the future [1].

This definition is based on the dynamic systems decision model, which shows that SA is the preparation of a person before making a decision and is also the interface between the person and the external environment. It is important to note that SA is a state of knowledge, rather than a process of acquiring this knowledge. The process of acquiring SA is generally called SA, or achieving, maintaining, and acquiring SA. SA also does not include all of a person's knowledge, because the process of SA is considered to take place in the person's working memory, i.e., short-term memory, but SA includes the relevant knowledge that is relevant to the task and is recalled by the current operator. That is, SA does not include knowledge stored in the operator's long-term memory that is not recalled in the course of performing the task. SA should also be distinguished from operator decision making and task performance, as experienced operators can make poor decisions and thus perform poorly when their SA is poor, while inexperienced novices can perform poorly with good SA due to their inexperience. Of course, factors that have an impact on SA, such as working memory, and stress, should be distinguished from SA.

From Fig. 1, it can be seen that SA is divided into three levels in progressive order: perception, comprehension, and projection. The higher levels depend on the formation of the lower levels.

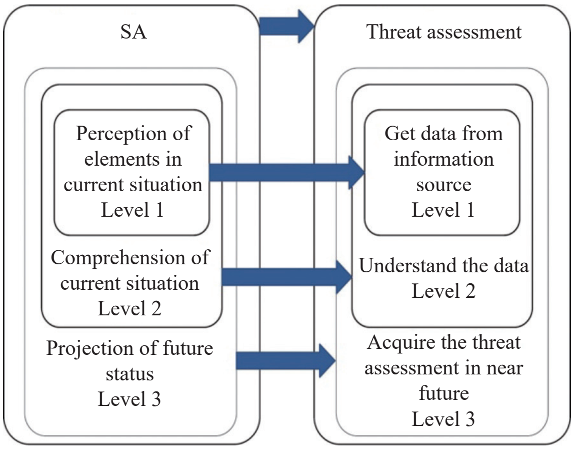

SA
Threat assessment
Perception of elements in current situation Level 1
Comprehension of current situation Level 2
Projection of future status Level 3
Get data from information source Level 1
Understand the data Level 2
Acquire the threat assessment in near future Level 3

Fig. 1 Three levels structure of SA [15]

(i) Level 1: Perception

The first step in gaining SA is to perceive the state of elements in the current environment, the attributes, and the current key elements. For example, a pilot should perceive aircraft-related parameters such as altitude, speed, heading and angle, environment-related parameters, the presence of mountains, and speed.

(ii) Level 2: Comprehension

The second level of SA is the understanding of the current situation. The disordered information of the first level is sorted out and integrated in a relevant way, processing the interrelationships between the elements and thus obtaining specific corresponding information, for example, the type of aircraft based on its maneuvering characteristics. From the viewpoint of the target, the unimportant elements are eliminated. With the same level of information at the first level, an experienced operator is likely to get better information at the second level than a novice.

(iii) Level 3: Projection

With the second level of information, the operator is ideally able to make certain predictions about the future state of the system based on his understanding of it. For example, the pilot predicts the aircraft will hit the mountains if the aircraft maintains the heading now, in order to avoid that, he/she must take necessary actions.

2.2 An overview of assessment methods for SA

There are a number of evaluation methods for SA [16]. These methods fall broadly into the following categories:

(i) SA requirement analysis

By the goal directed task analysis (GDTA) method, the needs of the SA are analyzed and the individual demand states are then measured separately [17].

(ii) Freeze probe techniques

At a random point in time, while the operator is performing a task, the operator is completely removed from the task situation and the operator's SA state is then immediately and rapidly measured by using the appropriate scale. The SAGAT is a very famous method that can be seen as a freeze probe technique [18].

(iii) Real-time probe techniques

The difference between the real-time probe technique and the frozen probe technique is that the operator is not removed from the task situation and the surveyor evaluates the information by directly asking the SA about it, based on parameters such as the correctness of the operator's answers and the operator's reaction time, but the evaluation process can interfere with the operator's SA state, resulting in inaccurate measurement results. The SPAM which was proposed by Durso et al. and used to assess the SA of air traffic controller (ATC) personnel, is representative of this approach [19].

(iv) Self-rating techniques

After the operator has performed the task, the operator is evaluated by using the relevant SA scale. Clearly, this is a post-test measurement method that avoids measuring the effects of the test process on the state of SA (in fact, as confirmed by the experiments of Endsley et al. The measurement during the SAGAT did not also have any influence on the operator in a statistically significant way). SA rating techniques, proposed by Taylor in 1990, represent this type of approach, using a 7-point scale in which operators self-assess after completing a task on 10 dimensions [20]. The advantage of this type of method is that the operator performs the task without interference, but the pilot's ability to assess himself objectively and accurately greatly affects the reliability and validity of the measurement.

(v) Observer-rating techniques

In the observer scoring technique, the operator, as the observed, is only responsible for performing the task and does not interact with the assessor and is thus not disturbed. The assessor consists of a subject matter expert (SME), who assesses the SA of the observed person by looking at the operator's performance, the behavior associated with the SA. The problem is whether the SME can accurately determine the SA of the observed person because the SME is not the observed person and he/she has no way of knowing what is going on in the mind of the observed person [21]. This makes the reliability and validity of the method extremely challenging.

(vi) Performance measures

Gugerty in 1997 measured the SA of drivers in a simulated driving environment using three performance indicators: hazard detection, blocking vehicle detection, and collision avoidance [22]. An expert can perform well when his SA is poor, while a novice can make poor decisions when his SA is good. That is, performance, not SA, is measured.

(vii) Process indices

The process of constructing one's own SA is accompanied by process indicators, such as observing relevant information, scanning for relevant information, and reasoning about the necessary information. Typical methods include verbal protocol analysis, where subjects report their SA in the verbal form in real-time as they perform the task. The use of eye-tracker and physiological data to determine the operator's SA is also part of this approach [23].

3. Method and model

The operator's level of SA is assessed in real-time using the operator's eye movement and physiological data. The data used is the operator's eye movements and physiological data while operating the aircraft in a cruise mission on the flight simulation experiment platform (Fig. 2), and the level of SA is assessed by a SAGAT, where the operator is completely isolated from the mission environment and assessed by a questionnaire as a label for the learning data.

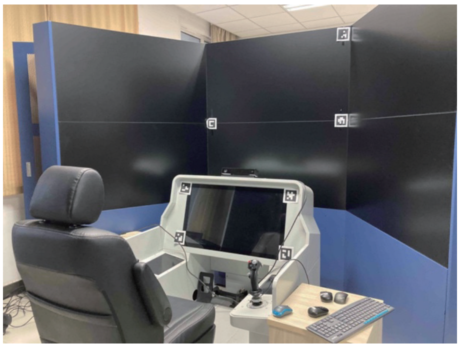


Fig. 2 Simulation experiment platform

After acquiring the relevant data, the appropriate preprocessing is carried out to extract the appropriate indicators, and feature selection is carried out appropriately. Subsequently, the filtered features and the test results of SAGAT are used as training data and the corresponding models and learning methods are selected for training.

3.1 Data collection

The data is obtained as follows:

(i) Flight simulations are performed by subjects on a flight simulation experiment platform, during which the subjects' eye movement data and physiological data are measured in real-time by oculomotor and physiological instruments.

(ii) The experimenter calls off the subject's task after a random time (longer than 6 s), the subject is removed from the simulation environment, and both the oculomotor and physiograph measurements are ended, saving the data from this measurement.

(iii) Immediately after the subject is removed from the simulation environment, the subject's level of SA is tested on a task-specific SAGAT and the results of this measurement are saved.

3.2 FCM

The FCM was proposed by Kosko in 1986 as a soft computing system. It is a combination of fuzzy logic and neural networks. It consists of a number of variable concepts and causal relations that illustrate the causal relationships between these concepts. Nodes represent concepts and directed edges with weights indicate the connections between these nodes. A simple FCM model is shown in Fig. 3.

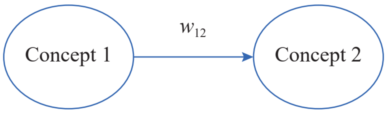


Fig. 3 A simple FCM

Each concept node has its own value, which is always a fuzzy value that represents how well the state matches the concept. Values in the interval [1, 1] or [0, 1] are usually chosen.

The weight is also a fuzzy number such as w_{12} shown in Fig. 3, and we usually choose a real number between the interval $[-1,1]$ as the weight value. A negative weight means that the presynaptic and postsynaptic nodes change in opposite directions during stimulation; a positive weight means that they change in the same direction. And the absolute value of the weights represents the magnitude of the effect of the presence of the two nodes.

Not only does the FCM have a powerful knowledge representations capability, but it can also use the information it collects to deduct the value of a concept. For example, in each iteration, each concept can be given a new value from the following equation:

$$\boldsymbol{C}_{t+1} = S\,(\boldsymbol{C}_t \cdot \boldsymbol{W}) \tag{1}$$

where $\boldsymbol{W}$ is the weight matrix and $\boldsymbol{C}$ is the node vector, $S(\cdot)$ is the activation function to compress the node value into [0, 1] or [1, 1], when the node value is in [0, 1], the activation function is

$$S\,(x) = \frac{1}{1 + \mathrm{e}^{-\lambda x}} \tag{2}$$

when the node value falls in [1, 1], $S(\cdot)$ is the hyperbolic tangent.

The positive parameter λ is used to control the steepness of the curve. The lager λ is, the steeper the curve is.

3.3 The learning algorithm of FCM

To the state of the art, learning algorithms for FCMs are mainly group-based heuristics and hybrid learning methods used to combine expert knowledge with knowledge from data. Heuristic learning methods have become the dominant algorithms for FCMs to extract expert knowledge from data [24]. However, heuristic learning algorithms, such as genetic algorithms and particle swarm algorithms, can give weights that meet the requirements of the learning task but have long learning times and poor learning accuracy. Therefore, many scholars have considered using non-heuristic learning algorithms to learn FCMs. Some scholars have already tried to use gradient descent to learn FCMs [11], but their proposed method requires data after each iteration as labels, while usually, the data of complex systems, in reality, are only the final output results of the system, for the intermediate process data, it is almost difficult to obtain, so such a gradient descent method is difficult to get realistic applications.

By using a numerical approach to calculate the gradient, the problem of the gradient being difficult to solve analytically due to the extreme complexity of the objective function is bypassed, thus eliminating the need to use intermediate data from the operation of the system as labels to learn FCMs.

Due to the complexity of the objective function, it is very easy for a single gradient descent method to fall into a local optimum. In fact, through simulation, the gradient descent method falls into an unacceptable local optimum almost every time, resulting in underfitting and poor learning. Therefore, a combination of the PSO algorithm and the gradient descent method is eventually used to avoid falling into a local optimum prematurely: the particle swarm algorithm is used to obtain a solution that has already reached a certain level of accuracy, and this solution is then used as the initial value for the gradient descent method to learn. That is coarse tuning using PSO and fine-tuning using gradient descent. The comprehensive performance of such a hybrid learning algorithm is the best by simulation.

3.3.1 FCM learning algorithm based on gradient descent

Applying gradient descent to FCM learning is to optimize the objective function using gradient descent, with the weights updated by the following formula:

$$w_{ij}(t+1) = w_{ji}(t) - \eta \Delta w_{ij} \tag{3}$$

where

$$\Delta w_{ij} = \frac{\partial f(\boldsymbol{W})}{\partial w_{ij}}. \tag{4}$$

Δw_{ij} is the gradient, η is the learning step size, the size of its value affects the learning rate and accuracy, the value of η can be dynamically adjusted according to the number of learning steps, and the smallest possible step size is used when the number of steps is large enough or close to the optimal solution. $f(\boldsymbol{W})$ is the objective function, and the following function is used as the objective function to learn the FCM.

$$f(\boldsymbol{W}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} (y_i - \widehat{y_i})^2 + \lambda L_2(\boldsymbol{W}) \tag{5}$$

where $\dfrac{1}{n_{\text{samples}}} \displaystyle\sum_{i=0}^{n_{\text{sample}}-1} (y_i - \widehat{y_i})^2$ is the mean square error and $L_2(\boldsymbol{W})$ is the l_2 norm of the weight matrix as regularization term, making the learned weight matrix as sparse as possible. Also, since in the FCM, when $|w_{ij}| < 0.05$, it is considered that the two nodes see no relationship, so in the process of learning, if $|w_{ij}| < 0.05$, then w_{ij} is set to 0. The algorithm to calculate the objective function is as function Aim$(\cdot)$.

```
Function Aim(X, Features, Y)
Data: The parameter vector X to be learned, training data feature matrix Features, each row represents all features of a sample feature the training data feature vector Y, the feature of each sample is $y_i$, the number of samples is $n_{\text{samples}}$, the number of features is $l$, the FCM($C_{\text{in}}$, $W$)
Result: Objective function value
$W$ = reshape($X$, [$l$+1, $l$ +1]); /* Transform the parameters to be learned into the form of a matrix of the size [$l$+1, $l$ + 1] required for FCM */
for $i = 0 : n_{\text{samples}}$ do
    cin = Features[$i$,:];
    out_FCM = FCM{$C_{\text{in}}$, $W$);
    $\hat{Y}[i]$ = out_FCM[−1] // The last node is the output node
end
mse = $\frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} (y_i - \hat{y}_i)^2$;
$l2 = \lambda L_2(\boldsymbol{W})$;
return mse +$l2$ ;
```

Due to the complexity of FCMs, solving Δw_{ij} by analytic methods is almost impossible, so the gradient vector is obtained directly through the definition of partial derivatives (6).

$$\frac{\partial f(\boldsymbol{W})}{\partial \boldsymbol{W}_{ij}} = \lim_{\delta \to 0} \frac{f(W_{11}, \cdots, W_{ij} + \delta, W_{nn}) - f(W_{11}, \cdots, W_{nn})}{\delta} \tag{6}$$

where δ is taken to be as close to 0 as possible within a computer-processable accuracy. To make the calculated partial derivatives as accurate as possible, $\delta = 10^{-14}$ is used in this paper. The algorithm for calculating the gradient is shown in the function Gradient$(\cdot)$.

```
Function Gradient(X)
Data: The parameter vector X to be learned, Training data feature matrix Features, Training data feature vector Y
Result: Gradient of the objective function ΔX
$\delta = 10^{-14}$;
for $i$ = 0: ($l$+1) × ($l$+1) do
    addtion = zeros(($l$+1) × ($l$ +1));
    addtion[$i$] = $\delta$;
    temp$_1$= Aim (X + addition, Features, Y);
    temp$_2$ = Aim (X, Features, Y);
    $\Delta X[i] = (\text{temp}_1 - \text{temp}_2)/\delta$;
end
return $\Delta X$;
```

With the iteration of the gradient descent method, the objective function will gradually approach the suboptimal value, and the learning step η should not be too large at this time, otherwise, it will easily oscillate back and forth around the suboptimal value, so the learning step needs to be adjusted dynamically, so the following dynamic adjustment strategy is set:

$$\eta = a,\ f(W_{t+1}) > f(W_t). \tag{7}$$

If a is set to a positive number less than 1, the learning step η will become smaller and smaller as the learning result gets closer to the suboptimal solution, and the learning accuracy will increase. a is usually taken to be between [0.9, 0.99].

The gradient descent method is particularly prone to local optimality when solving for the minimum of a complex function. Therefore, when the gradient is detected to have fallen to 0 and the learning effect is still unsatisfactory, the initial value and learning step should be reset and the search for optimality restarted. Generally, a threshold σ of the objective function is set:

$$\begin{cases} \eta = \eta_0 \\ \boldsymbol{W} = \text{rand}(\text{size}(\boldsymbol{W})),\ \ \Delta W == 0; f(w_t) > \sigma \end{cases} \tag{8}$$

where η_0 denotes the initial learning step, which is generally taken as 0.1, and $\text{rand}(\text{size}(\boldsymbol{W}))$ denotes the regeneration of the initial $\boldsymbol{W}$. In summary, the learning algorithm for FCMs based on gradient descent is shown in Algorithm 1.

Algorithm 1 FCM learning algorithm based on gradient descent

```
Data: Training data feature matrix Features,
Each row represents all features of one sample features,
training data feature vector $\boldsymbol{Y}$,
the features of each sample is $y_i$,
the number of samples is $n_{\text{sampies}}$,
the number of features is $l$,
fuzzy cognitive map FCM($C_{\text{in}}$, $\boldsymbol{W}$),
the maximum number of iterations
interation_number,
initial learning step $\eta = \eta_0$,
learning step decay coefficient $a$,
maximum threshold of objective function Threshold
Result: FCM weight matrix $\boldsymbol{W}$
$X$ = rand(($l$+1) × ($l$+1)); /* Initialize the parameters to be learned by gradient descent, of size ($l$+1) × ($l$ + 1) */
for $i$ = 0 : interation_number do Gradient descent
    $\Delta \boldsymbol{X}$ = Gradient($\boldsymbol{X}$);
    $\boldsymbol{X} = \boldsymbol{X} - \eta \Delta \boldsymbol{X}$;
    for $j$ = 0: ($l$+1) × ($l$+1) do Check that the parameters to be learned are not out of bounds
        if abs($X[j] > 1$) then
            $X[j]$ =rand(−1,1)
        end
    end
    $A[i]$ = Aim($\boldsymbol{X}$, Features, $\boldsymbol{Y}$);
    if $A[-1] > A[-2]$ then Dynamic adjustment of learning steps
        $\eta = a\eta$
    end
    if $\Delta X$ == 0 && $A[-1]$ > Threshold then
        $X_0$ = rand(($l$+1) × ($l$ +1));
        $\eta = \eta_0$,
    end
    if $\Delta X$ == 0 && $A[-1]$ < Threshold then
        Break;
    end
end
```

3.3.2 FCM learning algorithm based on PSO

This section focuses on a particle swarm-based FCM learning algorithm. In PSO, each possible solution to the optimal problem is imagined as a bird, called a “particle”. There are m particles in d-dimensional space, and at a given moment, the position of particle i is

$$X_i = (X_{i1}, X_{i2}, \cdots, X_{id}),\ i = 1, 2, \cdots, m. \tag{9}$$

The velocity of particle i is

$$V_i = (V_{i1}, V_{i2}, \cdots, V_{id}),\ i = 1, 2, \cdots, m. \tag{10}$$

The best ever position of particle i passing through is

$$P\text{best}_i = (P\text{best}_{i1}, P\text{best}_{i2}, \cdots, P\text{best}_{id}),\ i = 1, 2, \cdots, m. \tag{11}$$

The best historical position through which the population has passed is

$$G\text{best} = (G\text{best}_1, G\text{best}_2, \cdots, G\text{best}_d). \tag{12}$$

The bird will decide its speed in the process of feeding based on its own experience and the position of other birds in the flock. Based on the current position and speed, it can get the position at the next moment, so that each bird constantly updates its speed position by learning from itself and the flock. The update rule for the ith particle from moment t to moment $t+1$ can be summarized by the following equation:

$$V_i^{t+1} = IV_i^t + C_1 R_1 (\text{Pbest}_i^t - X_i^t) + C_2 R_2 (\text{Gbest}^t - X_i^t), \tag{13}$$

$$X_i^{t+1} = X_i^t + V_i^t, \tag{14}$$

where I is the inertia factor, usually taken as 1; C_1 and C_2 are the learning factors, usually taken as 2; R_1 and R_2 are random numbers between (0, 1). The above equation is updated for one particle in the population, and all particles in the population are updated in turn using the above equation.

For the N-node FCM, similar to the gradient descent method, each particle represents all values of the weight matrix.

$$\boldsymbol{X} = (w_{11}, \cdots, w_{1N}, w_{21}, \cdots, w_{2N}, \cdots, w_{N1}, \cdots, w_{NN}) \tag{15}$$

Each step of the update determines whether the velocity of each particle is within the specified range, and forces individuals whose velocity exceeds the maximum velocity of the particle back into the range $(-V_{\max}, V_{\max})$. In the later stages of learning an FCM using the particle swarm algorithm, the velocity range of each particle needs to be contracted, similar to the gradient descent method of dynamically adjusting the learning step, and a strategy for dynamically adjusting the velocity range of the particles is designed as follows:

$$V_{\max} = aV_{\max}, f(W_{t+1}) == f(W_t). \tag{16}$$

With a set to a positive number less than 1, the learning step η gets smaller and smaller as the learning result gets closer to the suboptimal solution, and the learning accuracy increases. a is usually taken to be between [0.9, 0.99]. In summary, the algorithm for learning FCMs using the particle swarm algorithm is shown in Algorithm 2.

Algorithm 2 FCM learning algorithm based on PSO

```
Data: Features, $Y$, $y_i$, $n_{\text{samples}}$, $l$ , FCM($C_{\text{in}}$, $W$), interation_number, $V_{\max}$, $a$, Threshold, $PN$
Result: FCM weight matrix $W$
$X$ = rand([$PN$, ($l$+1) × ($l$+1)]);
/* Initialize the particle swarm, each row is one particle, size of each particle is ($l$+1) × ($l$+1) */
for $t$ = 0: interation_number do PSO
  for $i$ = 0 : $PN$ do
    temp = Aim($X$[$i$], Features, $Y$) if
    temp < $p$fit[$i$] then Update Individual
    Optimum
      $p$fit[$i$] = temp;
      $p$best[$i$] = $X$[$i$];
      if $p$fit[$i$] < fit then Update group
        optimal
          $g$best = $X$[$i$];
          fit = $p$fit[$i$];
        end
    end
      $V$[$i$] = $IV$[$i$]+$C_1R_1$($p$best[$i$]−$X$[$i$])+$C_2R_2$($g$best−$X$[$i$]); //Update
      particle velocity
      $X$[$i$] = $X$[$i$] + $V$[$i$]; //Update
      particle position
  end
  if afs($X$[$i$][$j$ ]) > 1, abs($V$[$i$][$j$]) > $V_{\max}$ then Learning whether parameters are out of bounds
  $X$[$i$][$j$] = rand(−1, 1), $V$[$i$][$j$]=
  rand(−$V_{\max}$, $V_{\max}$)
  end
  fitness[$i$] = Aim($g$best, Features, $Y$);
  if fitness[−1] == fitness[−2] then Dynamic adjustment of learning steps
    $V_{\max}$=$aV_{\max}$
  end
  if fitness[−1] < Threshold then
    Break;
  end
end
```

3.3.3 Hybrid FCM learning algorithm based on PSO and gradient descent

The particle swarm algorithm has a better global search capability compared to the gradient descent method and can converge quickly to near the optimal or sub-optimal solution. However, as it is an evolutionary algorithm, it relies on comparing the fitness of a large number of solutions to find the optimal or suboptimal solution, so it is characterized by the fact that the value of the objective function decreases very rapidly at the beginning of the iteration, but the value of the objective function changes very little in subsequent iterations, often hovering around the optimal solution and taking up a lot of computation time. The gradient descent method, on the other hand, is a commonly used optimization algorithm with a very clear objective: to update the variables in the direction of a decreasing error function. However, the selection of the location of the initial values of the variables has a crucial impact on the results of the gradient descent method, and even directly affects the effectiveness of the algorithm. The objective function of the FCM learning algorithm is very complicated and the images are rugged, so the selection of the initial weight values is very important. Combining the above characteristics of the two algorithms, the two algorithms can be combined to further improve the accuracy: the weight matrix obtained from the optimization of the particle swarm algorithm is used as the initial weight matrix of the gradient descent method, which is equivalent to the particle swarm algorithm finding a solution near the optimal solution for the gradient descent method with its superior global search capability, and to avoid the particle swarm algorithm hovering near the optimal solution, followed by the gradient descent method to further fine-tune the weight matrix so that the weight matrix is as close to the true value as possible.

In order to achieve the desired value for both speed and accuracy of the algorithm, it is necessary to find the optimal point at which the two algorithms combine, i.e., the minimum number of iteration steps near which the particle swarm algorithm finds the global optimal solution when the solution output by the particle swarm algorithm not only takes less time but also tends to make the result of the gradient descent method closer to the true value. A relatively large threshold is generally set, and when the value of the objective function is less than this threshold, PSO is no longer run, and instead, the training parameters are passed to the gradient descent method to continue the learning process. The FCM learning algorithm based on PSO and gradient descent is shown in Algorithm 3, and the flow chart of the process is shown in Fig. 4.

Algorithm 3 FCM learning algorithm based on PSO and gradient descent

```
Data: Training data feature matrix Features, each row represents all the features of one sample feature, the training data feature vector $Y$, the features of each sample are $y_i$, the number of samples is $n_{\text{samples}}$
Result: FCM weight matrix $W$
$W$ = PSO(Features, $Y$) ; // Algorithm 2
$W$ = Gradientiy($W$, Features, $Y$);
// Algorithm 1, further optimization of the weight matrix
```

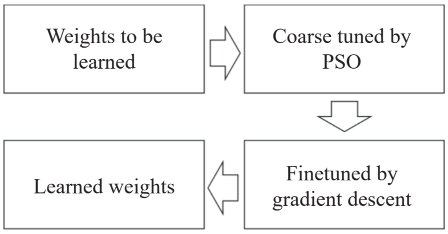


Fig. 4 Flow chart of the FCM learning algorithm based on PSO and gradient descent

4. Data preprocessing

The raw data obtained from the eye tracker and physiological instruments consisted of 30 columns, including physiological data (6 columns), oculomotor data (22 columns), and time markers (2 columns). The average binocular coordinates (2 columns), time markers (2 columns), binocular coordinates (4 columns), and binocular sweep (gaze) marker columns (4 columns), which are not relevant to the learning purpose, are now eliminated, leaving only the physiological data (6 columns), binocular pupil data (6 columns), binocular sweep time (2 columns), binocular gaze time (2 columns), and binocular sweep angle (2 columns). Also, as the sampling frequency of each sensor is different, the sensors with low sampling frequency in the table have many missing values in the corresponding columns, so the data are classified according to the sampling frequency of the sensors.

(i) Respiratory-electromyogram (EMG) data. Includes EMG, thoracic respiration, and diaphragmatic respiration.

(ii) Heart rate-blood oxygen saturation (SaO2). Includes heart rate (upper and lower limits), SaO2.

(iii) Ocular motility sensor. Includes left (right) pupil height, pupil width, pupil area, left (right) eye gaze marker, left (right) eye sweep marker, left (right) eye sweep time, left (right) eye sweep angle of view.

After classifying the data into the three types mentioned above, the missing values due to the sampling frequency are removed and the data from the first 20 s of the “freezing” moment is selected as valid data according to the time marker. In addition to missing values due to low sampling frequencies, there are also missing values due to temporary equipment failures, which are interpolated using K-nearest neighbor (K-NN) for random missing values [25] in continuous data sequences and rejected for large missing values due to temporary equipment failures.

The data from each source is processed and relevant features are extracted below.

4.1 Respiratory-EMG sensor data

(i) Electromyographic data. The surface electromyography (sEMG) of the long wrist extensors of the arm used to control the grip is collected during the experiment and the EMG data collected during the experiment is shown in Fig. 5.

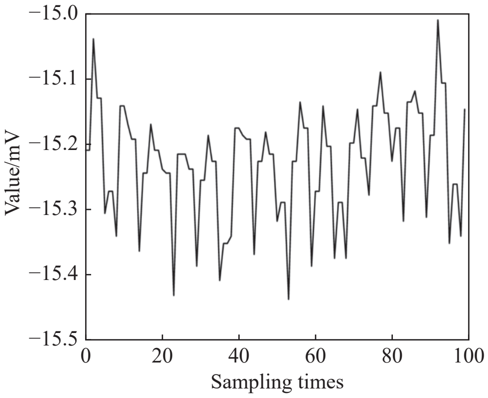


Fig. 5 Example of electromyographic data

The overall trend of the EMG data indicates the movement performed, and the magnitude of the EMG reflects the degree of muscle tension. In conducting the state of postural awareness assessment, it is considered that the degree of muscle tension partially reflects the state of SA [26], and therefore the metric relating to the magnitude of EMG is selected [27–29]. The following indicators are extracted as alternative features.

i) Integral of absolute value (IAV)

$$\mathrm{IAV}=\frac{1}{N}\sum_{i=1}^{N}|x_i| \tag{17}$$

ii) Mean square value (MSV)

$$\mathrm{MSV}=\frac{1}{N}\sum_{i=1}^{N}x_i^2=\mathrm{E}\left[X^2\right] \tag{18}$$

The MSV represents the energy of the signal and is the second-order moment of the signal.

iii) Variance (VAR)

$$\mathrm{VAR}=\frac{1}{N-1}\sum_{i=1}^{N}(x_i-\overline{x_i})^2 \tag{19}$$

The variance represents the dynamic component of the signal energy (the mean square is the static component) and is the second-order center moment.

iv) Root mean square (RMS)

$$\mathrm{RMS}=\sqrt{\frac{\sum_{i=1}^{N}x_i^2}{N}}=\sqrt{\frac{x_1^2+x_2^2+\cdots+x_N^2}{N}} \tag{20}$$

v) Willison amplitude (WAMP)

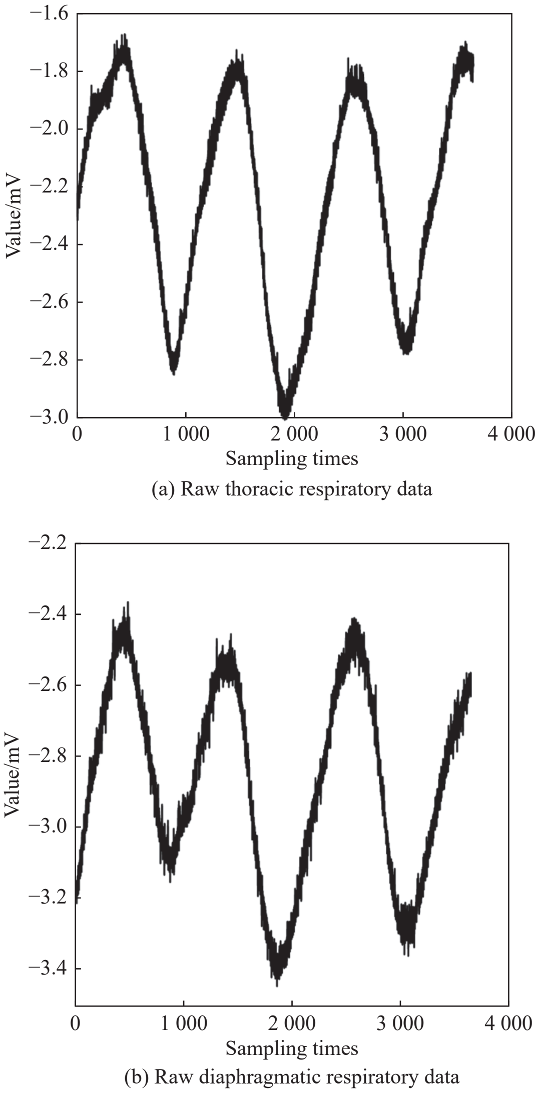
$$\mathrm{WAMP}=\sum_{i=1}^{N}f(|x_i-x_{i+1}|) \tag{21}$$

$$f(x)=\begin{cases}1, & x>\text{Threshold}\\ 0, & \text{otherwise}\end{cases}$$

Threshold is often set as 50−100 μV, in this experiment, it is 50 μV.

(ii) Thoracic respiration signal (TRS) and diaphragmatic respiration signal (DRS)

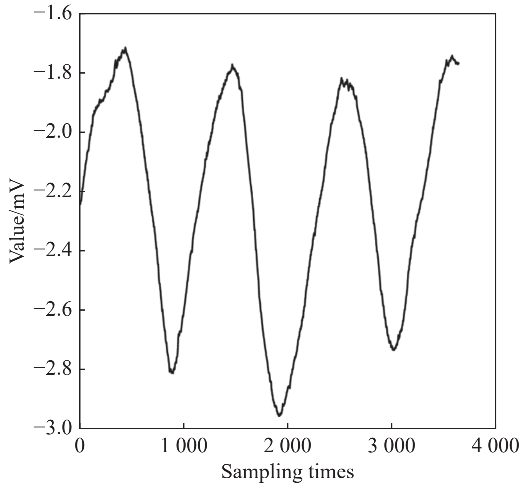
As both human thoracic respiration and diaphragmatic respiration are measured in relation to human respiration, the data for thoracic respiration and diaphragmatic respiration are similar and their original data are shown in Fig. 6. Because of this similarity, the same treatment is adopted. It is found that there are many burrs in the raw data due to measurement errors, so the data need to be filtered to remove the noise. In this paper, the data are smoothed by using the Kalman filter [30] and the observed data variance is chosen to be 1. The data after noise removal is shown in Fig. 7.



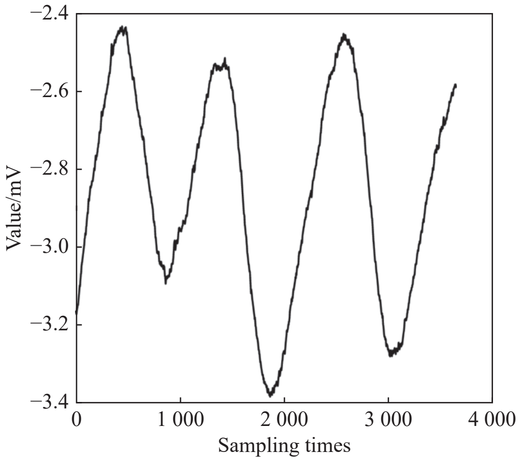
(a) Raw thoracic respiratory data

(b) Raw diaphragmatic respiratory data

Fig. 6 Raw thoracic respiratory data and diaphragmatic respiratory data



(a) Thoracic respiratory data after Kalman filtering



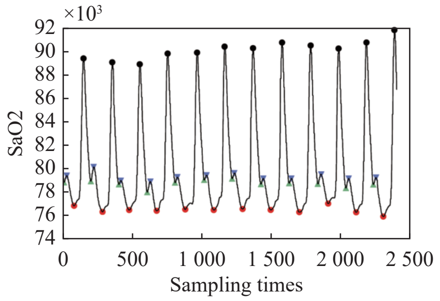
(b) Diaphragmatic respiratory data after Kalman filtering

Fig. 7 Thoracic respiratory data and diaphragmatic respiratory data after Kalman filtering

As with EMG data, absolute mean, mean square, variance, and RMS values are selected as alternative features, as well as period T , frequency f , peak P , trough V , and mean peak to peak (MPP), due to the apparent periodicity of the thoracic (diaphragmatic) respiratory data.

4.2 Heart rate-blood SaO2 sensor data

(i) Blood SaO2 data. A finger-held photoelectric sensor is used to calculate hemoglobin concentration and blood SaO2 by simply placing the sensor on a human finger and using the finger as a transparent container for hemoglobin, using red light at a wavelength of 660 nm and near-infrared light at 940 nm as the incoming light source, and measuring the light transmission intensity through the tissue bed. The raw data of the measured blood SaO2 are shown in Fig. 8.



— : SaO2; • : Crest; • : Trough; ▾ : Sub-crest; ▴ : Sub-trough.

Fig. 8 Raw data of the measured blood SaO2

As can be seen from Fig. 8, the data is basically smooth and the noise has little effect on the data, so no filtering is performed. At the same time, it can be found that the SaO2 signal shows obvious periodicity, which is caused by the pumping of the heart. It can be found that within each cycle there are also two small peaks and troughs, which are called sub-crests and sub-troughs in this paper. At the same time, a cycle reflects the duration of the heartbeat, so the minimum number of cycles within the time frame can be counted to calculate the de-rate data. Thus, the following alternative indicators can be obtained from SaO2: the mean, variance, effective value, and IAV, as well as the mean and variance of a peak, trough, cycle, heart rate, peak-peak, sub-crests and sub-troughs ’ cycle, frequency, peak, trough, and peak-peak values (24 alternative indicators in total).

(ii) Heart rate data. The heart rate data in this experiment gives the upper and lower limits of heart rate respectively, and its mean and variance are selected as alternative indicators (4 alternative indicators in total).

4.3 Eye-tracker data

The eye-tracker data mainly include pupil height, pupil width, pupil area, left (right) eye gaze marker, left (right) eye sweep marker, left (right) eye sweep time and left (right) eye sweep angle of view. The data are not significantly periodic or regular, but there are more missing data and outliers.

(i) Processing of pupil height, pupil width, and pupil area data for both eyes.

To facilitate the processing and visualization of the data, the z-score is first used.

$$f(x) = \frac{x-\mu}{\sigma} \tag{22}$$

where x is the observed value, μ is the overall mean, and σ is the overall standard deviation. Considering a large number of outliers in the data given by the oscillograph, in order to eliminate the influence of outliers on z-score normalization, a robust scale is used to calculate the mean and standard deviation using the data between the first and third quartiles, followed by z-score normalization. The distribution of the standardized data is shown in Fig. 9.

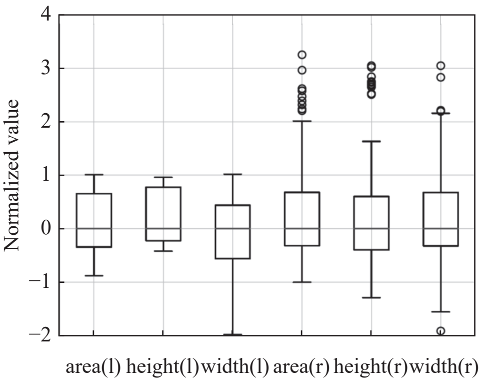


Fig. 9 Box plot of pupil data distribution for both eyes

The outliers in the data are then processed. Outliers are first detected by the Tukey test: outliers are detected using the interquartile range (IQR), which is the difference between the upper quartile and the lower quartile. Using 1.5 times IQR as standard, points exceeding 1.5 times the IQR in the upper quartile, or 1.5 times the IQR in the lower quartile, are specified as outliers [31]. After labeling these outliers, the K-NN algorithm is used to interpolate the outliers and missing values in the dataset. The Euclidean distance is used to find the nearest sample point. Each missing feature is estimated using the values of the n nearest sample points, which are weighted according to their distance from the sample point to be estimated. When the number of available nearest sample points is less than n, the average value of the feature is used for interpolation. In this paper, n is taken to be 5. The distribution of the data after interpolation using K-NN is shown in Fig. 10.

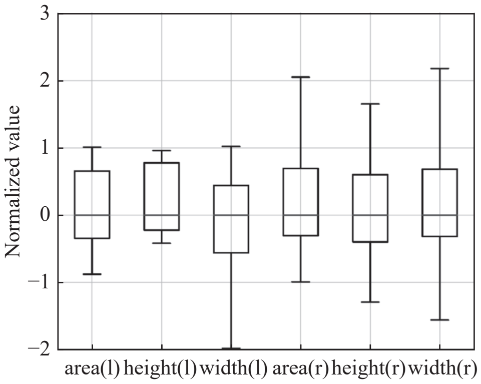


Fig. 10 Box plot of pupil data distribution for both eyes (after processing outliers and missing values)

After dealing with outliers and missing values, the mean and variance of the above indicators are calculated as alternative characteristics.

(ii) Gaze and sweep data. The sweep and gaze data focus on their number, maximum and minimum durations, average duration and variance of gaze (sweep), and duration and variance of sweep angle.

After processing all of the above data, a total of 101 alternative features are obtained. Of these, there are 40 features from eye movement data, five features from EMG, 14 features each from thoracic and diaphragmatic respiration, and 28 features from heart rate-SaO2 sensors. The features are considered for filtering to prevent dimensional catastrophes caused by high dimensionality.

4.4 Feature selection

The above features are filtered by using the random forest [32]. Using the 101 features obtained above, the results obtained from the SAGAT scale are used as labels, subjected to maximum-minimum normalization, and then used as training data for the random forest, which is trained with parameters set to 100 decision trees, each with a maximum depth of 10. Finally, the importance of each feature is obtained as shown in Fig. 11.

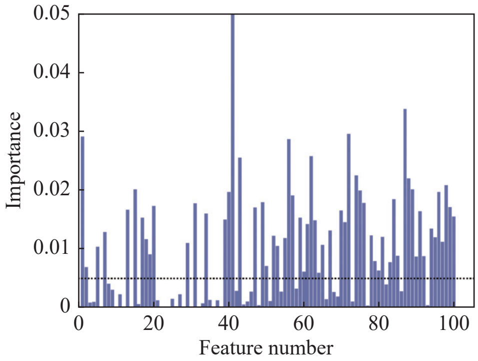


Fig. 11 Importance of each feature

The horizontal axis of Fig. 11 indicates the number of each feature. After eliminating features with importance less than 0.005, 59 features remain, and the sum of the importance of these 59 features is 0.949 7.

5. Results and analysis

The 59 features and 1 column of labels are used as each node of the FCM, i.e., a FCM with 60 nodes is constructed by the learning algorithm. In the learning data, missing data due to a sensor failure is treated as 0. A total of 200 sets of data are collected for the experiments, and 80% are randomly used as the training set and the remaining 20% as the test set. The PSO algorithm, the gradient descent algorithm, and the hybrid learning algorithm of PSO and gradient descent are used to construct the FCM respectively.

5.1 Metrics for the results

The effectiveness of the model fitting is evaluated by the following metrics.

5.1.1 Explained variance (EV) score

$$\mathrm{EV}(y,\hat{y}) = 1 - \frac{\mathrm{Var}\{y-\hat{y}\}}{\mathrm{Var}\{y\}} \tag{23}$$

where $\widehat{y}$ is the predicted value, y is the corresponding true value, and $\mathrm{Var}\{y\}$ is the variance. Its value takes the range [0,1], the closer to 1 means that the independent variable explains more of the variance of the dependent variable of the variance, with smaller values indicating poorer effects.

5.1.2 Maximum error (ME)

This metric calculates the maximum residual, i.e., the ME between the predicted and true values. In a fully fitted single-output regression model, the ME on the training set is 0, although in the real world this is almost impossible. This indicator shows the degree of error in the fit of the model. If $\widehat{y_i}$ is the predicted value of the ith sample and y_i is the corresponding true value, then the ME is defined as

$$\mathrm{ME}(y,\hat{y}) = \max(|y_i - \hat{y}_i|). \tag{24}$$

5.1.3 Mean absolute error (MAE)

If $\widehat{y_i}$ is the predicted value of the ith sample and y_i is the corresponding true value, then the MAE is defined as

$$\mathrm{MAE}(y,\hat{y}) = \frac{1}{n_{\text{samples}}}\sum_{i=0}^{n_{\text{samples}}-1}|y_i - \hat{y}_i|. \tag{25}$$

5.1.4 Mean squared error (MSE)

MSE is defined as

$$\mathrm{MSE}(y,\hat{y}) = \frac{1}{n_{\text{samples}}}\sum_{i=0}^{n_{\text{samples}}-1}(y_i - \hat{y}_i)^2. \tag{26}$$

5.1.5 Median absolute error (MedAE)

MedAE is defined as

$$\mathrm{MedAE}(y,\hat{y}) = \mathrm{median}(|y_1 - \hat{y}_1|,\cdots,|y_n - \hat{y}_n|). \tag{27}$$

The median of the residuals as a measure of performance is highly robust and does not change drastically due to the presence of outliers.

5.1.6 Coefficient of determination, R^2

$$R^2(y,\hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{28}$$

The coefficient of determination indicates the proportion of the variance in the dependent variable that can be explained by the independent variables in the model, providing an indication of the goodness of fit and thus a measure of the model's performance in predicting the unobserved sample by the proportion of variance explained. The best possible score for the decidability coefficient is 1.0, but it can also be negative. A model whose predicts value is always the expectation of y, ignoring input features, has a score of 0.

5.2 The results

The values of the objective function with the number of iteration steps for learning FCMs using the PSO algorithm are shown in Fig.12.

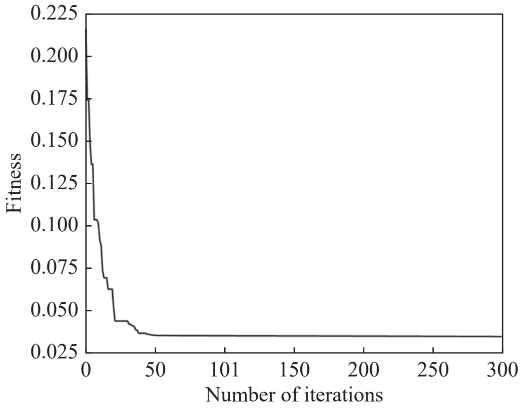


Fig. 12 Learning FCM objective function values with iteration steps using the PSO algorithm

The objective function for learning FCMs using gradient descent and the change in the l_2 norm of the gradient with the number of iterations is shown in Fig. 13.

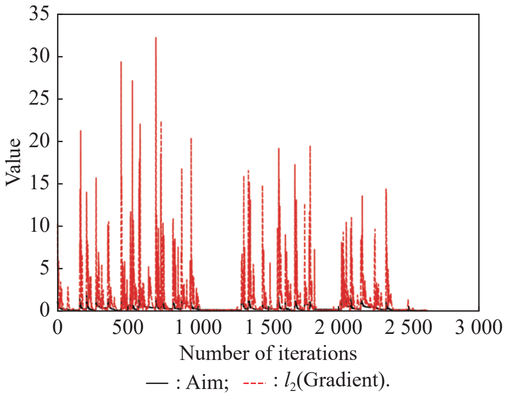


Fig. 13 Aim function values and l_2 values optimized by gradient descent method

As the learning effect of gradient descent is very dependent on the initial value, the learning effect is poor and unusable when the initial value is randomly given, so the required accuracy is achieved only after several reselections of the initial value are made. From Fig. 13, we can see that the gradient and the objective function go through several large ups and downs before reaching the required accuracy.

The variation of the objective function value with the number of iteration steps using a learning algorithm combining the PSO learning algorithm and gradient descent, and the variation of the number of second parity of the gradient with the number of iteration steps are shown in Fig. 14.

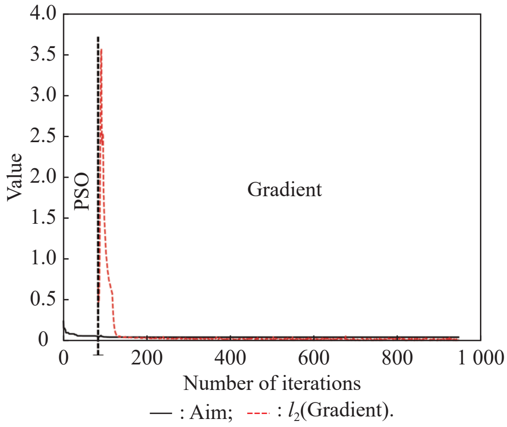


Fig. 14 Objective function curve and gradient (l_2) curve for FCM using PSO and gradient descent

As can be seen from Fig. 14, the objective function value is optimized below a threshold by the PSO algorithm in the first stage, followed by a gradient descent stage for fine-tuning.

The performance metrics of each learning algorithm on the training set are shown in Table 1, and on the test set in Table 2.

Compare the metrics for the training set and test set. It can be found that the PSO reached the best learning result, for its EV, and R^2 are perfect and the ME, MAE, and MSE are the lowest, but its running time is the longest. Almost perfect results in the training set may indicate over-fitting, and the long-running time shows a relatively low efficiency. As for the gradient descent, it can be found that it is the least effective learning algorithm, for its lowest R^2 and EV, the highest ME, MAE, and MedAE. Now scrutinize the hybrid algorithm, firstly, the PSO in the hybrid algorithm uses about one-third of the time of pure PSO to get a relatively good result, and then the gradient descent algorithm finetunes the results, using about 900 0 s, to get a good result. The hybrid algorithm obtains the almost same quality as the pure PSO, but the running time is about two-third of the pure PSO. Compared with Table 1 and Table 2, it is easy to check over-fitting or under-fitting. It can be found that all of the algorithms may confront a little bit of over-fitting, for EV and R^2, all of them decrease a little bit; for the ME, MAE, MSE, and MedAE, all of them increase a little bit. However, such little change of efficiency causes a negligible impact.

Table 1 Performance metrics for each learning algorithm (training set)

Learning algorithm	EV	ME	MAE	MSE	MedAE	R^2	Running time/s
Random forest	0.8960	0.2020	0.0843	0.0109	0.0710	0.8869	**0.1098**
PSO	**1.0000**	**0.0039**	**0.0012**	**0.0000**	**0.0005**	**1.0000**	21244.7169
Gradient descent	0.8766	0.5404	0.1258	0.0479	0.0406	0.8753	9283.7482
PSO (in hybrid algorithm)	0.9757	0.2740	0.0614	0.0099	0.0368	0.9743	6376.7343
Hybrid algorithm	0.9994	0.0350	0.0119	0.0002	0.0091	0.9994	14767.0731

Table 2 Performance metrics for each learning algorithm (test set)

Learning algorithm	EV	ME	MAE	MSE	MedAE	R^2	Running time/s
Random forest	0.8775	0.2025	0.0994	0.0259	0.0768	0.8780	**0.1098**
PSO	**0.9957**	**0.0056**	0.0306	0.0476	**0.0118**	0.9696	21244.7169
Gradient descent	0.8688	0.5866	0.1449	0.0599	0.0863	0.8708	9283.7482
PSO (in hybrid algorithm)	0.9500	0.2996	0.0705	**0.0250**	0.0593	0.9718	6376.7343
Hybrid algorithm	0.9680	0.0695	**0.0165**	0.0274	0.0189	**0.9800**	14767.0731

6. Conclusions

This paper focuses on the SA status measurement based on the FCM model. To the state of art, only the SAGAT method to measure the SA status has good statistical validity. However, the SAGAT cannot be applied in real-time since the operators have to be interrupted from the missions. In this case, lots of missions that need the operator's real-time SA status cannot be solved. To tackle such a problem, we use the eye-tracker and physiological data to predict the SAGAT results, since the data comes from the eye-tracker and physiological instruments are highly relevant to human's cognitive status and can be obtained in real-time. The learning machine is chosen as FCM, because of its interpretability and ease to use. However, there still not exists an efficient enough learning method for FCM. Thus, we apply hybrid methods of PSO and gradient descent method to learn the FCM and obtain good results.

To summarize, the contributions of this paper are construction of a 60-node FCM to assess the operators' SA status in real-time, and using the hybrid algorithm of PSO and gradient descent to learn the FCM in high efficiency.

The constructed FCM contains the knowledge from data and indicates the links between nodes, which can help us to understand how the movement of the eye and the relevant physiological signal is related to the SA status. Also, the model can assess the operator's SA status in real-time and do not interrupt the missions since the devices can be embedded in the platform, even in the clothes of the operator.

The future work will focus on how to deal with the uncertainty of the data and the dynamic environment which FCM cannot handle. For example, the data of heart rate comes from different sensors may be inconsistent, and the dynamic situation change may cause a sudden loss of SA of the operator, which may not be caught by the FCM. One of the solutions is to use relevant FCM extensions, such as dynamic fuzzy generad grey congnitive maps, to adapt the uncertainty data and dynamic environment, in this way, the learning algorithms for FCM extensions also need to be designed.

References

[1] ENDSLEY M R. Toward a theory of situation awareness in dynamic systems. Human Factors, 1995, 37(1): 32–64.

[2] CHEN H, PANG L P, WANYAN X R, et al. Effects of air route alternation and display design on an operator's situation awareness, task performance and mental workload in simulated flight tasks. Applied Sciences, 2021, 11(12): 5745.

[3] BONGO M, SEVA R. Effect of fatigue in air traffic controllers' workload, situation awareness, and control strategy. International Journal of Aerospace Psychology, 2022, 32(1): 1–23.

[4] XU R B, LIU Z Q, LIN Y M, et al. Unified adaptive deep classification for industrial real-time situation awareness in edge environment. Concurrency Computation: Practice and Experience, 2021: e6488.

[5] ALHAIDER A, LAU N, DAVENPORT P, et al. Distributed situation awareness: a health-system approach to assessing and designing patient flow management. Ergonomics, 2020, 63(6): 682–709.

[6] FENG C, WANYAN X R, LIU S, et al. An approach to situation awareness (SA) assessment in flight simulation: SA dynamic circulation (SADC) model. Human Factors and Ergonomics in Manufacturing, 2021, 31(5): 559–569.

[7] LIANG N, YANG J, YU D, et al. Using eye-tracking to investigate the effects of pre-takeover visual engagement on situation awareness during automated driving. Accident Analysis and Prevention, 2021, 157: 106143.

[8] TABER M. Investigating offshore helicopter pilots' cognitive load and physiological responses during simulated in-flight emergencies. International Journal of Aerospace Psychology, 2021, 31(1): 56–69.

[9] DONG Y J, ZHANG X. Study on the effect of awakening daylight in dormitories on morning alertness, mood, fatigue and sleep quality of college students. Building and Environment, 2021, 203: 108060.

[10] KOSKO B. Fuzzy cognitive maps. International Journal of Man-Machine Studies, 1986, 24(1): 65–75.

[11] WANG J Y, PENG Z, WANG X D, et al. Deep fuzzy cognitive maps for interpretable multivariate time series prediction. IEEE Trans. on Fuzzy Systems, 2021, 29(9): 2647–2660.

[12] CHEN J, GAO X D, RONG J, et al. The dynamic extensions of fuzzy grey cognitive maps. IEEE Access, 2021, 9: 98665–98678.

[13] CHEN J, GAO X D, RONG J. Enhance the uncertainty modeling ability of fuzzy grey cognitive maps by general grey number. IEEE Access, 2020, 8: 163844–163856.

[14] CHEN J, ZHANG Y, GAO X D. Hybrid FCM learning algorithm based on particle swarm optimization and gradient descent algorithm. Proc. of the 16th IEEE International Conference on Control, Automation, Robotics and Vision, 2020: 801–806.

[15] CHEN J, GAO X D, ZHONG L. Using fuzzy grey cognitive maps to model threat assessment for UAVs. Proc. of the IEEE International Conference on Control and Automation, 2018: 594–599.

[16] SALMON P, STANTON N, WALKER G, et al. Situation awareness measurement: a review of applicability for C4I environments. Applied Ergonomics, 2006, 37: 225–238.

[17] NASSER-DINE A, MOISE A, LAPALME J. Does explicit categorization taxonomy facilitate performing goal-directed task analysis? IEEE Trans. on Human-Machine Systems, 2021, 51(3): 177–187.

[18] ENDSLEY M R. Measurement of situation awareness in dynamic systems. Human Factors, 1995, 37(1): 65–84.

[19] LOFT S, BOWDEN V, BRAITHWAITE J, et al. Situation awareness measures for simulated submarine track management. Human Factors, 2015, 57(2): 298–310.

[20] THOMAS M, GOODE N, GRANT E, et al. Can we talk about speed? The effect of verbal protocols on driver speed and perceived workload. Procedia Manufacturing, 2015, 3: 2629–2634.

[21] ORIQUE S, DESPINS L. Evaluating situation awareness: an integrative review. Western Journal of Nursing Research, 2018, 40(3): 388–424.

[22] GUGERTY L. Evidence from a partial report task for forgetting in dynamic spatial memory. Human Factors, 1998, 40(3): 498–508.

[23] RATWANI R M, MCCURRY J M, TRAFTON J G. Single operator, multiple robots: an eye movement based theoretic model of operator situation awareness. Proc. of the ACM/IEEE International Conference on Human-Robot Interaction, 2010: 235–242.

[24] STACH W, KURGAN L, PEDRYCZ W, et al. Genetic learning of fuzzy cognitive maps. Fuzzy Sets and Systems, 2005, 153(3): 371–401.

[25] SAHOO A, GHOSE D K. Imputation of missing precipitation data using KNN, SOM, RF, and FNN. Soft Computing, 2022, 26(12): 5919–5936.

[26] ZARDOSHTI-KERMANI M, WHEELER B C, BADIE K, et al. EMG feature evaluation for movement control of upper extremity prostheses. IEEE Trans. on Rehabilitation Engineering, 1995, 3(4): 324–333.

[27] TRIWIYANTO T, YULIANTO E, WISANA I, et al. Electromyography feature analysis to recognize the hand motion in a prosthetic hand design. Journal of Biomimetics, Biomaterials and Biomedical Engineering, 2021, 50: 25–37.

[28] KHAN S M, KHAN A A, FAROOQ O. Pattern recognition of EMG signals for low level grip force classification. Biomedical Physics & Engineering Express, 2021, 7(6): 065012.

[29] TURNER A, SHIEFF D, DWIVEDI A, et al. Comparing machine learning methods and feature extraction techniques for the EMG based decoding of human intention. Proc. of the 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society, 2021: 4738–4743.

[30] SHARMA S, KULKARNI R. Noise reduction in speckle interferometry fringe pattern using adaptive Kalman filter. Optical Engineering, 2021, 60(12): 124105.

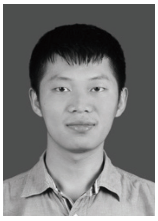
[31] BALOGUN K A, ZUROMSKI L M, KIM R, et al. Establishing age-stratified red blood cell fatty acid reference ranges using model-based clustering and iterative application of the harrisboyd method. Clinical Biochemistry, 2021, 97: 25–33.

[32] FAN L L, YANG J, SUN X, et al. The effects of landsat image acquisition date on winter wheat classification in the north china plain. ISPRS Journal of Photogrammetry and Remote Sensing, 2022, 187: 1–13.

Biographies

CHEN Jun was born in 1979. He received his B.S., M.S., and Ph.D. degrees in system engineering from Northwestern Polytechnical University, China in 2001, 2005, and 2009, respectively. Since 2012, he has been an associate professor with the School of Electronics and Information, Northwestern Polytechnical University, China. His current research interests include modeling and application based on fuzzy cognitive map, and intelligent decision-making for complex autonomous system.

E-mail: junchen@nwpu.edu.cn



GAO Xudong was born in 1996. He received his B.S. degree in detection guidance and control technology from Northwestern Polytechnical University, China in 2018. He is currently pursuing his M.S. degree from Northwestern Polytechnical University, China. His research interests include complex system modeling, data mining, and machine learning.

E-mail: gxdcoolest@mail.nwpu.edu.cn

RONG Jia was born in 1981. She received her B.S., M.S., and Ph.D. degrees from Deakin University in 2012. Her current research interest lies in joint areas of machine learning, deep learning and image processing and their applications in digital health, medical image analysis, cancer diagnosis, cardiac disease detection, pavement condition and maintenance, and tourism management.

E-mail: jiarong@acm.org

GAO Xiaoguang was born in 1957. She received her B.S., M.S., and Ph.D. degrees from Northwestern Polytechnical University, China, in 1982, 1986, and 1989, respectively. She joined the School of Electronics and Information in 1989 and became a professor in 1994. Her research interests include advanced control theory and its applications in complex systems.

E-mail: xggao@nwpu.edu.cn