

Driving fatigue fusion detection based on T-S fuzzy neural network evolved by subtractive clustering and particle swarm optimization

Sun Wei Zhang Weigong Li Xu Chen Gang

(School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China)

Abstract: In order to improve the accuracy and reliability of the driving fatigue detection based on a single feature, a new detection algorithm based on multiple features is proposed. Two direct driver's facial features reflecting fatigue and one indirect vehicle behavior feature indicating fatigue are considered. Meanwhile, T-S fuzzy neural network (TSFNN) is adopted to recognize the driving fatigue of drivers. For the structure identification of the TSFNN, subtractive clustering (SC) is used to confirm the fuzzy rules and their correlative parameters. Moreover, the particle swarm optimization (PSO) algorithm is improved to train the TSFNN. Simulation results and experiments on vehicles show that the proposed algorithm can effectively improve the convergence speed and the recognition accuracy of the TSFNN, as well as enhance the correct rate of driving fatigue detection.

Key words: driving fatigue; fusion detection; particle swarm optimization (PSO); subtractive clustering (SC)

Detection and early warning technology of driving fatigue has become a focus in the field of vehicle active safety. Especially, the methods for detecting driving fatigue based on machine vision have attracted considerable interest recently. However, most present vision detection methods consider only a single fatigue feature of drivers from one aspect, such as frequent blinking^[1] or yawning^[2], abnormal head movement^[3] and vehicle behaviors^[4] etc. With the development of information fusion technology, many detection methods based on multiple fatigue features have been proposed. Unfortunately, they only pay attention to several facial fatigue features of drivers^[5-6], and the indirect features from vehicle behavior such as abnormal lane departure and speed variations are totally ignored. Therefore, the performance of the methods mentioned above is not satisfactory. In addition, because driving fatigue has various causes, driving fatigue detection is a challenging research topic.

At present, artificial neural networks and fuzzy theory have been widely applied in pattern recognition and intelligent decision. It has become possible to detect the driving fatigue by combining fuzzy logic systems with artificial neural networks. In view of this, a novel driving fatigue fusion detection algorithm based on multiple fatigue features is pro-

posed, which considers the indirect fatigue feature of abnormal lane departure as well as two direct facial fatigue features. To improve the performance of the algorithm, first, the structure and the initial parameters of the T-S fuzzy neural network (TSFNN) are determined by subtractive clustering (SC) in order to avoid blindness and randomness. Then the particle swarm optimization (PSO) algorithm is improved to train the TSFNN in order to enhance the optimization efficiency of parameters. Finally, experimental results demonstrate that the proposed algorithm has a good performance.

1 Selection and Calculation of Fatigue Features

There are two main considerations for selecting fatigue features. On the one hand, fatigue features can represent the driving fatigue state of drivers; on the other hand, they are easy to detect and have no negative influence on driving. For comprehensive detection, the percentage of an eyelid closure over the pupil over time (PERCLOS) and yawn frequency (YF) are selected as the direct features of detecting driving fatigue, while the frequency of abnormal lane departure (FALD) is selected as the indirect feature.

1) Calculation of PERCLOS

According to Ref. [1], image processing and pattern recognition technologies are used to estimate whether eyes are open or closed. Suppose that if the opening of eyelid is more than 20%, then eyes are open; otherwise, eyes are closed. If the collected images are N frames per unit of time and there are n_1 frames to be closed, then PERCLOS can be obtained by

$$F_1 = \frac{n_1}{N} \quad (1)$$

2) Calculation of YF

When drivers yawn, the distance H between the upper lip and the lower lip obviously increases, and the distance W between the left corner and the right corner of the mouth decreases. So R_{MO} is defined as the opening rate of the mouth by Eq. (2) to judge whether a driver is yawning or not. H and W can be calculated according to Ref. [2].

$$R_{MO} = \frac{H}{W} \quad (2)$$

If $R_{MO} \geq R_{TH}$ (threshold), the driver is yawning. YF can be calculated by Eq. (3) using the same method as PERCLOS.

$$F_2 = \frac{n_2}{N} \quad (3)$$

where n_2 denotes the frame number of the images where the driver is recognized to be yawning.

Received 2009-06-08.

Biographies: Sun Wei (1980—), male, graduate; Zhang Weigong (corresponding author), male, doctor, professor, zhangwg@seu.edu.cn.

Foundation items: The National Key Technologies R&D Program during the 11th Five-Year Plan Period (No. 2009BAG13A04), the Ph. D. Programs Foundation of Ministry of Education of China (No. 200802861061), the Transportation Science Research Project of Jiangsu Province (No. 08X09).

Citation: Sun Wei, Zhang Weigong, Li Xu, et al. Driving fatigue fusion detection based on T-S fuzzy neural network evolved by subtractive clustering and particle swarm optimization[J]. Journal of Southeast University (English Edition), 2009, 25(3): 356 – 361.

3) Calculation of FALD

Vehicles may deviate from the lane when drivers are fatigued. Whether a vehicle is deviating from a lane can be expressed as the following decision rule^[7-8]:

$$(\xi(k) < \lambda_{L1}) \text{ OR } (\xi(k) > \lambda_{R1}) \text{ OR } ((\xi(k) < \lambda_{L2}) \text{ AND } (\Delta\xi(k) < -\mu_L)) \text{ OR } ((\xi(k) > \lambda_{R2}) \text{ AND } (\Delta\xi(k) > -\mu_R)) \quad (4)$$

where $\xi = (\pi/2 - \theta_L) / (\theta_R - \pi/2)$, θ_L and θ_R express the slope angles of the left and right lane lines, respectively. $\xi(k)$ denotes the ξ value in the k -th frame image, $\Delta\xi(k) = \xi(k) - \xi(k-1)$. λ_{L1} and λ_{R1} are the thresholds of left deviation and right deviation when a vehicle seriously deviates from a lane, respectively. λ_{L2} and λ_{R2} signify the thresholds of left deviation and right deviation when a vehicle mildly deviates from a lane, respectively. μ_L and μ_R represent the thresholds of left deviation tendency and right deviation tendency, respectively.

Besides, the operation of direction signal lights and Eq. (4) are combined to judge whether the vehicle deviates from the lane abnormally.

FALD can be denoted as

$$F_3 = \frac{n_3}{N} \quad (5)$$

where n_3 is the frame number of images where a vehicle is judged to be deviating from the lane abnormally.

2 Design of TSFNN Evolved by SC and IPSO

2.1 Structure of TSFNN

A TSFNN is introduced in this paper and its structure is shown in Fig. 1^[9]. The TSFNN consists of an antecedent network and a consequent network. Typical fuzzy inference rules are as follows:

$$R_j: \text{if } x_1 \text{ is } A_1^j \text{ And } x_2 \text{ is } A_2^j \text{ And } \dots \text{ And } x_n \text{ is } A_n^j \\ \text{then } y_j = \kappa_0^j + \kappa_1^j x_1 + \kappa_2^j x_2 + \dots + \kappa_n^j x_n$$

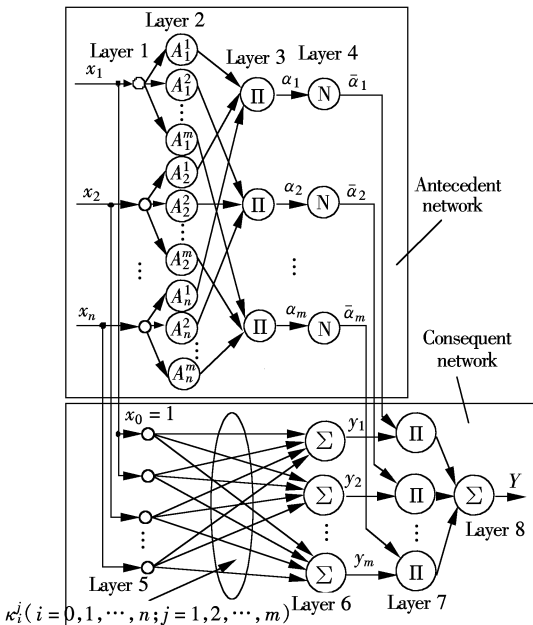


Fig. 1 Structure of TSFNN

where A represents the fuzzy sets of the input space; κ is the consequent parameter, and j denotes the number of fuzzy rules.

The antecedent network is composed of four layers. The functions in each layer are as follows:

1) Layer 1 is the input layer, and node number n is equal to the dimensions of the input variables.

2) Layer 2 is the fuzzy layer, and the node number is the same as the total number of linguistic values of each input variable. Each node expresses a fuzzy subset A_i^j described by the Gaussian membership function,

$$\mu_i^j = \mu_{A_i^j}(x_i) = \exp\left[-\frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}\right] \quad i = 1, 2, \dots, n; j = 1, 2, \dots, m \quad (6)$$

where c_{ij} and σ_{ij} are the center and the width of the Gaussian membership function, respectively.

3) Layer 3 is the rule layer, which is used to calculate the firing strength α_j of every fuzzy rule. Node number M is equal to the number of the fuzzy rules m .

$$\alpha_j = \prod \mu_i^j \quad j = 1, 2, \dots, M \quad (7)$$

4) Layer 4 is the normalized layer to calculate the normalized firing strength of corresponding rules, where the node number is equal to that of layer 3,

$$\bar{\alpha}_j = \frac{\alpha_j}{\sum_{i=1}^M \alpha_i} \quad j = 1, 2, \dots, M \quad (8)$$

The following are the functions of every layer in the consequent network:

1) Layer 5 is the input layer, and it has an additional input node $x_0 = 1$ compared with layer 1 in order to compensate the constant of the consequent network.

2) Layer 6 is the function layer to calculate the consequent parameters of every rule. The linear relationship between input and output in every layer can be described by

$$y_j = \kappa_0^j + \kappa_1^j x_1 + \kappa_2^j x_2 + \dots + \kappa_n^j x_n \quad j = 1, 2, \dots, M \quad (9)$$

where $\kappa_i^j (0 \leq i \leq n)$ represents the weight of the consequent network.

3) Layer 7 is the combined layer to fuse layer 4 and layer 6, which can be expressed as

$$\bar{y}_j = \bar{\alpha}_j y_j \quad j = 1, 2, \dots, M \quad (10)$$

4) Layer 8 is the output layer with only a node to obtain the ultimate reference results of the whole system.

$$Y = \sum_{j=1}^M \bar{y}_j \quad (11)$$

2.2 Learning algorithm

The network learning algorithm includes two parts: one is structure learning, and the other is parameter learning. For the former, SC is used to determine the initial structure of the network by extracting rules from the training samples given. For the latter, the improved PSO (IPSO) is intro-

duced to adjust the system parameters dynamically for making the network more accurately reflect the mapping relationships between input and output.

2.2.1 SC for determining network structure

To make the TSFNN applicable, SC is applied to reduce the fuzzy rules in the TSFNN. It can partition the input-output data into groups called clusters, and modify the structure of the TSFNN with the minimum rules for distinguishing the fuzzy quantities with respect to each cluster^[10]. The SC algorithm can be described as follows:

1) Sample data are normalized. Then set accept rate R_{ac} and reject rate R_{re} of cluster, respectively.

2) For the set of n' data points $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k, \dots, \mathbf{x}_{n'}\}$, where $\mathbf{x}_k = \{x_{k1}, \dots, x_{km'}, x_{k(m'+1)}\}$ in $m' + 1$ dimensional space, $x_{k1}, \dots, x_{km'}$ denote m' input values of the TSFNN, $x_{k(m'+1)}$ denotes the output value. Let $D_{i'}$ be the potential value of data point $x_{i'}$ being a cluster center, which is defined as

$$D_{i'} = \sum_{j'=1}^{n'} \exp\left(-\frac{\|\mathbf{x}_{i'} - \mathbf{x}_{j'}\|^2}{(R_a/2)^2}\right) \quad (12)$$

where $R_a > 0$ denotes the neighborhood radius of each cluster center.

3) Compute $D_{i'}$ for each \mathbf{x}_k using Eq. (12) and let \mathbf{x}_c^1 with the highest D_c^1 be the first cluster center. Then update $D_{i'}$ for other data points by

$$D_{i'} = D_{i'} - D_c^1 \exp\left(-\frac{\|\mathbf{x}_{i'} - \mathbf{x}_c^1\|^2}{(R_b/2)^2}\right) \quad (13)$$

where $R_b = 1.5R_a$ to avoid cluster centers being too adjacent.

4) If $D_{i'} > R_{ac}$, then a new cluster center emerges, and $p = p + 1$. Where p is the number of clusters.

5) If $D_{i'} > R_{re}$ and $D_{i'}$ is far away from other cluster centers, then a new cluster center emerges, and $p = p + 1$.

6) When a new cluster center emerges, step 3) to 5) are executed repeatedly.

Once SC ends, a set of fuzzy rules will be obtained, and each cluster represents a rule. Suppose that $\mathbf{x}_{ci'}^j$ is the cluster center obtained by SC, $\mathbf{x}_{ci'}^{j'}$ is the nearest cluster center from $\mathbf{x}_{ci'}^j$; namely, the Euclidean distance of between $\mathbf{x}_{ci'}^j$ and $\mathbf{x}_{ci'}^{j'}$ is the shortest. Let $c_{i'j'} = \mathbf{x}_{ci'}^j$ and $\sigma_{i'j'} = 0.5 \times \|\mathbf{x}_{ci'}^j - \mathbf{x}_{ci'}^{j'}\| =$

$0.5 \times \sqrt{\sum_{j=1}^p (\mathbf{x}_{ci'}^j - \mathbf{x}_{ci'}^{j'})^2}$, ($i' = 1, 2, \dots, m'; j' = 1, 2, \dots, p$); accordingly, the initial structure identification of the TSFNN is achieved.

2.2.2 IPSO for optimizing network parameters

PSO is a stochastic, population-based computer algorithm for problem solving. In PSO, particles move in the search space of an optimization problem. The position of a particle represents a candidate solution to the optimization problem at hand. Each particle searches for a better position in the search space by changing its velocity according to Eq. (14)^[11],

$$\left. \begin{aligned} v_{i''j''}(t+1) &= \omega v_{i''j''}(t) + c_1 r_1 [I_{i''j''}(t) - x_{i''j''}(t)] + \\ &\quad c_2 r_2 [\hat{G}_{j''}(t) - x_{i''j''}(t)] \\ x_{i''j''}(t+1) &= x_{i''j''}(t) + v_{i''j''}(t+1) \end{aligned} \right\} \quad (14)$$

where $i'' = 1, 2, \dots, m''$ denotes the i'' -th particle; $j'' = 1, 2, \dots, n''$ represents the j'' -th component of the vector space; c_1 and c_2 are called acceleration coefficients ($c_1 = 2.8$, $c_2 = 1.3$); t is the iteration time; r_1, r_2 are two random numbers distributed uniformly in the range of $[0, 1]$; ω is the inertia weight; $x_{i''j''}(t)$ and $v_{i''j''}(t)$ represent the position and the velocity of the i'' -th particle in the t -th time, respectively. $I_{i''j''}(t)$ is the local best position of each particle; $\hat{G}(t) = \min\{G_{i''}(t)\} (i'' = 1, 2, \dots, m'')$ denotes the global best position of the whole population.

Experiments demonstrate that the traditional PSO has fast convergence in earlier stages, but its convergence becomes slow in the latter as all particles fly toward the optimum and the diversity of the population disappears gradually. So it is difficult to obtain a better result and the particle may even fall into the local optimum. To solve this problem, an evolution rate factor H is introduced and the traditional PSO algorithm is also improved to help the particles escape from the local optimum.

H can be expressed as

$$H = \frac{f(\hat{G}(t-1))}{f(\hat{G}(t))} \quad (15)$$

where $f(\cdot)$ is the fitness function. The greater H , the faster the evolution velocity. In the evolutionary process, if H is always approximately equal to 1 and the global optimum is not obtained, then particles may have fallen into the local optimum. Experiments show that a local optimum can be obtained, when $H < \lambda$, $\lambda \in [2, 5]$ ^[11].

To help particles escape from the local optimum, the traditional PSO algorithm is improved as follows:

1) Chaotic disturbance strategy is introduced to change the flying direction of particles^[12].

2) For inertia weight ω , a decreasing strategy with an exponential function is adopted to enhance the convergence speed^[13].

3) Constriction coefficient τ is introduced to improve the convergence performance^[14].

The IPSO updates the position and velocity of particles by

$$\left. \begin{aligned} v_{i''j''}(t+1) &= \tau \{ \omega v_{i''j''}(t) + c_1 r_1 [I_{i''j''}(t) - x_{i''j''}(t)] + \\ &\quad c_2 r_2 [\hat{G}_{j''}(t) - x_{i''j''}(t)] \} - C_D \\ x_{i''j''}(t+1) &= x_{i''j''}(t) + v_{i''j''}(t+1) \end{aligned} \right\} \quad (16)$$

where C_D is a chaos term, $C_D = \beta(1 - 2r_3)R_{i''}$; $\beta > 0$ is called the chaos constant; r_3 represents a random number in $[0, 1]$. $R_{i''}$ expresses the search step of the i'' -th particle. The constriction coefficient $\tau = 2 / \left| 2 - \phi - \sqrt{\phi^2 - 4\phi} \right|$, $\phi = c_1 + c_2$, $\phi > 4$. Inertia weight $\omega = (\omega_{\max} - \omega_{\min}) / e^{(4t/t_{\max})^2} + \omega_{\min}$. In which, ω_{\max} and ω_{\min} signify the maximum and minimum of ω , respectively; t and t_{\max} denote the current iterative time and the maximal iterative time, respectively.

2.2.3 Network training based on SC and IPSO

The parameters to be encoded in every fuzzy rule include the center and the width of the Gaussian membership function as well as the weight of the consequent network. The structure of the j -th rule to be encoded is as follows:

$$c_1^j \ c_2^j \dots \ c_n^j \ \sigma_1^j \ \sigma_2^j \dots \ \sigma_n^j \ \kappa_0^j \ \kappa_1^j \dots \ \kappa_n^j$$

All the fuzzy rules are linked together to construct a particle. The center and the width of the Gaussian function are initialized according to the SC, and the initial consequent parameters are random numbers in the range of $[0, 1]$.

The mean square error(MSE) is used as the fitness function to evaluate all particles.

$$f(x) = \text{MSE}(\hat{Y}_i(x)) = \frac{1}{s} \sum_{i=1}^s (Y_i(x) - \hat{Y}_i)^2 \quad (17)$$

where s is the number of training samples; $Y_i(x)$ and \hat{Y}_i are the actual output and the desired output, respectively.

Update $I_{rj}(t)$ and \hat{G}_j of all the particles according to the fitness value of every particle, then update the position and velocity of every particle. The training process of the TS-FNN is shown in Fig. 2.

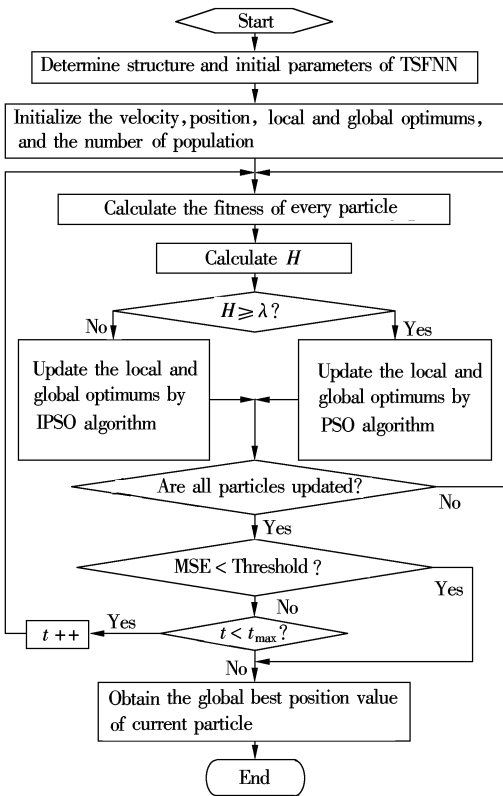


Fig. 2 Flowchart of network training based on SC and IPSO

3 Simulation and Experiments

3.1 Quantification of fatigue degree

To quantify the fatigue degree, a new method that combines the subjective and the objective quantifications is proposed. First, participant drivers can subjectively estimate the fatigue degree and keep the corresponding score for themselves according to their current mental state by Tab. 1. Afterwards, a test program is designed for the objective quantification of fatigue degree inspired by the psychomotor vigilance task(PVT) method^[15]. Before the experiment, the participants are made to learn six traffic sign pictures with corresponding interpretation. During the experiment, one of the six pictures randomly emerges on the program interface every 2 s, participants must respond quickly to the emerging picture within 500 ms by clicking the corresponding inter-

pretation button. When the experiment ends, the program can record the false rate of picture recognition and takes it as the objective quantification value of the fatigue degree and displays it on the interface. Every experiment lasts for 2 min. Define that the higher the false rate, the more fatigued the participants; more-over, if the participants cannot respond to the picture within 500 ms after each picture emerges, then this would be regarded as a false recognition. Finally, considering the subjective score and the objective score, if the objective score is in the range of the subjective score, then the objective score will be taken as the final fatigue degree (FFD).

Tab. 1 Subjective quantification of fatigue degree

Degree	Description of fatigue	Quantification value
1	Not fatigued at all	0 to 0. 2
2	Little fatigued	0. 2 to 0. 4
3	Fatigued	0. 4 to 0. 6
4	Quite fatigued	0. 6 to 0. 8
5	Seriously fatigued	0. 8 to 1

3.2 Acquisition of sample data

For security considerations, experiments are carried out in the special automobile test field in Dingyuan, Anhui province and the experimental vehicle is equipped with necessary security devices. There are two cameras in the experimental vehicle which are used to collect the driver’s facial image and the lane line image in front of the vehicle, respectively. Physiological researches of human body show that the mental state achieves its best at about 10:00 am and 4:00 pm; at 1:00 to 3:00 pm it is a little fatigued, and at around 1:00 to 3:00 am it reaches the worst. So sample data with different fatigue degrees can be obtained as long as experiments are performed at different time sections. In this paper, 80 sample data of 20 drivers are collected at every time section as shown in Tab. 2. Sixty samples among them are used to train the TSFNN, and the other 20 samples are used to test the TSFNN.

Tab. 2 Experimental data(80 samples)

No.	PERCLOS	YF	FALD	FFD
1	0. 067	0. 160	0. 102	0. 008
2	0. 683	0. 262	0. 198	0. 620
⋮	⋮	⋮	⋮	⋮
80	0. 867	0. 833	0. 628	0. 842

3.3 Training TSFNN

A 3-input and single-output neural network is constructed. Input variables $x_i(i = 1, 2, 3)$ express the PERCLOS, YF and FALD, respectively. Output variable Y expresses the fatigue degree of drivers. 80 samples are divided into four clusters by SC, so the number of fuzzy rules is determined as four and the number of linguistic values for every input variable is also four. Tab. 3 shows the comparison of the characteristics between the ordinary TSFNN and the TSFNN with SC.

Tab. 3 indicates that the network structure has an obvious improvement after SC is adopted. In addition, although the TSFNN with SC has an increased number of linguistic val-

ues for every input variable, the total number of rules and the total number of the parameters required for training the network decrease remarkably.

Tab. 3 Comparison of the characteristics between ordinary TSFNN and TSFNN with SC

Parameters of TSFNN		Without SC	With SC
Input-output space		3 inputs, 1 output	3 inputs, 1 output
Shape of membership function		Gaussian	Gaussian
Number of linguistic values		3	4
Number of fuzzy rules		27	4
Number of parameters for training	Nonlinear	18	24
	Linear	108	16
Total		126	40

The TSFNN is trained using the BP and IPSO, respectively, and their MSE curves are shown in Fig. 3 and Fig. 4. Fig. 3 and Fig. 4 reveal that the MSE value of the BP algorithm declines from 4.84 to 0.001 after 1996 iterations, and it produces many oscillations and has a slow convergence speed. However, the IPSO can search for the global optimal solution of TSFNN parameters effectively with a smooth and steady MSE curve and a fast convergence speed. For a same MSE, the IPSO will cost less time of iteration than BP. Thus, the IPSO has a better performance in convergence speed, accuracy, and computational complexity compared with BP.

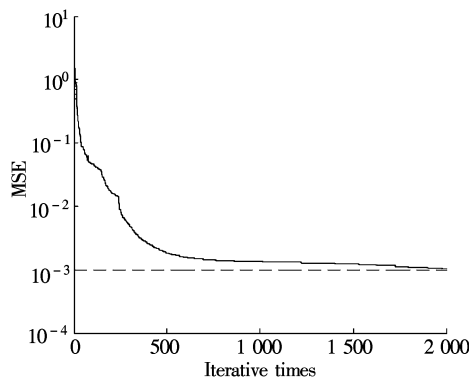


Fig. 3 MSE curve based on BP

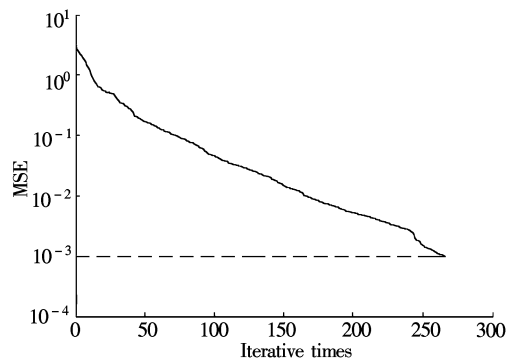


Fig. 4 MSE curve based on IPSO

Twenty sample data are used to test the TSFNN optimized by the IPSO and BP, respectively. The errors between the desired outputs and the actual outputs are shown in Fig. 5. It can be seen from Fig. 5 that, the maximal error and the MSE of the TSFNN optimized by the IPSO are 0.004 and 5.6×10^{-6} , respectively, but the optimized ones by BP are 0.006

and 9.45×10^{-6} . It is obvious that the output accuracy of the TSFNN optimized by the IPSO is higher than that by BP.

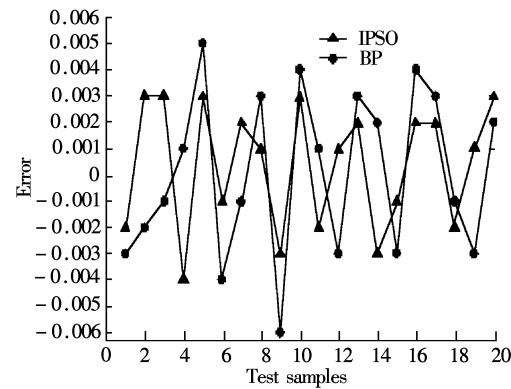


Fig. 5 Error distribution of test samples

3.4 Experiments in vehicle

Considering the degree of fatigue, the output value Y of the TSFNN is divided into four grades: no fatigue (0 to 0.3), a little fatigue (0.3 to 0.5), moderate fatigue (0.5 to 0.7), serious fatigue (0.7 to 1.0). Experiments on vehicles are carried out with the proposed algorithm in this paper. The detection system can judge whether the driver is fatigued and an alarm device can also issue a strong or a weak signal to remind the driver to pay attention to safety in terms of the grade of Y . The proposed comprehensive detection algorithm combining the direct and indirect fatigue features is compared with the PERCLOS and the algorithm fusing facial fatigue features^[5], and the performance of three methods is shown in Tab. 4. As seen from Tab. 4, the proposed algorithm has the highest correct rate, the lowest false-alarm rate and missing rate.

Tab. 4 Performance comparison of three methods %

Detection methods	Correct rate	Missing rate	False-alarm rate
PERCLOS	85.3	14.7	9.2
Ref. [5]	88.6	11.4	8.7
Proposed algorithm	94.5	4.5	4.2

4 Conclusions

1) A new driving fatigue fusion detection algorithm based on TSFNN evolved by SC and PSO is proposed, which considers two direct driver’s facial features and one indirect vehicle behavior feature. Compared with the traditional methods, the presented algorithm can obviously decrease the false-alarm rate and the missing rate, and remarkably increases the correct rate of driving fatigue detection.

2) In the structure, identification and parameter training of TSFNN, SC and IPSO are introduced, which makes TSFNN have fast convergence speed and high recognition accuracy.

References

[1] Xu C, Zheng Y, Wang Z F. Efficient eye states detection in real-time for drowsy driving monitoring system [C]// *Proceedings of the 2008 IEEE International Conference on Information and Automation*. Zhangjiajie, China, 2008: 170 – 174.

- [2] Fan X, Yin B C, Sun Y F. Yawning detection for monitoring driver fatigue [C]//*Proceedings of the Sixth International Conference on Machine Learning and Cybernetics*. Hong Kong, China, 2007: 664 – 668.
- [3] Lee D, Oh S, Heo S, et al. Drowsy driving detection based on the driver's head movement using infrared sensors[C]//*Proceedings of the 2nd International Symposium on Universal Communication*. Osaka, Japan, 2008: 231 – 236.
- [4] Koh A, Jones G R, Spencer J W, et al. Chromatic analysis of signals from a driver fatigue monitoring unit [J]. *Measurement Science and Technology*, 2007, **18**(3): 747 – 754.
- [5] Narole N G, Bajaj P R. A neuro-genetic system design for monitoring driver's fatigue: a design approach[C]//*Proceedings of the First International Conference on Emerging Trends in Engineering and Technology*. Maharashtra, India, 2008: 711 – 714.
- [6] Bergasa L M, Nuevo J, Sotelo M A, et al. Real-time system for monitoring driver vigilance [J]. *IEEE Transactions on Intelligent Transportation Systems*, 2006, **7**(1): 63 – 77.
- [7] Li Xu, Zhang Weigong. Research on lane departure warning system based on machine vision [J]. *Chinese Journal of Scientific Instruments*, 2008, **29**(7): 1554 – 1558. (in Chinese)
- [8] Li Xu, Zhang Weigong, Bian Xiaodong. Research on detection of lane based on machine vision [J]. *Journal of South-east University: English Edition*, 2004, **20**(2): 176 – 180.
- [9] Wong W K, Zeng X H, Au W M R. A decision support tool for apparel coordination through integrating the knowledge-based attribute evaluation expert system and the T-S fuzzy neural network [J]. *Expert Systems with Applications*, 2009, **36**(2): 2377 – 2390.
- [10] Han M, Sun Y N, Fan Y N. An improved fuzzy neural network based on T-S model [J]. *Expert Systems with Applications*, 2008, **34**(4): 2905 – 2920.
- [11] Chen Ruqing. A novel PSO based on-line fault diagnosis approach for power electronic system [J]. *Proceedings of the CSEE*, 2008, **28**(24): 70 – 74. (in Chinese)
- [12] Mahfouf M, Chen M, Linkens D A. Design of heat-treated alloy steels using intelligent multi-objective optimization [J]. *ISIJ International*, 2005, **45**(5): 694 – 699.
- [13] Zhang Dingxue, Guan Zhihong, Liu Xinzhi. Adaptive particle swarm optimization algorithm with dynamically changing inertia weight [J]. *Control and Decision*, 2008, **23**(11): 1253 – 1257. (in Chinese)
- [14] Clerc M, Kennedy J. The particle swarm—explosion, stability, and convergence in a multidimensional complex space [J]. *IEEE Transactions on Evolutionary Computation*, 2002, **6**(1): 58 – 73.
- [15] Drummond S P A, Bischoff-Grethe A, Dinges D F, et al. The neural basis of the psychomotor vigilance task [J]. *Sleep*, 2005, **28**(9): 1059 – 1068.

减法聚类和粒子群优化 TS 模糊神经网络的驾驶疲劳融合检测

孙 伟 张为公 李 旭 陈 刚

(东南大学仪器科学与工程学院, 南京 210096)

摘要:为提高基于单一特征检测算法的准确率和可靠性,提出基于多个特征的驾驶疲劳融合检测算法.从直接反映驾驶员疲劳的2个面部特征和间接反映疲劳的1个车辆行为特征2个方面对驾驶疲劳进行综合检测.该算法运用TS模糊神经网络来识别驾驶疲劳,采用减法聚类对网络进行结构辨识,确定模糊规则的条数及相关参数的初始值,并改进了粒子群优化算法对网络进行训练.仿真和实车实验表明,该算法不仅能有效改善TS模糊神经网络的收敛速度和识别精度,而且能提高驾驶疲劳的检测正确率.

关键词:驾驶疲劳;融合检测;粒子群优化;减法聚类

中图分类号:U495