Vehicle Driving Behavior Recognition and Optimization Strategies Based on Cloud Computing and SSA-BP Algorithm

Xing YANG[1]*, Ke XIANG[2], Shan YUAN[3], Jilan HUANG[1]

[1] Geely University of China, No. 123 Chengjian Avenue, East New District, Chengdu, Sichuan, 610000, China
rryangxing@163.com (**Corresponding author*), hlhuanglan@163.com

[2] Sichuan Post and Telecommunication College, No. 536 Jingkang Road, Jinjiang District, Chengdu, Sichuan, 610000, China
xiangke@sptc.edu.cn

[3] Sichuan Changjiang Vocational College, No. 828 Chengluo Road, Longquanyi District, Chengdu, Sichuan, 610000, China
yuanshan0715@163.com

Abstract: As the number of cars increases annually, driving behavior analysis has become a direction worthy of in-depth research in the field of public transportation. Traditional vehicle brakes generate torque during non-braking high-speed rotation, which has a negative impact on transmission efficiency. In this paper, the sparrow search algorithm was combined with the backpropagation neural network algorithm in the context of collecting and analysing multi-source data during vehicle driving, and the backpropagation neural network algorithm was improved by using the sparrow search algorithm's global search ability. A driving behavior recognition model was built in order to accurately distinguish various driving behaviors and evaluate their impact on fuel consumption and driving safety. The experimental outcomes showed that the vehicle driving behavior recognition method based on cloud computing and the backpropagation neural network algorithm combined with the sparrow search algorithm featured a high driving behavior recognition efficiency, and the research method featured a small error in the identification of vehicle driving behavior. The prior accuracy of the driving behavior-based fuel consumption evaluation model was as high as 89.216%, and the regression fit for the employed model after training reached around 98%. To sum up, the obtained results not only allow the implementation of new ideas and methods for the intelligent vehicle driving and traffic safety management, but they also provide an important theoretical support and practical guidance for the design and optimization of intelligent transportation systems.

Keywords: Sparrow search algorithm, BP neural network, Cloud computing, Vehicle driving behavior, Cloud platform.

1. Introduction

With the quick growth of intelligent transportation systems, vehicle driving behavior (DB) recognition and optimization strategies have become important research directions to ensure driving safety and improve traffic efficiency (Ali et al., 2023). Cloud computing platforms provide powerful computing power and scalable storage resources for data processing and model training, ensuring real-time and efficient research (Okita et al., 2022; Thieme et al., 2020). Traditional DB recognition methods are often limited by data processing capabilities and the efficiency of model optimization algorithms, making it difficult to achieve accurate recognition of complex DBs (Fan et al., 2021). The sparrow search algorithm (SSA) can realize global search in the search space and achieve global optimization through sparrow group search and position adjustment (Ou et al., 2023; Zhang et al., 2021). When the SSA is used alone, there is a problem of insufficient local search ability, and it is mainly suitable for continuous optimization, lacking universality (Wang, 2021). In this paper, the SSA was combined with the backpropagation (BP) algorithm (SSA-BP), and the SSA-BP is a method that combines heuristic search and neural networks. It has the advantages of global optimization and precise modeling, providing a new solution for DB recognition. Research uses cloud computing platforms and the SSA-BP algorithm to build a DB recognition model, accurately distinguishing different DBs from fuel consumption and driving safety evaluations.

The remainder of this paper is as follows. Section 2 introduces the research context and summarizes the related research. Section 3 illustrates the specific methods and cloud platform construction of the BP DB evaluation model combined with SSA. Section 4 validates the SSA-BP DB evaluation model and experimentally validates the training results of the BP algorithm. Finally, Section 5 concludes this paper.

2. Literature Review

The SSA simulates the swarm intelligence behavior exhibited by sparrows when foraging and avoiding natural enemies, achieving efficient global

https://doi.org/10.24846/v33i3y202402

exploration of complex search spaces. It exhibits strong optimization performance when facing various complex problems, and has broad application prospects in multiple function optimization and parameter optimization related fields.

Fan et al. (2023) put forward an SSA-based Elman neural network to forecast the energy consumption changes and distribution trends. The findings showed that this study analysed the effects of energy efficiency management and peak shaving measures, proposed reasonable control requirements and assumptions, and studied the operability of enterprise energy saving measures. To improve the prediction accuracy for rubber fatigue life, Wang & Liu (2023) comprehensively considered the strain rate effect and proposed a prediction method based on the improved Sparrow Search Algorithm (SSA). The modified SSA algorithm was used to optimize various hyperparameters in the support vector machine model. The results showed that in comparison with other search methods or models, this research method featured a better stability and prediction accuracy. Zhu and Yousefi (2021) proposed a new optimization algorithm for the adaptive sparrow search algorithm (ASSA) to identify the optimal model parameters. The results indicated that the proposed ASSA was the most efficient compared to other algorithms. Based on SSA, Zhang and Ding (2021) introduced a stochastic configuration network based on chaotic sparrow search algorithm (CSSA-SCN). First, to improve SSA's global optimization capabilities, an algorithm was developed that primarily used logical mapping, adaptive parameters, and mutational operators. Second, CSSA was applied to automatically provide better parameters for SCN, since the effects of SCN were related to the regularization parameter r and the ratio factor λ between weight and bias. In contrast with SCN and other comparable algorithms, the experimental results showed that CSSA-SCN was feasible and effective. Li et al. (2022) proposed a multi-scale knowledge sensing transformer and constructed a knowledge-guided multi-scale feature alignment framework. First, a knowledge-aware Transformer (KAT) was designed for the interaction between semantic knowledge and visual features. Second, in order to understand knowledge-guided sample differences, a knowledge-guided alignment loss was proposed to facilitate the separation of identity-related and identity-independent features. Experiments showed that the method captured knowledge-guided visual consistency features at different scales and was up to date on three widely used vehicle re-recognition benchmarks.

The emergence of in-vehicle network systems and the most advanced sensors and communication technologies have provided convenience for collecting a large amount of data about vehicles and drivers. Elassad et al. (2020) provided excellent solutions for DB analysis through machine learning techniques. Wu et al. (2021) conducted a qualitative analysis on the torque characteristics of wet clutches, lubricating oil flow rate, meshing force and its rate of change, lubricating oil temperature, and friction surface temperature. Sekiguchi et al. (2023) found two different slopes in the relationship among fatigue crack growth rate, strain energy release rate, and average load parameters, depending on loading conditions. Zhou et al. (2021) demonstrated the relationship between driving environment, systems, and psychological models, and determined the response to intervention requests based on knowledge-based learning. Guiding drivers in typical scenarios helped improve the success rate of vehicle control intervention, increasing it from 55% to 95%. Niu et al. (2021) built a classification framework for unsafe DB of truck drivers. The classification framework contains 6 primary input dimensions and 51 secondary input indicators. As an output, 9 unsafe DBs were identified. The findings indicated that the model's predictive effects varied with DBs, which was due to the different formation mechanisms of different DBs.

In summary, vehicle DB recognition and optimization strategies have received widespread attention in the field of intelligent transportation. The introduction of cloud computing technology provides strong support for large-scale data processing. As an optimization algorithm that combines SSA and BP neural networks, the innovation of SSA-BP algorithm lies not only in traditional DB recognition, but also in the in-depth exploration of its application breadth, which comprises multiple aspects of intelligent transportation.

3. Research Methodology

This study uses the Normalized Cut (NCut) as the segmentation criterion for spectral clustering algorithms, providing a more accurate and reliable method for DB analysis. The SSA-BP algorithm is improved by utilizing the global optimization ability of SSA to build a DB recognition model.

3.1 Analysis and Platform Deployment of Vehicle Driving Behavior based on Cloud Computing

This paper focused on the method of feature parameter extraction and selected DB as the training feature vector for clustering analysis. Clustering analysis can form clusters representing different DBs. After further processing of these clusters, labeled classification training samples and corresponding categories are obtained. Subsequently, a classifier is designed to evaluate the classifier online using new vehicle data samples to determine which category the DB belongs to. In the case of labeled training samples, a supervised learning classifier, that is, a neural network classifier, is used. In the case of unlabeled training samples, an unsupervised learning classifier, that is, a hierarchical clustering classifier, is used. The entire process of DB analysis is shown in Figure 1.

The technical solution illustrated in Figure 1 comprises three main core technologies, the first of which is the feature parameter extraction module, whose core task is to extract key feature parameters from preprocessed data as training samples. The DB clustering analysis module uses big data analysis algorithms to cluster DB data, adding label columns to the original data samples to identify the DB category to which each sample belongs. Finally, the classifier module is the core output part of the DB analysis technical solution. The Normalized Cut (NCut) criterion can not only effectively evaluate the similarity between samples within a cluster, but also accurately measure the degree of dissimilarity between samples in different clusters (Ismail et al., 2022; Wang et al., 2020). In comparison with the Minimum Cut criterion and the Ratio Cut criterion, NCut can overcome its shortcomings simultaneously. Therefore, in spectral clustering algorithms, the use of the NCut criterion as the segmentation criterion is studied to ensure the accuracy and stability of the spectral clustering algorithm. The process related to the spectral clustering algorithm is shown in Figure 2.

In Figure 2, based on the spectral segmentation criterion, the clustering problem is transformed into an issue of finding the minimum value of the objective function, and the solution of the spectral clustering algorithm is obtained. A graph is constructed based on a given set of data samples, and the NCut algorithm is used for spectral

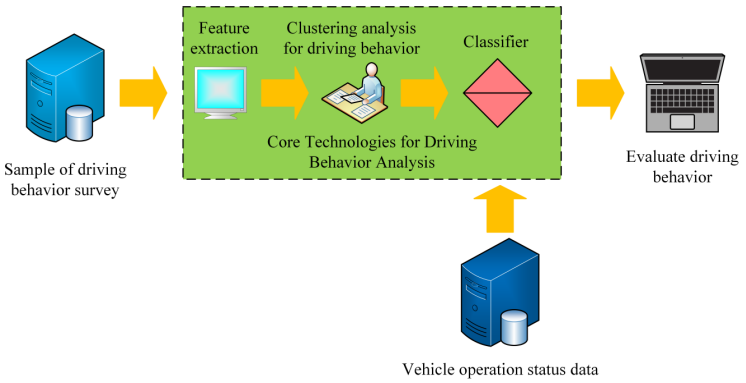


Figure 1. Technical schematic diagram for driving behavior analysis

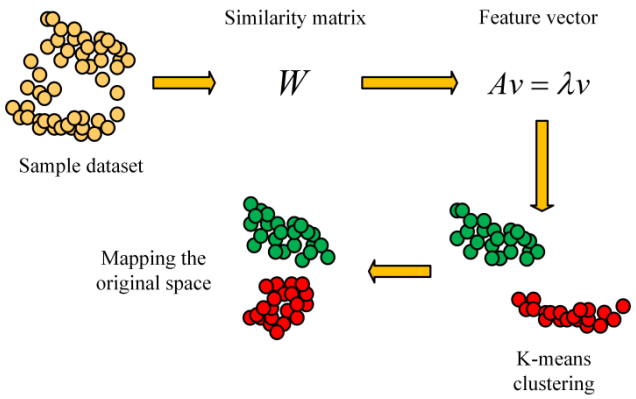


Figure 2. Genealogical algorithm framework diagram

segmentation. This process aims to optimize the objective function and divide the data samples into different categories or clusters, thereby achieving the purpose of clustering analysis. In terms of building a cloud computing platform, the first step is to build a IaaS infrastructure environment based on OpenStack, virtualize cloud hosts, install Ubuntu systems for each cloud host, and then build a Hadoop distributed cluster in the cloud host to provide Hadoop distributed file system (HDFS) file storage for Spark, and then build a Spark distributed cluster (Fu et al., 2023; Kim & Kim, 2019). Finally, a software platform development environment for OpenStack-Hadoop-Spark is implemented. The deployment of the OpenStack cloud platform is shown in Figure 3.

The OpenStack cloud platform deployment illustrated in Figure 3 mainly includes three nodes. The core components of the OpenStack Compute Node mainly include Nova Compute, Neutron Plugin ML2 and Neutron Openvswitch Agent (Ramamoorthi & Ramasamy, 2023; Wang & Fang, 2023). The core components of an OpenStack Network Node include Neutron Plugin ML2, Neutron L3 Agent, Neutron DHCP Agent, and Neutron Openvswitch Agent, which are used to implement gateway and routing functionality. The core components of OpenStack Controller Node include Keystone, Glance, Nova Conductor, Nova Scheduler, Nova API, Neutron Server, Horizon, Cinder, and Swift.

3.2 DB and Evaluation Model based on SSA-BP Algorithm

BP is a common feedforward network that achieves the goal of training the BP network by BP of errors (Hu et al., 2022). As it is shown in Figure 3, it mainly consists of three parts: input, hidden, and output layers. The specific process of BP first initializes the weights and thresholds, and the output value a_i^0 of input layer node i is expressed in equation (1):

$$a_i^0 = X_i \tag{1}$$

In equation (1), x_i represents the output value of input layer node i, and the input and output values of node j in the hidden layer are expressed in equation (2):

$$\begin{cases} I_j = \sum_{i=1}^{p} w_{ij} a_i^0 + \theta_j \\ a_i^1 = f(I_j) = \dfrac{1}{1+e^{-I_j}} \end{cases} \tag{2}$$

In equation (2), w_{ij} denotes the connection weight between input layer node i and hidden layer node j, and θ_j is the threshold on node j. The input I_k and output a_i^2 of node k in the output layer are shown in equation (3):

$$\begin{cases} I_k = \sum_{j=1}^{p} w_{jk} a_i^1 + \theta_k \\ a_i^2 = f(I_k) = \dfrac{1}{1+e^{-I_k}} \end{cases} \tag{3}$$

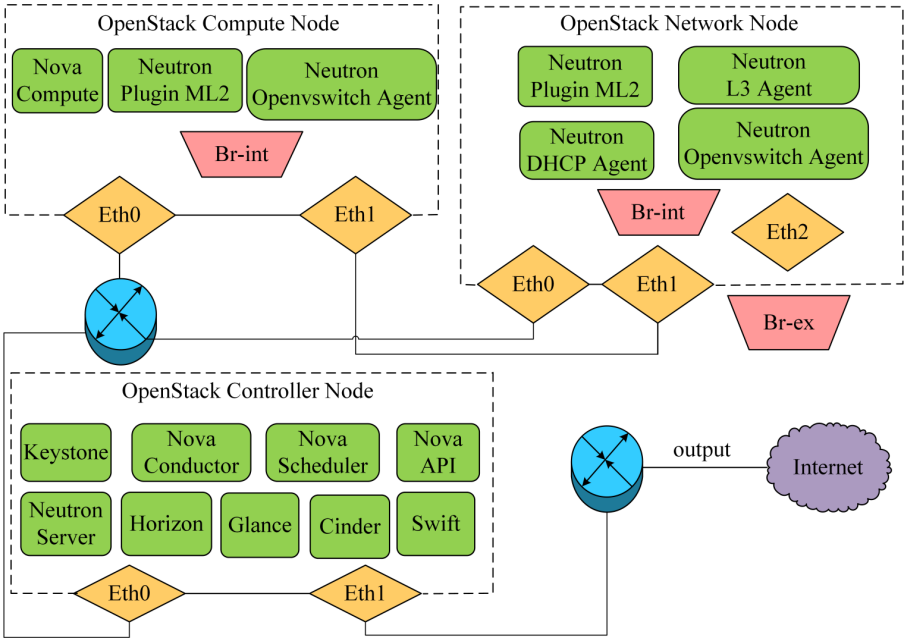


Figure 3. OpenStack cloud platform deployment diagram

In equation (3), w_{jk} indicates the connection weight between node j and node k, and θ_k is the threshold on node k. There are m outputs in each sample, and the training error for N samples is expressed in equation (4):

$$E = \sum_{i=1}^{N} E_i = \frac{1}{2} \sum_{i=1}^{m} \sum_{k=1}^{m} (d_{ik} - y_{ik})^2 \quad (4)$$

In equation (4), E_i denotes the training error of the i-th sample. d_{ik} is the expected output of training sample output value. y_{ik} represents the neural network predicted value for training sample output value. By training neural networks, the amount of hidden layer nodes that can minimize prediction error is found. The determination of the number of nodes is expressed in equation (5):

$$p = \sqrt{n+m} + a \quad (5)$$

In equation (5), p denotes the number of the nodes. n represents the number of input layer nodes. m represents the number of output layer nodes, and a is a constant up to 10. It is necessary to process the sample data using a normalization function and map it to the [-1,1] interval. The normalization function is expressed in equation (6):

$$y = \frac{(y_{\max} - y_{\min})(x - x_{\min})}{x_{\max} - x_{\min}} + y_{\min} \quad (6)$$

In equation (6), x represents the pre-normalization data. y represents the post-normalization data. y_{max} defaults to 1, and y_{min} defaults to -1. In this study, two error indicators were selected, namely the Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE), to prove the predictive ability of the proposed model. RMSE reflects the degree of error between the predicted and the actual values. The lower the RMSE value, the higher the accuracy of the model. MAE is the average absolute error, and the lower the value of MAE, the better the predictive effectiveness of the employed model. The formula for calculating the average absolute error is expressed in equation (7):

$$MAE = \frac{1}{m} \sum_{i=1}^{m} |y_i - x_i| \quad (7)$$

In equation (7), m refers to the number of samples. x_i and y_i denote the predicted and the actual values, respectively. The formula for calculating the root mean square error is expressed in equation (8):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{m} (x_i - y_i)^2}{m}} \quad (8)$$

The SSA is proposed as it imitates the collective behavior of sparrows in foraging and anti-predation. This algorithm has few parameters, a strong search ability, and a fast performance. The SSA process first sets the sparrow population matrix, as it is shown in equation (9):

$$X = \begin{bmatrix} x_{11}, x_{12}, ..., x_{1d} \\ x_{21}, x_{22}, ..., x_{2d} \\ ... \\ x_{n1}, x_{n2}, ..., x_{nd} \end{bmatrix} \quad (9)$$

In equation (9), n represents the population size. d represents the population dimension. The population dimension is expressed in equation (10):

$$d = EF + FG + F + G \quad (10)$$

In equation (10), E, F, and G represent the amount of neurons in the input, hidden, and output layers. The sparrow population fitness value is expressed in equation (11):

$$F_X = \begin{bmatrix} f([x_{11}, x_{12}, ..., x_{1d}]) \\ f([x_{21}, x_{22}, ..., x_{2d}]) \\ ... \\ f([x_{n1}, x_{n2}, ..., x_{nd}]) \end{bmatrix} \quad (11)$$

In equation (11), f denotes the fitness value. The discoverer's position is updated. In order to obtain more food, sparrows will change their position based on individual energy changes. The position update is expressed in equation (12):

$$X_{ij}^{t+1} = \begin{cases} X_{ij}^{t} * \exp(-\frac{i}{\propto * N}), R_2 < ST \\ X_{IJ}^{t} + D * L, R_2 \geq ST \end{cases} \quad (12)$$

In equation (12), X_{ij}^{t+1} denotes the t-th dimensional position of the t-th generation. N refers to the maximum number of iterations. L refers to a 1xd matrix with all elements being 1. R_2 is the warning value, in the interval [0,1]. ST represents the safe value, in the range [0.5,1]. $R_2 < ST$ denotes that the environment in which sparrows are searching for food is safe and not dangerous at the current time. $R_2 \geq ST$ points out that some individuals in the sparrow population have already felt the threat of predators, and that

the environment in which sparrows are searching for food is dangerous. The position update is as shown in equation (13):

$$X_{ij}^{t+1}=\begin{cases} D*\exp(\frac{X_{worst}^{t}-X_{ij}^{t}}{i_2}), i>\frac{n}{2} \\ X_{p}^{t+1}+\left|X_{ij}^{t}-X_{p}^{t+1}\right|*A^{+}*L, i\le\frac{n}{2} \end{cases} \quad (13)$$

In equation (13), X_{worst}^{t} denotes the global worst position at the *t*-th iteration. X_{p}^{t+1} refers to the best position of the discoverer at the *t*+1-th iteration. A^{+} is the 1xd matrix, with each element being 1 or -1. $i>\frac{n}{2}$ means that the *i*-th addition with low energy values did not obtain food, indicating poor fitness. This sparrow individual needs to go to areas with richer food resources to obtain food. In addition, participants can forage near the current optimal sparrow individual. The alert position update is expressed in equation (14):

$$X_{ij}^{t+1}=\begin{cases} X_{ij}^{t}+K*\left(\frac{\left|X_{ij}^{t}-X_{worst}^{t}\right|}{(f_i-f_g)+\varepsilon}\right), f_i=f_g \\ X_{best}^{t}+\beta*\left|X_{ij}^{t}-X_{worst}^{t}\right|, f_i\neq f_g \end{cases} \quad (14)$$

In equation (14), *K* indicates a random number in the interval [-1,1]. f_i indicates the individual fitness value. X_{best}^{t} denotes the global optimal position in the *t*-th iteration. β is the step size control coefficient and ε is a minimum constant to prevent the denominator from being set to 0. $f_i=f_g$ indicates that the current sparrow individual is at the center of the sparrow population and will randomly follow other individuals to cut down the risk of being preyed upon. $f_i\neq f_g$ indicates that the current sparrow individual is located at the periphery of the sparrow population and is susceptible to predation. For verifying the performance of SSA-BP neural network the correlation coefficient R^2 is used, which is an important indicator for evaluating model accuracy. It represents the percentage of the dependent variable *Y* in the total variation that the constructed model can explain, and it is expressed in equation (15):

$$R^2=1-\frac{SSR}{SST} \quad (15)$$

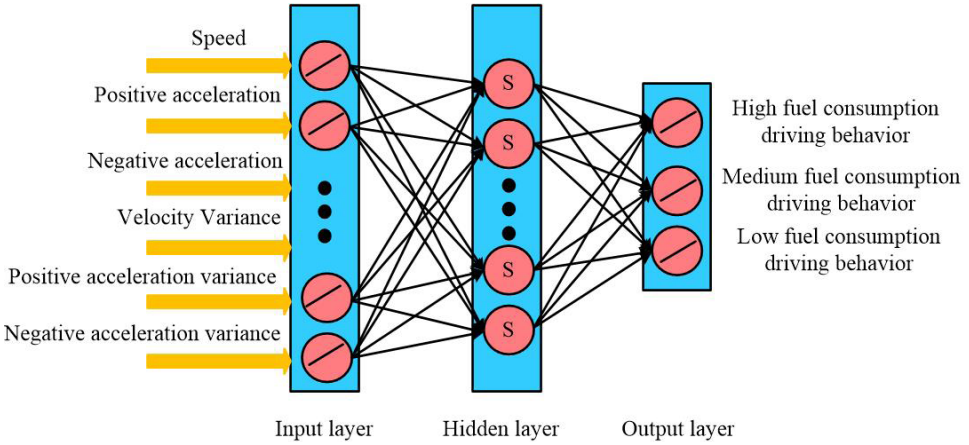
In equation (15), *SSR* is the sum of squared residuals. *SST* is the total sum of squares. To evaluate the relationship between DB and fuel consumption, a three-layer BP neural network model was utilized in this study. Features related to fuel consumption from the original feature data were extracted using the Pearson correlation coefficient, and a column was added as the oil category label in the dataset. The newly constructed feature vector was used as the input layer training vector. The BP neural network structure and the DB and fuel consumption evaluation process are shown in Figure 4.

Figure 4 depicts a BP-based classifier in order to establish a fuel consumption assessment model. In this model, the hidden layer adopts the widely used sigmoid function as the activation function. Due to their nonlinear characteristics, neural networks have a strong function approximation ability. The BP neural network is trained using the MATLAB toolbox, and the remaining driving data samples are utilized as the test set to prove the accuracy and generalization ability of the proposed model. The construction principle of the DB safety assessment model is similar to that of the DB fuel consumption assessment model. Similarly, a three-layer BP neural network is employed to cluster the DB data using spectral clustering algorithm, and the clustering results for the DB are obtained. The original sample data is used as training samples, and the corresponding mapping model is built. The BP neural network structure, and the DB and driving safety assessment process are shown in Figure 5.

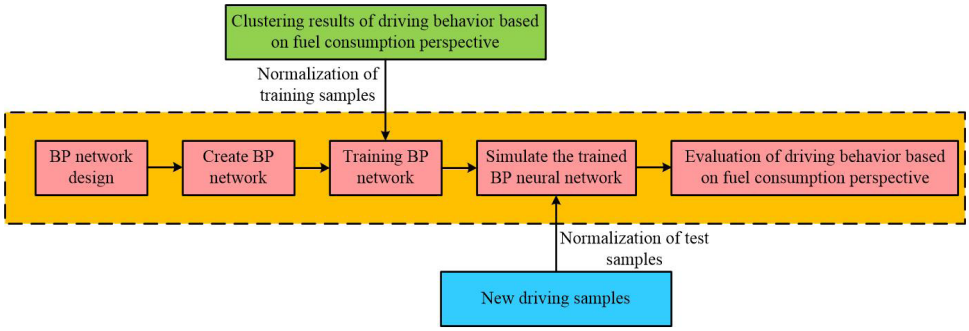
Figure 5 depicts a BP-based classifier in order to assess the DB safety for drivers. The spectral clustering algorithm is first used to cluster and analyze the DB data, thereby obtaining the classification results for DB (Kannout et al., 2023). Using these clustering results as training samples, the network learns the relationship between DB and safety through training. To prove the accuracy of the proposed model, the remaining driving data samples are applied as the test set to test the trained BP neural network.

4. Results and Discussion

Combining fuel consumption and safety assessment models, this study used MATLAB tools to analyze the influence of various DBs on fuel consumption and visualise the data samples, and proposed targeted optimization strategies.

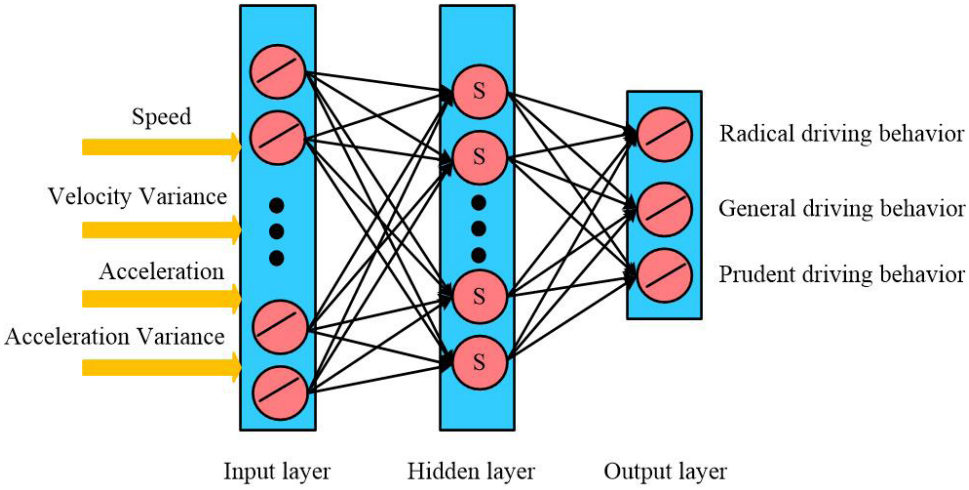


(a) SSA-BP neural network structure diagram

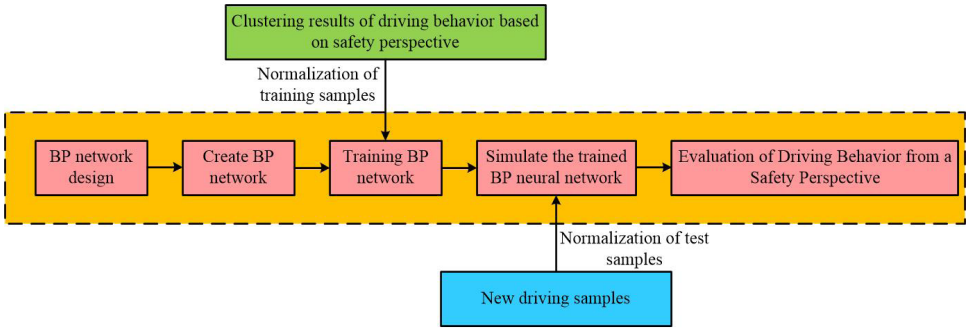


(b) Driving Behavior - Fuel Consumption Assessment Flowchart

Figure 4. BP neural network structure and the driving behavior and fuel consumption evaluation process



(a) SSA-BP neural network structure diagram

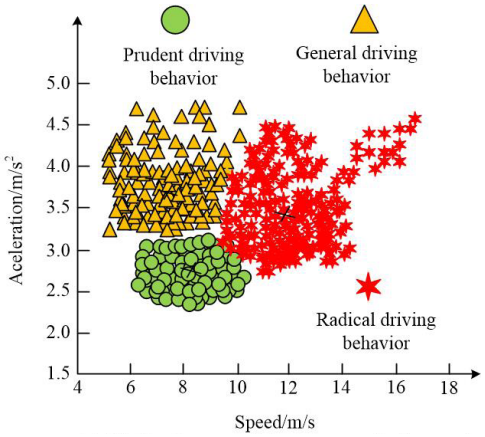


(b) Driving Behavior - Safety Assessment Process Diagram

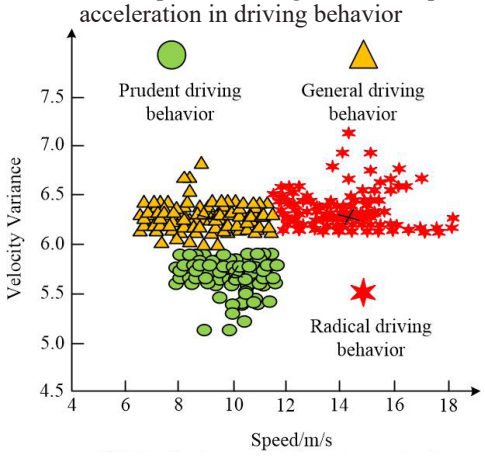
Figure 5. BP neural network structure and driving behavior and safety assessment process

4.1 Clustering Analysis for Driving Behavior Based on Safety Perspective

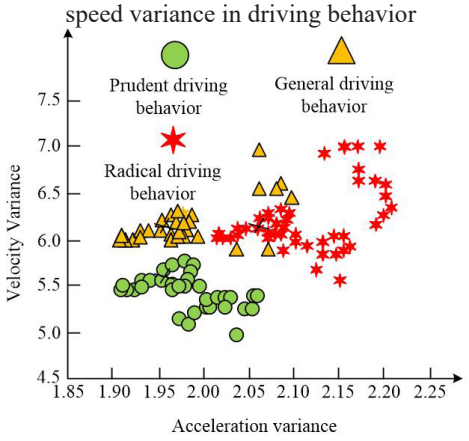
Based on the results obtained for the Spark distributed cluster parallel spectral clustering algorithm, the DB of drivers could be mainly divided into three categories. In this study MATLAB tools were employed to visualize sample data for each category. In the visualization process, a more intuitive display of the distribution of different DB categories is shown in Figure 6.



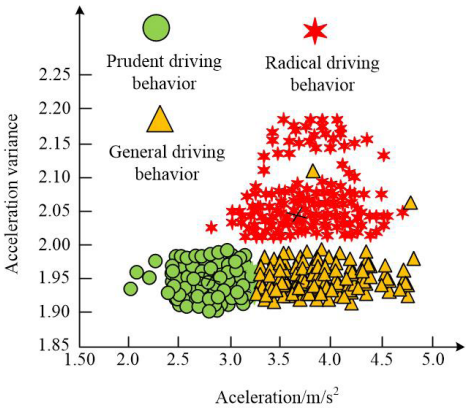
(a) Distribution map of clustering results for speed and acceleration in driving behavior



(b) Distribution map of clustering results for speed and speed variance in driving behavior



(c) Distribution map of clustering results for acceleration variance and speed variance in driving behavior



(d) Distribution map of clustering results for acceleration and acceleration variance in driving behavior

Figure 6. Cluster interval distribution for Spark cloud platform

In Figure 6, the clusters marked in red indicated a higher acceleration, which usually meant that when the speed exceeded 30 km/h, the driver's acceleration was also higher, and this DB was considered aggressive. By contrast, clusters marked in green displayed moderate acceleration and velocity, with smaller acceleration and velocity variances in comparison with yellow clusters, indicating a stable DB and pointing to cautious driving. The speed and acceleration of the yellow clusters were also relatively moderate, but their variance was slightly larger than that of the green clusters, so they were classified as moderate DB. Sample size statistics for the output results were obtained, as it is shown in Figure 7.

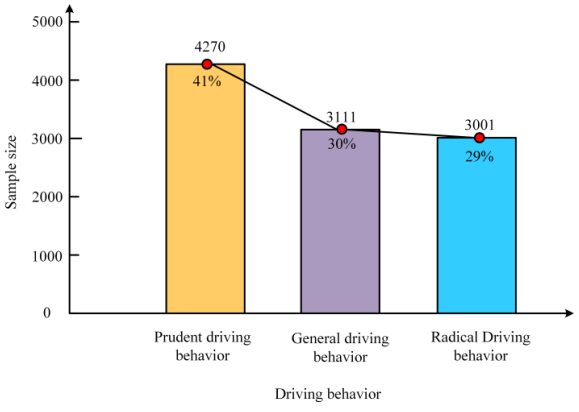
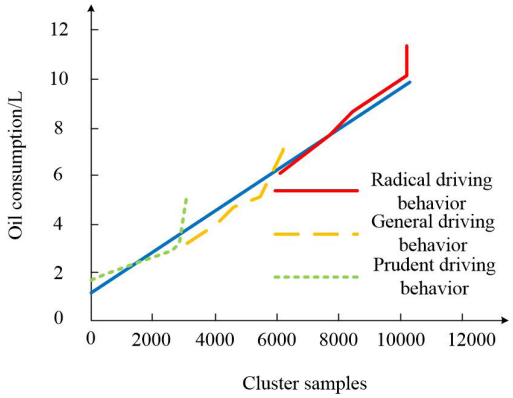


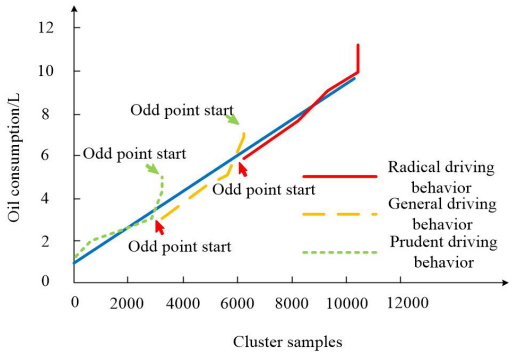
Figure 7. Sample distribution of driving behavior clusters

As it can be seen in Figure 7, after cluster analysis, a total of 4270 samples belonged to prudent DB, accounting for 41% of the total. 3001 samples belonged to radical DB, accounting for 29% of the total. There were 3111 samples

of general DB, accounting for 30% of the total. The analysed data provided a clear overview of the distribution of DB, which helped to further analyse and optimise DB. Cluster analysis was conducted on six parameters closely related to fuel consumption which were extracted through Pearson correlation coefficient using the Spark distributed cluster, to better understand the differences between different DBs and their relationship with fuel consumption. Sample data for each type of DB was visualised using MATLAB tools, as it is shown in Figure 8.



(a) Cluster distribution of driving behavior



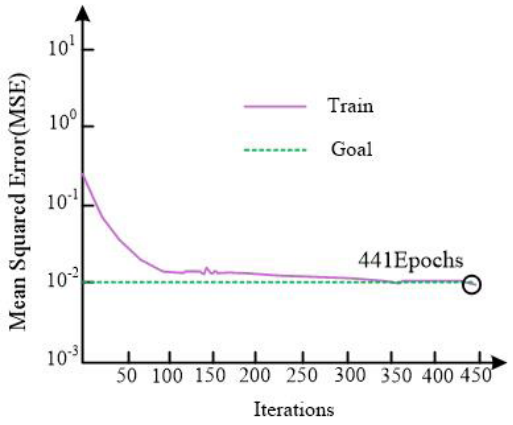
(b) Cluster distribution of driving behavior

Figure 8. Fuel consumption trend chart for different driving behavior clusters

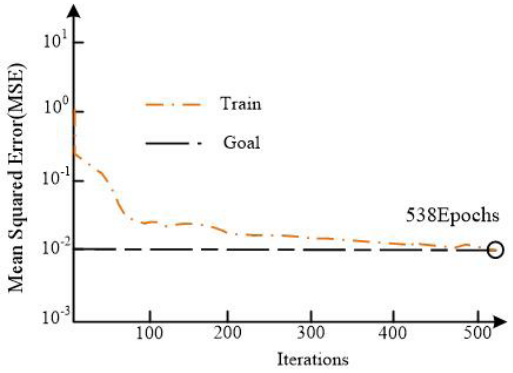
In Figure 8, the closer the color is to red, the greater the fuel consumption. After clustering analysis, DB was divided into three clusters, corresponding to high, medium, and low fuel consumption DBs. The cluster center and sample data exhibited obvious characteristics, which were related to high fuel consumption.

4.2 DB Analysis, Evaluation, and Optimization Strategies Based on SSA-BP Algorithm

A BP-based DB safety evaluation model was constructed, with new feature vectors as training samples and 2000 samples were retained as predictive samples for the validation and evaluation of the model's accuracy. The convergence curves for the DB safety and DB fuel consumption assessment errors are shown in Figure 9.



(a) Driving safety evaluation convergence curve



(b) Fuel consumption evaluation convergence curve

Figure 9. Convergence curves for driving safety and fuel consumption evaluation errors

As it can be seen in Figure 9(a), the DB safety assessment model successfully converged after 441 iterations, with a prior accuracy of 86.504%. During the validation process, the number of samples which were correctly predicted by the model reached 1730, while the number of incorrectly predicted samples was 270. These results fully demonstrated that the proposed BP neural network model could serve as an effective tool for DB safety assessment. As it can be seen in Figure 9(b), the DB fuel consumption evaluation model successfully converged after 538 iterations, with a prior accuracy of up to 89.216%. During the validation process, the number of samples which were correctly predicted by the model reached 1784, while the number of incorrectly predicted samples was 216. These results fully demonstrated that the proposed model could also serve as an effective

tool for evaluating DB fuel consumption. The SSA-BP regression fit is shown in Figure 10.

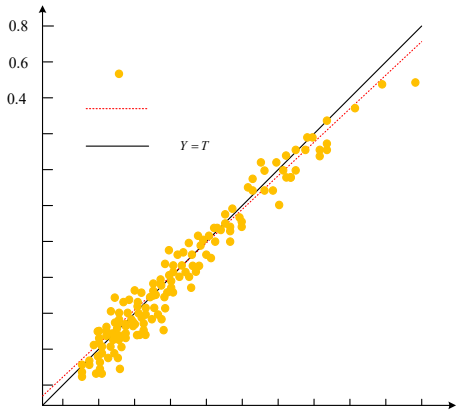


Figure 10. SSA-BP regression fit

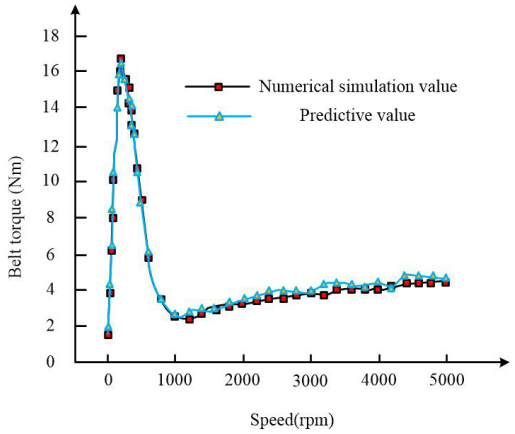
As it can be seen in Figure 10, the regression fit of the trained model reached around 98%, indicating a high regression accuracy. After calculation verification, the value of the fitting correlation coefficient R^2 of the SSA-BP model on the training set was as high as 0.993, and it also reached the value of 0.991 on the test set. These results fully demonstrated that the model had excellent fitting and prediction capabilities, and its prediction accuracy was also very high. This provided a solid foundation for the subsequent use of the model for DB analysis and evaluation. The specific prediction results of SSA-BP for the belt torque and belt torque error are shown in Figure 11.

In Figure 11, the overall trend of the predicted values and numerical simulation values was highly consistent, and the results of numerical agreement were almost ideal. Specifically, the average error was controlled at 0.0785Nm, and the maximum error did not exceed 1.4Nm. This indicated that the predicted results for the model had a high degree of overlap with the numerical simulation results. In summary, the employed neural network model could accurately reflect the relationship between torque and wet brakes under different operating conditions, demonstrating good simulation and prediction capabilities. To prove the advantages of the SSA-BP neural network model, multiple prediction models were selected in order to compare their effectiveness and to comprehensively evaluate the advantages and applicability of the SSA-BP model. The training outcomes for the linear regression model are included in Table 1.

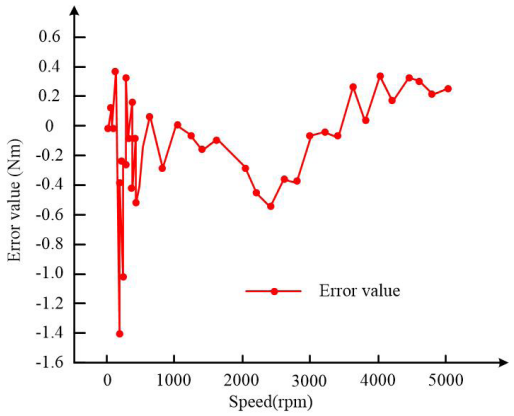
As it can be seen in Table 1, the employed linear regression model showed significant errors in the prediction process, with a maximum value exceeding 25Nm, equivalent to an error rate of over 50%. Especially in the low-speed range, this error was particularly evident. However, as the speed increased, the error gradually decreased, and when the speed reached 5000rpm, the error was almost zero, indicating that linear regression models had certain limitations when dealing with such complex relationships.

Table 1. Analysis of training results for the employed linear regression model

Speed/rpm	Numerical simulation/Nm	Model prediction value/Nm	Relative error
20	6.59862	19.79076	13.19214
200	35.81198	19.38032	16.43166
400	44.41275	19.05415	25.3586
600	42.39716	18.63537	23.76179
800	25.38251	18.24532	7.13719
1000	15.92605	17.85638	1.93033
1500	14.66761	17.00804	2.34044
2000	11.61611	15.93265	4.31654
2500	12.24546	14.99396	2.7485
3000	11.48592	14.14108	2.65517
3500	12.41518	13.09661	0.68143
4000	10.20604	12.17937	1.97334
4500	11.46941	11.0524	0.41702
5000	9.96277	10.26891	0.30614
Average relative error	/	/	7.375020714



(a) Prediction results based on SSA-BP model



(b) Line chart of SSA-BP prediction of torque error with belt arrangement

Figure 11. Prediction of belt torque and torque error for the SSA-BP algorithm model

5. Conclusion

As the number of cars increases year by year, the analysis of DB has become increasingly important in the field of public transportation. Traditional vehicle brakes generate power loss during non-braking high-speed rotation, which has a negative impact on transmission efficiency. To address this issue, multiple data sources were collected and analysed during vehicle operation. This paper innovatively combines the SSA with the BP algorithm, utilizing the best global search ability of the SSA to optimize the BP algorithm. Through this approach, a DB recognition model was built that can distinguish different DBs and evaluate their impact on fuel consumption and safety. The experimental findings showed that the prior accuracy of the DB-based fuel consumption evaluation model based on the SSA-BP algorithm was as high as 89.216%, and the regression fit of the model after training reached around 98%. However, this paper also features certain shortcomings, such as the failure to horizontally compare the SSA-BP algorithm with other algorithms and to explore the possibility of choosing more classification algorithms.

Acknowledgements

This work was supported by the 2023 Higher Education Scientific Research Planning Project of the China Association of Higher Education, “Theoretical and Practical Research on the Industry-Education Integration Community” (Project No. 23ZYJ0214).

REFERENCES

Ali, A. M., Moulik, B. & Söffker, D. (2023) Intelligent Real-Time Power Management of Multi-Source HEVs Based on Driving State Recognition and Offline Optimization. *IEEE Transactions on Intelligent Transportation Systems*. 24(1), 247–257. doi: 10.1109/TITS.2022.3215607.

Elassad, Z. E. A., Mousannif, H., Moatassime, H. A. & Karkouch, A. (2020) The application of machine learning techniques for driving behavior analysis: A conceptual framework and a systematic literature review. *Engineering Applications of Artificial Intelligence.* 87, 103312. doi: 10.1016/j.engappai.2019.103312.

Fan, X., Wang, F., Song, D., Lu, Y. & Liu, J. (2021) GazMon: Eye Gazing Enabled Driving Behavior Monitoring and Prediction. *IEEE Transactions on Mobile Computing*. 20(4), 1420-1433. doi: 10.1109/TMC.2019.2962764.

Fan, Y., Sui, T., Peng, K., Sang, Y. & Huang, F. (2023) Study on load monitoring and demand side management strategy based on Elman neural network optimized by sparrow search algorithm. *Circuit World*. 49(1), 56-66. doi: 10.1108/CW-07-2021-0199.

Fu, Z., He, M., Tang, Z. & Zhang, Y. (2023) Optimizing data locality by executor allocation in spark computing environment. *Computer Science and Information Systems*. 20(1), 491-512. doi: 10.1108/CW-07-2021-0199.

Hu, K., Wang, L., Li, W., Cao, S. & Shen, Y. (2022) Forecasting of solar radiation in photovoltaic power station based on ground-based cloud images and

BP neural network. *IET Generation, Transmission & Distribution*. 16(2), 333-350. doi: 10.1049/gtd2.12309.

Ismail, F. F., El-Aasser, M. A. & Gad, N. H. (2022) A Parasitic Hat for Microstrip Antenna Design Based on Defected Structures for Multiband Applications. *Applied Computational Electromagnetics Society Journal*. 37(5), 568-575. doi: 10.13052/2022.ACES.J.370506.

Kannout, E., Grodzki, M. & Grzegorowski, M. (2023) Towards addressing item cold-start problem in collaborative filtering by embedding agglomerative clustering and FP-growth into the recommendation system. *Computer Science and Information Systems*. 20(4), 1343-1366. doi: 10.2298/CSIS221116052K.

Kim, B. S. & Kim, T. G. (2019) Cooperation of Simulation and Data Model for Performance Analysis of Complex Systems. *International Journal of Simulation Modelling*. 18(4), 608-619. doi: 10.2507/IJSIMM 18(4)491.

Li, H., Li, C., Zheng, A., Tang, J. & Luo, B. (2022) MsKAT: Multi-Scale Knowledge-Aware Transformer for Vehicle Re-Identification. *IEEE Transactions on Intelligent Transportation Systems*. 23(10), 19557-19568. doi: 10.1109/TITS.2022.3166463.

Niu, Y., Li, Z. & Fan, Y. (2021) Analysis of truck drivers' unsafe driving behaviors using four machine learning methods. *International Journal of Industrial Ergonomics*. 86(8), 103192. doi: 10.1016/j.ergon.2021.103192.

Okita, N. T., Camargo, A. W., Ribeiro, J., Coimbra, T. A., Benedicto, C. & Faccipieri, J. H. (2022). High-performance computing strategies for seismic-imaging software on the cluster and cloud-computing environments. *Geophysical Prospecting*. 70(1), 57-78. doi: 10.1111/1365-2478.13158.

Ou, Y., Yu, L. & Yan, A. (2023) An Improved Sparrow Search Algorithm for Location Optimization of Logistics Distribution Centers. *Journal of Circuits, Systems and Computers*. 32(9). doi: 10.1142/S0218126623501505.

Ramamoorthi, R. & Ramasamy, A. (2023) Block Chain Technology Assisted Privacy Preserving Resource Allocation Scheme for Internet of Things Based Cloud Computing. [*Technical Gazette*]. 30(6), 1943-1950. doi: 10.17559/TV-20230404000503.

Sekiguchi, Y., Houjou, K., Shimamoto, K. & Sato, C. (2023) Two-parameter analysis of fatigue crack growth behavior in structural acrylic adhesive joints. *Fatigue & Fracture of Engineering Materials and Structures*. 46(3), 909-923. doi: 10.1111/ffe.13908.

Thieme, A., Yadav, S., Oddo, P. C., Fitz, J. M. & Hively, W. D. (2020) Using NASA Earth observations and Google Earth Engine to map winter cover crop conservation performance in the Chesapeake Bay watershed. *Remote Sensing of Environment*. 248(1), 111943-111955. doi: 10.1016/j.rse.2020.111943.

Wang, F., Zhao, C., Liu, J. & Huang, H. (2020) A Variational Image Segmentation Model based on Normalized Cut with Adaptive Similarity and Spatial Regularization. *SIAM Journal on Imaging Sciences*, 13(2), 651-684. doi: 10.1137/18M1192366.

Wang, M. (2021) Manufacturing capacity evaluation of smart job-shop based on neural network. *International Journal of Simulation Modelling*. 20(4), 778-789. doi: 10.2507/IJSIMM20-4-C019.

Wang, Y. & Fang, R. (2023) An Approach for Fast Fault Detection in Virtual Network. [*Technical Gazette*]. 30(4), 1146-1151. doi: 10.17559/TV-20230207000330.

Wang, X. & Liu, J. (2023) Intelligent prediction of fatigue life of natural rubber considering strain ratio effect. *Fatigue & Fracture of Engineering Materials and Structures*. 46(5), 1687-1703. doi: 10.1111/ffe.13952.

Wu, B., Qin, D., Hu, J., Wang, X., Wang, Y. & Lv, H. (2021) Analysis of influencing factors and changing laws on friction behavior of wet clutch. *Tribology International*. 162(1), 107125–107135. doi: 10.1016/j.triboint.2021.107125.

Zhang, C. & Ding, S. (2021) A stochastic configuration network based on chaotic sparrow search algorithm. *Knowledge-Based Systems*. 220(10), 106924-106943. doi: 10.1016/j.knosys.2021.106924.

Zhang, J., Xia, K., He, Z., Yin, Z. & Wang, S. (2021) Semi-Supervised Ensemble Classifier with Improved Sparrow Search Algorithm and Its Application in Pulmonary Nodule Detection. *Mathematical Problems in Engineering*. 2021, 1-18. doi: 10.1155/2021/6622935.

Zhou, H., Itoh, M. & Kitazaki, S. (2021) How Does Explanation-Based Knowledge Influence Driver Take-Over in Conditional Driving Automation? *IEEE Transactions on Human-Machine Systems*. 51(3), 188-197. doi: 10.1109/THMS.2021.3051342.

Zhu, Y. & Yousefi, N. (2021) Optimal parameter identification of PEMFC stacks using Adaptive Sparrow Search Algorithm. *International Journal of Hydrogen Energy*. 46(14), 9541-9552. doi: 10.1016/j.ijhydene.2020.12.107.