



OPEN

Construction and optimization of representative actual driving cycles based on the improved autoencoder

Zhichao Zhao^{1,2}, Xilei Sun^{3✉}, Xun Wang³, Yi Wang^{1,2}, Jianqin Fu^{3,4✉} & Jingping Liu^{3,4}

In this study, much work has been performed to accurately and efficiently develop representative actual driving cycles. Electric vehicle road tests were conducted and the associated data were gathered based on the manual driving method, and the Changsha Driving Cycle Construction (CS-DCC) method was proposed to achieve systematical construction of a representative driving cycle from the original data. The results show that the refined data exhibit greater stability and a smoother pattern in contrast to the original data after noise reduction by five-scale wavelet analysis. The Gaussian Kernel Principal Component Analysis (KPCA) algorithm is chosen to reduce the dimensionality of the characteristic matrix, and the number of principal components is selected as 5 with a cumulative contribution rate of 85.99%. The average error of the characteristic parameters between the optimized drive cycle and the total data is further reduced from 13.6 to 6.1%, with a reduction ratio of 55.1%. Meanwhile, the constructed driving cycle has prominent local characteristics compared with four standard driving cycles, demonstrating the necessity of constructing an actual driving cycle that reflects localized driving patterns. The findings present a powerful application of artificial intelligence in advancing engineering technologies.

Abbreviations

AI	Artificial intelligence
BEVs	Battery electric vehicles
Birch	Balanced iterative reducing and clustering using hierarchies
CDCS	Charge depletion and charge sustain
CLTC	China light-duty vehicle test cycle
CS-DCC	Changsha driving cycle construction
DP	Dynamic programming
J10-15	Japanese 10–15 mode
KPCA	Kernel principal component analysis
MC	Markov chain
MCMC	Markov chain Monte Carlo
MMACO	Minimum maximum ant colony optimization
NEDC	New European Driving Cycle
NN	Neural networks
PCA	Principal component analysis
PHEV	Plug-in hybrid electric vehicle
SA	Simulated annealing
SDC	Singapore driving cycle
SVM	Support vector machine
UDDS	Urban dynamometer driving schedule
WLTC	World light-duty vehicle test cycle

¹China Automotive Engineering Research Institute Co., Ltd., Chongqing 401122, China. ²New Energy Technology of CAERI Co., Ltd., Chongqing 401122, China. ³State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha 410082, China. ⁴Chongqing Research Institute of Hunan University, Chongqing 401120, China. ✉email: xileisun@hnu.edu.cn; fujianqin@hnu.edu.cn

The driving cycle delineates the intricate correlation between the temporal progression of vehicle speed within a specific setting, which is commonly referred to as the vehicle test cycle^{1,2}. As an essential fundamental technology for the automotive industry, the driving cycle plays a vital role in the realms of vehicle advancement, assessment and experimentation³. It is also the leading benchmark when optimizing various vehicle performance indicators^{4,5}. With the growth of vehicle ownership, the gap between the actual fuel consumption and the results of regulatory certification using standard driving cycles is gradually widening^{6,7}. Simultaneously, regional disparities in topography, climatic conditions, socioeconomic advancement, road topologies, and traffic patterns engender considerable variation, rendering the adoption of a uniform driving cycle impractical across diverse regions^{8,9}. At present, individual nations are endeavoring to devise driving cycles tailored to the idiosyncrasies of their actual road networks during the development, testing and evaluation of automobiles^{10,11}. As the most critical basic technology for the automotive industry, it is increasingly essential to develop actual driving cycles that fit the country or even a specific city^{12,13}. Therefore, in order to save development costs and improve reusability, it is necessary to propose a method for facilitating the transition from original vehicle data to a final representative driving cycle¹⁴.

In order to construct the actual driving cycle with regional characteristics, numerous scholars have carried out extensive empirical vehicle tests and research endeavors. Amirjamshidi et al.¹⁵ used simulation data derived from a coordinated micro-traffic model of the Toronto waterfront to generate a representative driving cycle, which had distinct regional characteristics and high emission factors. Representative driving cycles tailored for passenger cars and motorcycles were developed to reflect the authentic driving conditions of Chennai in Ref.¹⁶, resulting in two driving cycles of 1448 and 1065 s, respectively. Berzi et al.¹⁷ obtained a driving cycle by monitoring a fleet of electric vehicles and employing pseudo-random selection of raw data, and the results demonstrated commendable regenerative braking capacity and smooth traction at low speeds of the cycle. A new driving cycle construction method based on a two-level optimization process was proposed in Ref.¹⁸, which produced a more representative driving cycle that was closer to the statistical data of 2.49% than the traditional Markov chain (MC) method. Cui et al.¹⁹ presented a novel method based on simulated annealing (SA) algorithm, which led to a velocity-acceleration model that better matched real-world driving characteristics and significantly reduced errors by up to 23%. An innovative data-driven driving cycle development method based on minimum maximum ant colony optimization (MMACO) and the MC method was introduced to improve the representativeness of driving cycles in Ref.²⁰, potentially serving as a benchmark for establishing fuel consumption standards. Gong et al.²¹ collected high-frequency operational data from battery electric vehicles (BEVs) and established the Beijing driving cycle through statistical and MC methods, laying a robust foundation for the precise evaluation of BEV performance in Beijing. The inaugural driving cycle for gasoline-powered vehicles was produced for the Greater Cairo of Egypt based on a diverse collection of high-resolution on-board measurements in Ref.²², which was superior in estimating fuel consumption and emissions.

In addition, many studies have delved into examining vehicle energy consumption and emissions by utilizing autonomously constructed actual driving cycles. Achour et al.²³ estimated the contribution of private cars to local emission inventories based on a proposed representative driving cycle, and the strong representativeness of the driving cycle was verified by comparing with empirical measurements. The actual data from electric vehicles over six months were used to derive a driving cycle specifically tailored for evaluation in Ref.²⁴, and energy consumption calculations indicated that the driving cycle adeptly mirrors the genuine local driving conditions. Ho et al.²⁵ compared emissions of the Singapore Driving Cycle (SDC) and New European Driving Cycle (NEDC) using micro-estimation models, which revealed that NEDC underestimated most of vehicle emissions and SDC was more appropriate in Singapore. The online energy management of the Plug-in Hybrid Electric Vehicles (PHEV) was implemented using the dynamic programming (DP) algorithm based on the constructed actual driving cycle in Ref.²⁶, and simulation results demonstrated a minimum 19.83% improvement in fuel efficiency compared to the charge depletion and charge sustain (CDCS) control strategy. Koossalapeerom et al.²⁷ developed the driving cycle of electric motorcycles and measured the energy consumption, affirming the faithful reflection of the constructed cycle for real driving conditions. The fuel economy for both conventional and autonomous vehicles was precisely predicted according to the customized driving cycle in Ref.²⁸, further emphasizing the necessity of enhancing fuel economy estimates through the use of customized driving cycles. Ma et al.²⁹ introduced the AMarkov chain method to create representative driving cycles with actual driving characteristics, and the study underlined the significance of addressing real-world characteristics when improving fuel economy regulations. The CO₂ emissions of five passenger cars were simulated in actual driving cycles in Ref.³⁰, which revealed that local driving cycles were 30% closer to empirical data compared to the World Light-duty Vehicle Test Cycle (WLTC).

The constructed driving cycles exhibit diverse characteristics due to the varied objectives and methods adopted by researchers^{31,32}. However, there is a lack of a complete systematic construction method from the collected original data to the actual urban driving cycles, thus improving efficiency, saving costs and facilitating comparisons. At the same time, the representativeness of the actual driving cycles constructed at present needs to be further improved, so as to reflect local driving characteristics more realistically. Currently, artificial intelligence (AI) has become increasingly ubiquitous across diverse domains, playing a pivotal role in numerous applications^{33,34}. Deep learning is one of the essential components in the field of artificial intelligence³⁵, which has become an important technology because of its exceptional generalization and prediction performance^{36,37}. It is a novel and valuable research to construct driving cycles that are more relevant to the actual conditions by using deep learning as an effective tool. Therefore, a systematic driving cycle construction method called CS-DCC is proposed in this study, which integrates multi-scale wavelet analysis, KPCA, the Balanced Iterative Reducing and Clustering using Hierarchies (Birch) algorithm, and the improved autoencoder. The key contributions are delineated below.

- Electric vehicle road tests were conducted and relevant data were collected using the manual driving method.
- The CS-DCC method was proposed to systematically generate a representative driving cycle from the original data.
- The constructed driving cycle was compared with four standard driving cycles to verify the regional characteristics.
- The study introduced a new way to effectively use deep learning for constructing highly representative actual driving cycles.

Methods

In this study, the CS-DCC method is proposed to systematically generate a representative driving cycle from the original data, and the workflow is illustrated in Fig. 1. More specifically, the original data collected from electric vehicle road tests are sequentially subjected to seven steps, including data preprocessing, micro-trip division, characteristic extraction, dimensionality reduction, micro-trip clustering, driving cycle establishment and driving cycle optimization. These steps collectively lead to the creation of a highly representative actual driving cycle. During the data preprocessing phase, a sequential process is implemented involving missing data addressing, abnormal data handling and noise reduction, where multi-scale wavelet analysis is employed to mitigate data noise. In the micro-trip division step, the collected continuous data are segmented in a specific way according to the CS-DCC method. The characteristic extraction stage involves extracting characteristics from the micro-trips based on a predetermined set of 14 parameters specified in this algorithm. In the dimensionality reduction step, the KPCA algorithm is utilized to reduce the dimensionality of the characteristic matrix, which serves to alleviate computational complexity. The micro-trip clustering phase uses the Birch algorithm to cluster all micro-trips into three distinct classes based on predetermined criteria within the algorithm. During the driving cycle establishment stage, the Markov chain Monte Carlo (MCMC) method is applied to construct the driving cycle, leveraging the properties of stochastic processes with Markov characteristics. In the driving cycle optimization step, the constructed driving cycle is optimized based on the improved autoencoder to enhance its representativeness. It is worth noting that although the CS-DCC method is proposed based on city-specific road test data, it can still be applied to other road test data to generate actual driving cycles. More details about the seven steps are shown below.

The data preprocessing mainly includes two steps: (1) Interpolation of Missing and Abnormal Data; (2) Noise Reduction. The abnormal data are composed of driving at consistently low speeds intermittently and unusual acceleration patterns. Situations such as prolonged traffic congestion or intermittent low-speed driving (below 10 km/h within 30 s) are considered as an idling state, and the maximum continuous idling time is limited to 180 s. The processing method for intermittent low-speed driving is directly setting the value to 0 m/s and eliminating segments exceeding 180 s of idling, and the provided example in Fig. 2 demonstrates the comparison before and after this processing step. In this study, the time taken for the tested electric vehicles from 0 km/h to reach 100 km/h is assumed to be at most 7 s³⁸, and the maximum deceleration during emergency braking is set at -8 m/s²³⁹. When an abnormal acceleration point is identified, the speed value is replaced by interpolation based on the surrounding 10 s data (5 s before and after), and then the acceleration at that point is adjusted accordingly. If abnormal acceleration persists even after processing, the timestamp of the data point is recorded and the micro-trip containing the abnormal data is removed. The driving behavior of a vehicle is a complex and random process, and the driving status is influenced by various factors, such as non-motorized vehicles driving on the motorway and diverse road conditions. As a result, there is a large amount of noise in the collected original data. To address this, a five-scale wavelet decomposition is employed for signal reconstruction through inverse transformation, effectively removing diverse noise sources. The micro-trip division entails segmenting the speed-time curve within a specific duration according to the trajectory of “end of idling—driving—end of idling”, which is a prevalent technique for segment division and can effectively realize the continuous combination of micro-trip.

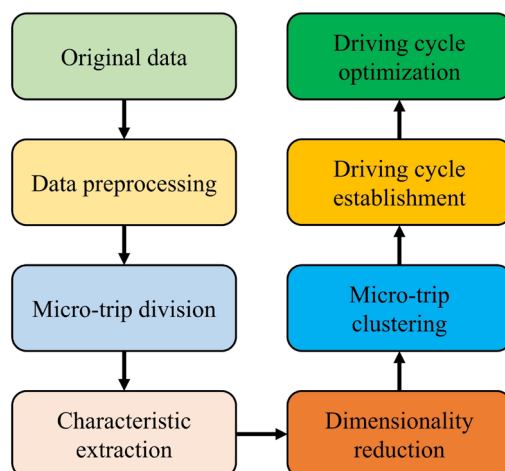


Figure 1. Workflow of the CS-DCC method.

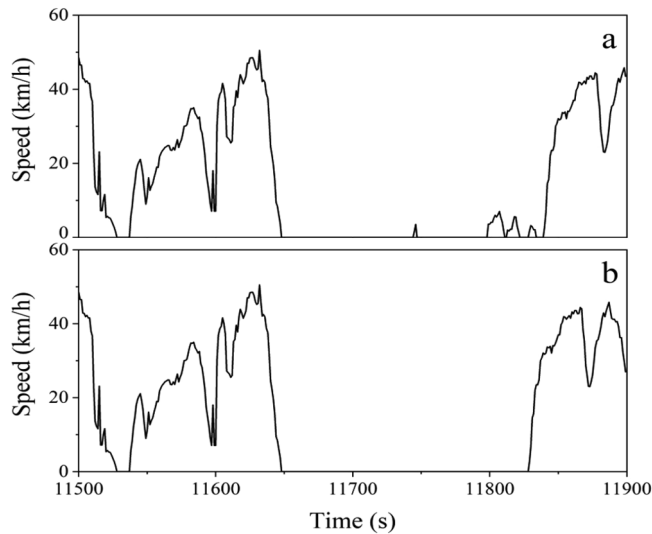


Figure 2. Speed-time curves before (a) and after (b) abnormal data processing.

Characteristic extraction involves the comprehensive characterization of the micro-trip by extracting essential parameters, and 14 parameters are selected in this study which are enumerated in Table 1 and calculated by Eqs. (1), (2), (3), (4), (5), (6), (7), (8), (9), (10), (11), (12), (13) and (14).

$$P_a = \frac{T_a}{T} \times 100\% \tag{1}$$

$$P_d = \frac{T_d}{T} \times 100\% \tag{2}$$

$$P_i = \frac{T_i}{T} \times 100\% \tag{3}$$

$$P_u = 1 - P_a - P_d - P_i \tag{4}$$

where P_a , P_d , P_i and P_u represent the time percentage corresponding to acceleration, deceleration, idling and uniform speed for the micro-trip, while T_a , T_d , T_i and T_u are the respective durations of these states; T is the total duration of the micro-trip.

$$v_{\max} = \max \{v_i, i = 1, 2, \dots, N\} \tag{5}$$

No	Parameter	Unit	Abbreviation
1	Time percentage of acceleration	-	P_a
2	Time percentage of deceleration	-	P_d
3	Time percentage of idling	-	P_i
4	Time percentage of uniform speed	-	P_u
5	Maximum speed	km/h	v_{\max}
6	Mean speed	km/h	v_m
7	Mean driving speed	km/h	v_{md}
8	Standard deviation of the speed	km/h	v_{sd}
9	Maximum acceleration	m/s ²	a_{\max}
10	Mean acceleration	m/s ²	a_{am}
11	Maximum deceleration	m/s ²	a_{\min}
12	Mean deceleration	m/s ²	a_{dm}
13	Standard deviation of the acceleration	m/s ²	a_{asd}
14	Standard deviation of the deceleration	m/s ²	a_{dsd}

Table 1. 14 characteristic parameters selected from the basic evaluation criteria.

$$v_m = \left(\sum_{i=1}^N v_i \right) / T \quad (6)$$

$$v_{md} = \left(\sum_{i=1}^N v_i \right) / (T - T_i) \quad (7)$$

$$v_{sd} = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (v_i - v_m)^2}, \quad i = 1, 2, \dots, N \quad (8)$$

where v_{max} , v_m , v_{md} and v_i are the maximum, mean, mean driving and idling speeds, respectively; v_{sd} and N are the standard deviation of speed and total time, respectively.

$$a_{max} = \max \{a_i, i = 1, 2, \dots, N - 1\} \quad (9)$$

$$a_{am} = \frac{\sum a_i^a}{T_a}, \quad i = 1, 2, \dots, N - 1 \quad (10)$$

$$a_{min} = \min \{a_i, i = 1, 2, \dots, N - 1\} \quad (11)$$

$$a_{dm} = \frac{\sum a_i^d}{T_d}, \quad i = 1, 2, \dots, N - 1 \quad (12)$$

$$a_{asd} = \sqrt{\frac{1}{N_1 - 1} \sum_{i=1}^{N_1} (a_i^a - a_{am})^2}, \quad i = 1, 2, \dots, N_1 \quad (13)$$

$$a_{dsd} = \sqrt{\frac{1}{N_2 - 1} \sum_{i=1}^{N_2} (a_i^d - a_{dm})^2}, \quad i = 1, 2, \dots, N_2 \quad (14)$$

where a_{max} and a_{am} are the maximum and mean accelerations, respectively; a_{min} and a_{dm} are the maximum and mean decelerations, respectively; a_{asd} and a_{dsd} are the standard deviation of acceleration and deceleration, respectively; a_i^a and a_i^d are the acceleration and deceleration at the i -th second, respectively; N_1 and N_2 are the total time of acceleration and deceleration processes, respectively.

The 14 characteristic parameters extracted from micro-trips contain information pertaining to speed, acceleration and time percentage, which has overlapping and redundant information. To address this, the KPCA algorithm⁴⁰ is employed in this study for dimensionality reduction to reduce computational complexity and improve efficiency. The KPCA algorithm operates on the kernel trick, enabling the kernelization of linear dimensionality reduction techniques to identify an appropriate low-dimensional embedding. This approach effectively mitigates dimensional discrepancies and preserves distinct information across each characteristic factor, ultimately enhancing the realism and reliability of the final outcomes.

In accordance with the classification criteria observed in the WLTC and China Light-duty Vehicle Test Cycle (CLTC), the vehicle speed is categorized into three classes which are low, medium and high speeds. In this study, all micro-trips are clustered using the Birch algorithm⁴¹ following these principles. The Birch algorithm is a robust and classical hierarchical clustering technique that can efficiently cluster data with a single scan and handle outliers, making it well-suited for managing extensive datasets. This distance-based hierarchical clustering method initially performs a bottom-up hierarchical coalescing process and subsequently employs iterative relocation to refine the results. During the hierarchical coalescing phase, individual objects are considered as atomic clusters and progressively combined to create larger clusters until all objects belong to a cluster or a specified end condition is met.

The process of establishing a driving cycle involves selecting specific micro-trips in a manner that constructs a speed-time curve of a defined length, endeavoring to capture the actual driving characteristics to the greatest extent possible. The MCMC method⁴² is employed for constructing the actual driving cycle in this study, which estimates the posterior distribution of interest parameters by utilizing random sampling in the probability space. Each subsequent sample in this process relies on the previous sample, thereby creating a stochastic process model with Markov properties. The Markov chain principle assumes that the probability of transitioning from one state to another at any given moment depends solely on the preceding state. This might seem arbitrary, but it serves as an effective approach to streamline the complexity of the model, substantially simplifying calculations. Mathematically, assuming that the state sequence is $\dots X_{t-2}, X_{t-1}, X_t, X_{t+1} \dots$, then the conditional probability of the state X_{t+1} depends only on the previous state X_t :

$$P(X_{t+1} | \dots X_{t-2}, X_{t-1}, X_t) = P(X_{t+1} | X_t) \quad (15)$$

The process of constructing the actual driving cycle through the MCMC method involves several steps: (1) Calculating the Markov chain state transition matrix P derived from the clustering results; (2) Given an arbitrarily set initial state distribution probability Z , obtaining the final stable probability distribution Q through the iterative action of Z and P ; (3) Employing the Monte Carlo sampling technique to generate the driving cycle that satisfies specified conditions based on the derived probability distribution Q .

In order to create a driving cycle that authentically reflects the data characteristics of the vehicle under actual driving conditions, it is crucial not only to align with the real data distribution but also to minimize the average characteristic error with the entire dataset. Therefore, an improved version of the autoencoder⁴³ is employed to optimize the constructed actual driving cycle, which learns the effective encoding of a dataset in an unsupervised way. The traditional autoencoder constructs a reconstruction loss based on the input and output data, which minimizes the gap between the output and input data by reducing the reconstruction loss, aiming to closely align the output with the input data. In this study, the constraints of the specific physical problem are integrated into the design of the autoencoder. Specifically, the average error of the characteristic parameters between the constructed driving cycle and the total data is taken as the characteristic loss, and a new loss is derived by weighted summation of the characteristic loss and the reconstruction loss. In this way, the driving cycle output from the improved autoencoder can maintain the essential driving characteristics of the original driving cycle under the constraint of the reconstruction loss. Meanwhile, the output driving cycle can further reduce the average error of the characteristic parameters with the total data under the constraint of feature loss, so as to improve the overall representativeness of the driving cycle.

$$L = \lambda_1 L_r + \lambda_2 L_f \tag{16}$$

$$L_r = \sum_{t=1}^T (x_t - x_{t'})^2 \tag{17}$$

$$L_f = \frac{1}{M} \sum_{i=1}^M (F_{i'} - F_i^{data})^2 \tag{18}$$

where $X = x_1, x_2, \dots, x_T$ is the initial driving cycle, x_t is the vehicle speed at time t , T is the total cycle time, F_{data} i represents the value of the i -th characteristic parameter of the total data, M is the total number of characteristic parameters, L_r and L_f are the reconstruction loss and the characteristic loss, respectively.

The structure of the improved autoencoder is presented in Fig. 3, which consists of three primary parts: the input layer, the hidden layer and the output layer. The driving cycle constructed based on the MCMC method is fed into the input layer, while the driving cycle optimized by the improved autoencoder is output from the output layer. The hidden layer comprises an encoder and a decoder, the encoder involves three fully-connected neural networks with a decreasing number of neurons for the purpose of down-sampling. Conversely, the decoder is comprised of three fully connected neural networks with an increasing number of neurons to facilitate up-sampling.

Experiments and data

The experiment was conducted using three light-duty battery electric vehicles in Changsha, China. The data acquisition equipment comprised the Speed BOX data acquisition instrument, CAN module and transmission line, with a data acquisition frequency of 1 Hz. The Speed BOX serves as an input–output terminal capable of real-time signal collection, including vehicle speed, altitude, latitude, and longitude. Furthermore, it communicates with external systems through the CAN module, allowing reception of external signals such as accelerator and brake pedal opening signals. The primary signals recorded during vehicle operation include vehicle speed, torque and accelerator pedal opening. The manual driving method was employed to obtain the speed-time curve

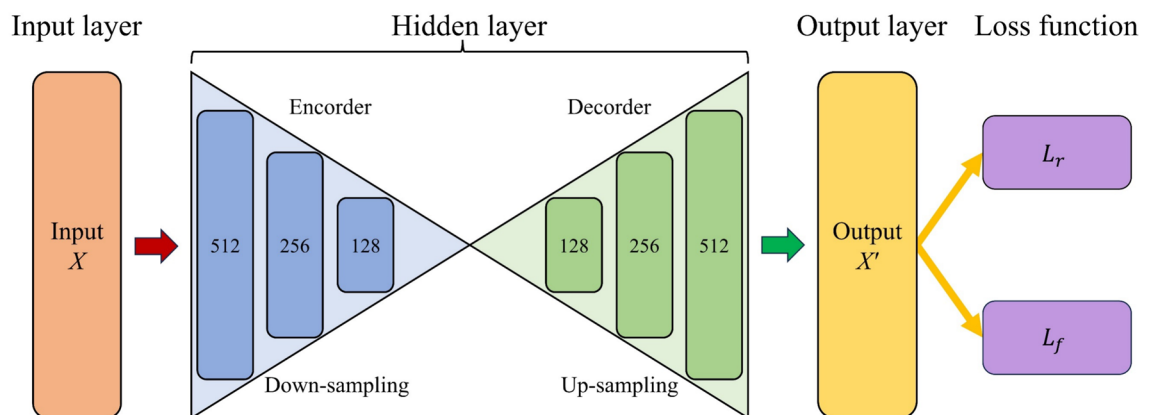


Figure 3. Structure of the improved autoencoder.

during actual driving, which granted drivers the freedom to conduct tests based on their driving habits without limitations concerning time, space or location. Consequently, the test data obtained through this manual driving method demonstrated a higher degree of randomness and closer approximation to real-world conditions. At the same time, a different driver was designated each day to prevent data collection from being influenced by driving styles. Considering fewer vehicles on the road during night and early morning hours, data collection was set between 7:00 a.m. and 6:00 p.m. daily. The test equipment, driving paths and datasets are illustrated in Fig. 4. Although only one driving path is displayed due to the multitude of paths, it is evident that the driving routes cover the city comprehensively.

Results and discussions

Firstly, a total of 136 data points with missing information were identified after detailed examination, primarily resulting from weakened GPS signals when passing through tunnels or underground passages. In addition, 1115 abnormal data instances were initially detected, and this number was reduced to 295 through interpolation, which were found to be within 216 micro-trips. Consequently, these 216 micro-trips were excluded, resulting in 1058, 699 and 793 micro-trips extracted from the three datasets, summing up to a total of 2550 micro-trips as defined. The first dataset along with its approximation and detail coefficients under five-scale wavelet decomposition before and after noise reduction are illustrated in Fig. 5. The original data were subjected to five-scale wavelet decomposition, where the noise signal within the detail coefficients with higher frequencies was addressed using a gate threshold. Subsequently, the detail and approximate coefficients were wavelet reconstructed to achieve noise reduction. Considering the vast amount of data, the change in a micro-trip within the first dataset before and after noise reduction was specifically selected to illustrate the effect of multi-scale wavelet analysis, as shown in Fig. 6. It can be seen that the refined data after noise reduction exhibit greater stability and a smoother pattern in contrast to the original data.

Secondly, the Pearson correlation coefficients between the 14 characteristic parameters are demonstrated in Fig. 7 and calculated by Eq. (19):

$$Pearson_{XY} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (19)$$

where $X = (x_1, x_2, \dots, x_n)$ and $Y = (y_1, y_2, \dots, y_n)$ are n -dimensional vectors; \bar{x} and \bar{y} are the mean values of X and Y , respectively. The Pearson correlation coefficient is an indicator to measure the linear correlation between two variables, which offers a straightforward calculation, easy comprehension, and widespread applicability. The value of the Pearson correlation coefficient ranges from -1 to 1 , where a value closer to 1 signifies a stronger positive correlation, while a value closer to -1 indicates a stronger negative correlation. It can be seen that there are extremely strong positive or negative correlations among various variables, so it is necessary to reduce the dimensionality of the 14 characteristic parameters, thus avoiding unnecessary calculations and improving efficiency. Figure 8 presents the variation in the cumulative contribution rate concerning the number of principal components based on the KPCA algorithm. In this study, the criterion for selecting the number of principal components is based on a cumulative contribution rate of 80% and above. It can be observed that the cumulative contribution of the principal components stands at 85.99% when the number of principal components reaches 5. As a result, the determined number of principal components is 5. The principal component load matrix of the

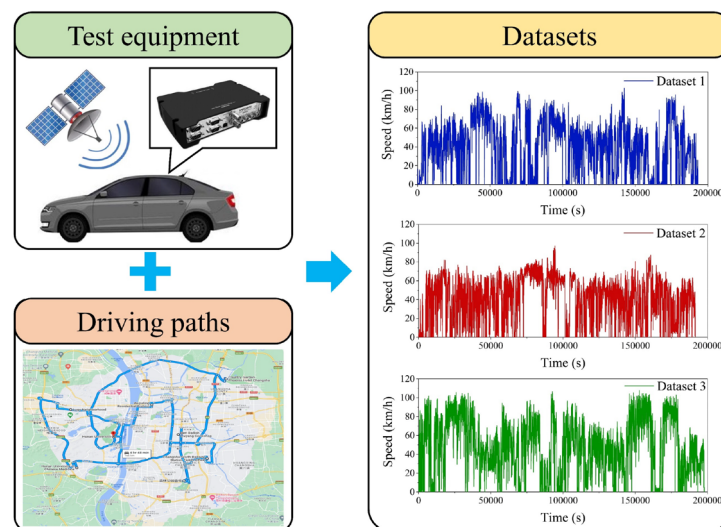


Figure 4. Test equipment, driving paths and datasets (Google Maps 2023, <https://www.google.com/maps>).

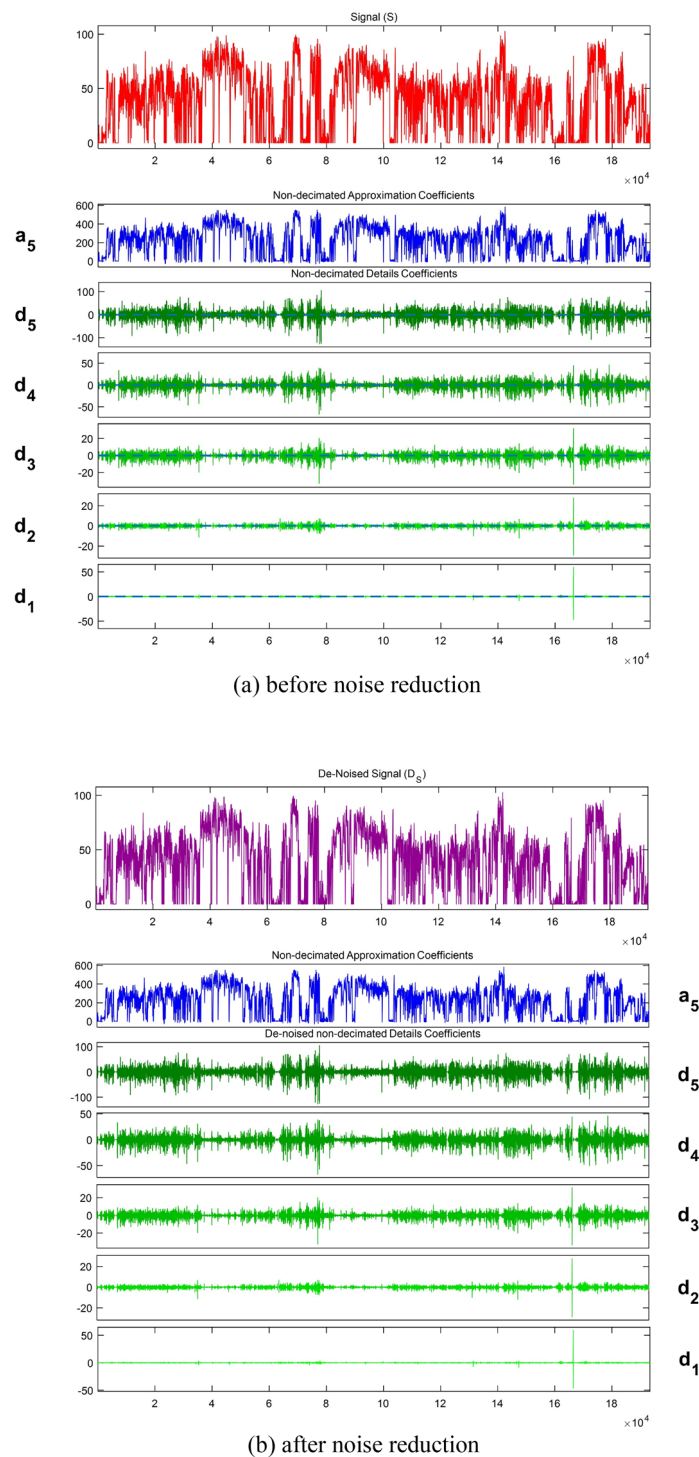


Figure 5. The first dataset with its approximation and detail coefficients before and after noise reduction.

principal components and original characteristics is listed in Table 2, in which $r_{6,1} = 0.9467$ indicates that the original characteristic X_6 has a strong correlation with the first principal component Z_1 . The principal component load matrix not only illustrates the robust representativeness of the five chosen principal components, but also emphasizes the validity and soundness of this selection.

Thirdly, the KPCA algorithm using four distinct kernel functions was used to reduce the dimensionality of the matrix, which contained the characteristic parameters of all micro-trips. The comparison and analysis of these results were conducted to choose the optimal kernel function, considering that the number of principal components discussed earlier was determined to be five. The scatter plots of the first and second principal components after the dimensionality reduction based on the KPCA algorithm are displayed in Fig. 9, and it can be found that various kernel functions map the data into different high-dimensional spaces. Notably, the Gaussian KPCA

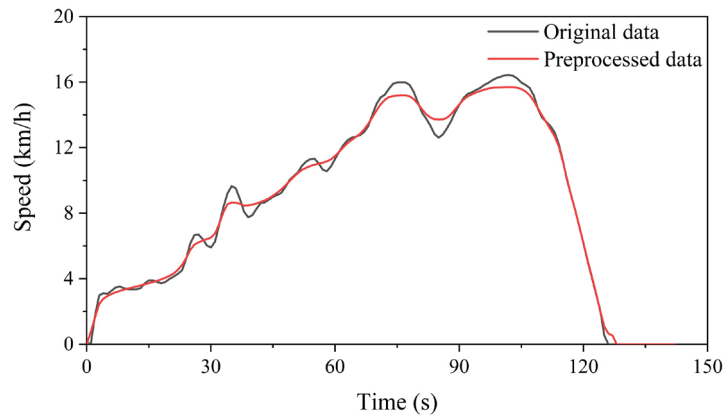


Figure 6. Change in a micro-trip within the first dataset before and after noise reduction.

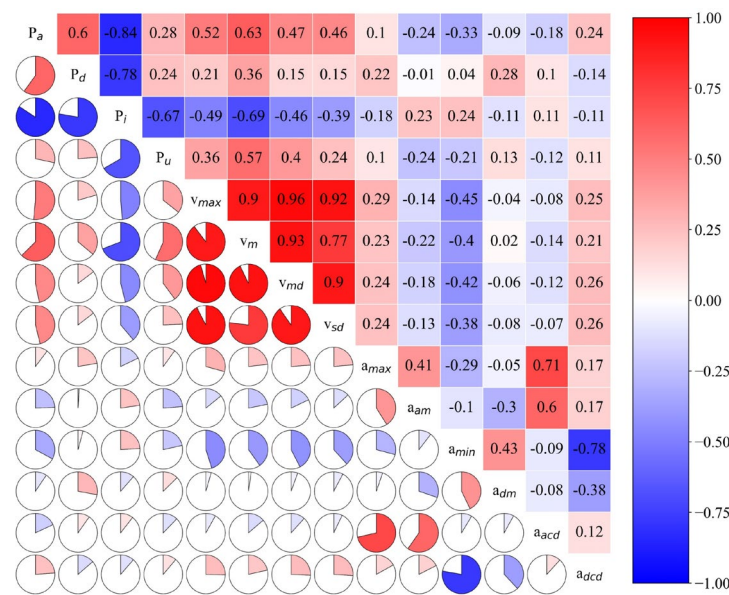


Figure 7. Pearson correlation coefficients between the 14 characteristic parameters.

and Cosine KPCA exhibit relatively dense data in high-dimensional spaces, while the Polynomial KPCA and Linear KPCA display scattered data. The micro-trips after dimensionality reduction by four KPCA algorithms were clustered using the Birch algorithm, and the results are given in Table 3 while the scatter plots of the first and second principal components are presented in Fig. 10. It should be noted that there is a significant difference in the number of three classes after dimensionality reduction by the Linear KPCA and Poly KPCA algorithms, which indicates that the results obtained using these two KPCA algorithms are notably less reasonable. Employing the Linear KPCA algorithm for dimensionality reduction results in 1725 micro-trips in the first class and only 214 in the third class, while using the Poly KPCA algorithm produces merely one micro-trip in the second class and 2537 micro-trips in the third class. On the contrary, the Gaussian KPCA and Cosine KPCA algorithms yield more balanced and coherent outcomes in terms of the distribution of micro-trips across the three classes, particularly the Gaussian KPCA algorithm. Considering the distribution of data in high-dimensional space and the class balance following clustering, the Gaussian KPCA algorithm is chosen for reducing the dimensionality of the characteristic matrix.

The characteristic parameters of the three classes obtained by the Gaussian KPCA and Birch algorithms are detailed in Table 4, which differ from each other. The first class exhibits a moderate speed, and the ratio of the acceleration and deceleration time is relatively high which may be in the unobstructed period on urban roads. The average speed of the second class is small and the proportion of idling time is high, suggesting probable scenarios of vehicle start-up or road congestion. The third class showcases high speed, a minimal proportion of idling time and a significant time percentage of acceleration, which may represent driving at high speeds in suburban areas.

Finally, the MCMC algorithm is applied to construct the actual driving cycle, which is further optimized by the improved autoencoder to enhance its representativeness. The loss function during the training process of

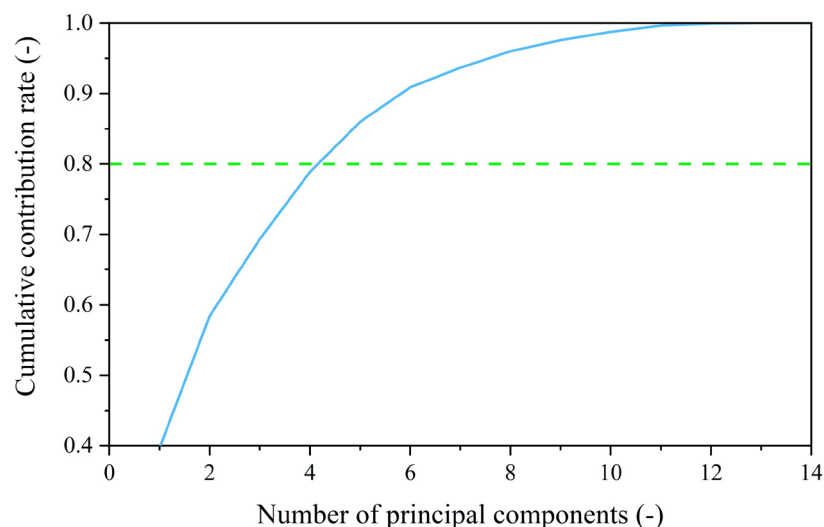


Figure 8. Variation in the cumulative contribution rate concerning the number of principal components.

Original characteristic	Principal component				
	Z ₁	Z ₂	Z ₃	Z ₄	Z ₅
X ₁	0.7926	-0.0879	0.2613	0.3203	0.2776
X ₂	0.5678	-0.3258	0.6598	0.0575	0.1894
X ₃	-0.8072	0.2239	-0.4710	-0.2630	-0.0428
X ₄	0.4497	-0.1201	0.1445	0.2228	-0.6732
X ₅	0.8887	0.2209	-0.2231	-0.2746	0.0482
X ₆	0.9467	0.0836	-0.0315	-0.1354	0.0401
X ₇	0.8789	0.2115	-0.2562	-0.2784	0.0514
X ₈	0.7985	0.2480	-0.2935	-0.3051	0.1126
X ₉	0.1395	0.6042	0.4589	-0.4580	-0.2039
X ₁₀	-0.4708	0.5813	0.3206	-0.0226	0.2675
X ₁₁	-0.4353	-0.6418	0.1194	-0.4106	0.1439
X ₁₂	0.2218	-0.6453	0.0349	-0.3172	-0.4296
X ₁₃	-0.3877	0.5730	0.4745	-0.3136	-0.1896
X ₁₄	0.1620	0.6504	-0.1486	0.5233	0.2251

Table 2. Principal component load matrix of the principal components and original characteristics.

the model is illustrated in Fig. 11, where the loss function gradually decreases and finally converges. The model undergoes training initially at a learning rate of 0.1 to accelerate the training for the first 30,000 iterations, followed by a learning rate adjustment to 0.01 to refine model accuracy after 30,000 iterations. The total training spans 50,000 iterations, resulting in a final loss of 0.9084, with loss outputs recorded every 50 iterations. The characteristic parameters of the constructed driving cycle before and after optimization compared with the total data are given in Table 5. The average error of the characteristic parameters between the optimized drive cycle and total data is notably reduced from 13.6% to 6.1%, with a reduction ratio of 55.1%. This reduction in error is achieved while preserving driving properties, showcasing the remarkable optimization performance of the improved autoencoder model. In order to demonstrate the effectiveness and advancement of the proposed CS-DCC method, a comparison is carried out based on the same dataset using the method developed in Ref.¹⁹, where a SA-based method is introduced. The driving cycles obtained based on the two methods are illustrated in Fig. 12. It can be seen that although the two methods use the same dataset, the constructed driving cycles consist of different micro-trips. The 14 characteristic parameters of the driving cycle obtained by the SA-based method are listed in Table 4, and the average error with respect to the total data is 9.7%. The characteristic parameters of the driving cycle obtained by the CS-DCC method have an average error of 6.1%, with an improvement ratio of 37.1%.

The driving cycle constructed by the CS-DCC method is compared with four standard driving cycles and the results are shown in Table 6. It can be seen that the time percentage of acceleration in the constructed driving cycle exceeds that of the four driving cycles, while the ratio of idling time is lower than the Japanese 10–15 Mode (J10–15). However, the time proportion of uniform speed in the constructed driving cycle is the lowest.

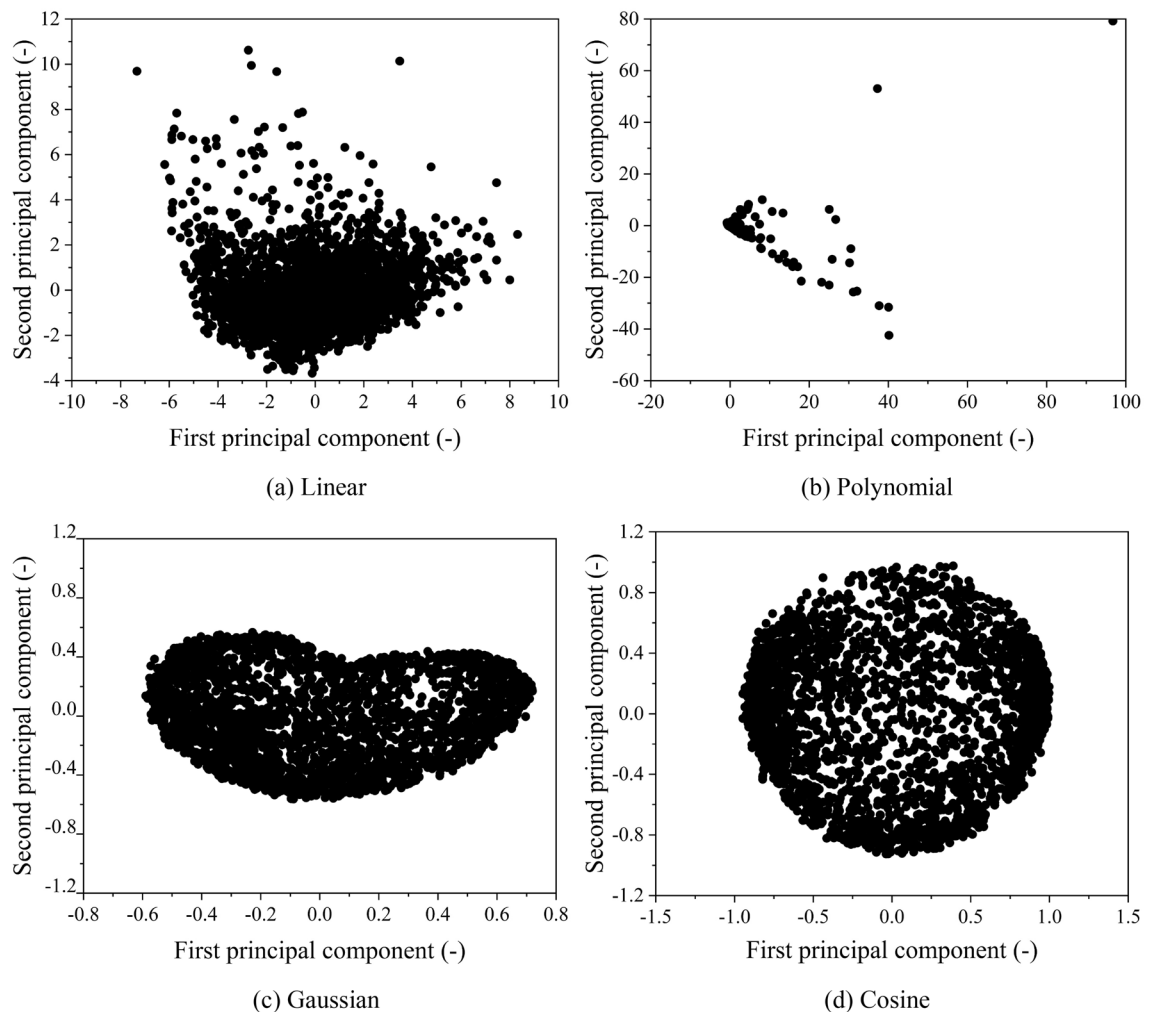


Figure 9. Scatter plots of the first and second principal components after dimensionality reduction by KPCA.

Class	Linear	Poly	Gaussian	Cosine
1	1725	12	774	1203
2	611	1	972	934
3	214	2537	804	413

Table 3. Results of micro-trip clustering using the Birch algorithm.

Notably, the average driving speed aligns closely with that of Urban Dynamometer Driving Schedule (UDDS), whereas the average acceleration is close to WLTC and significantly lower than the other three driving cycles. This comparison underscores the locally prominent characteristics present in the driving cycle generated through the CS-DCC method in contrast to standard driving cycles, which emphasizes the necessity of constructing an actual driving cycle that reflects localized driving patterns.

Conclusions

In this study, extensive efforts have been dedicated to the development of representative actual driving cycles. Electric vehicle road tests were conducted and relevant data were collected using the manual driving method, and the CS-DCC method was proposed to systematically generate a representative driving cycle from the original data. Besides, the constructed driving cycle was compared with four standard driving cycles to verify the regional characteristics, and the main conclusions are summarized as follows.

- (1) After noise reduction by five-scale wavelet analysis, the refined data exhibit greater stability and a smoother pattern in contrast to the original data. Analysis based on the Pearson correlation coefficients indicates

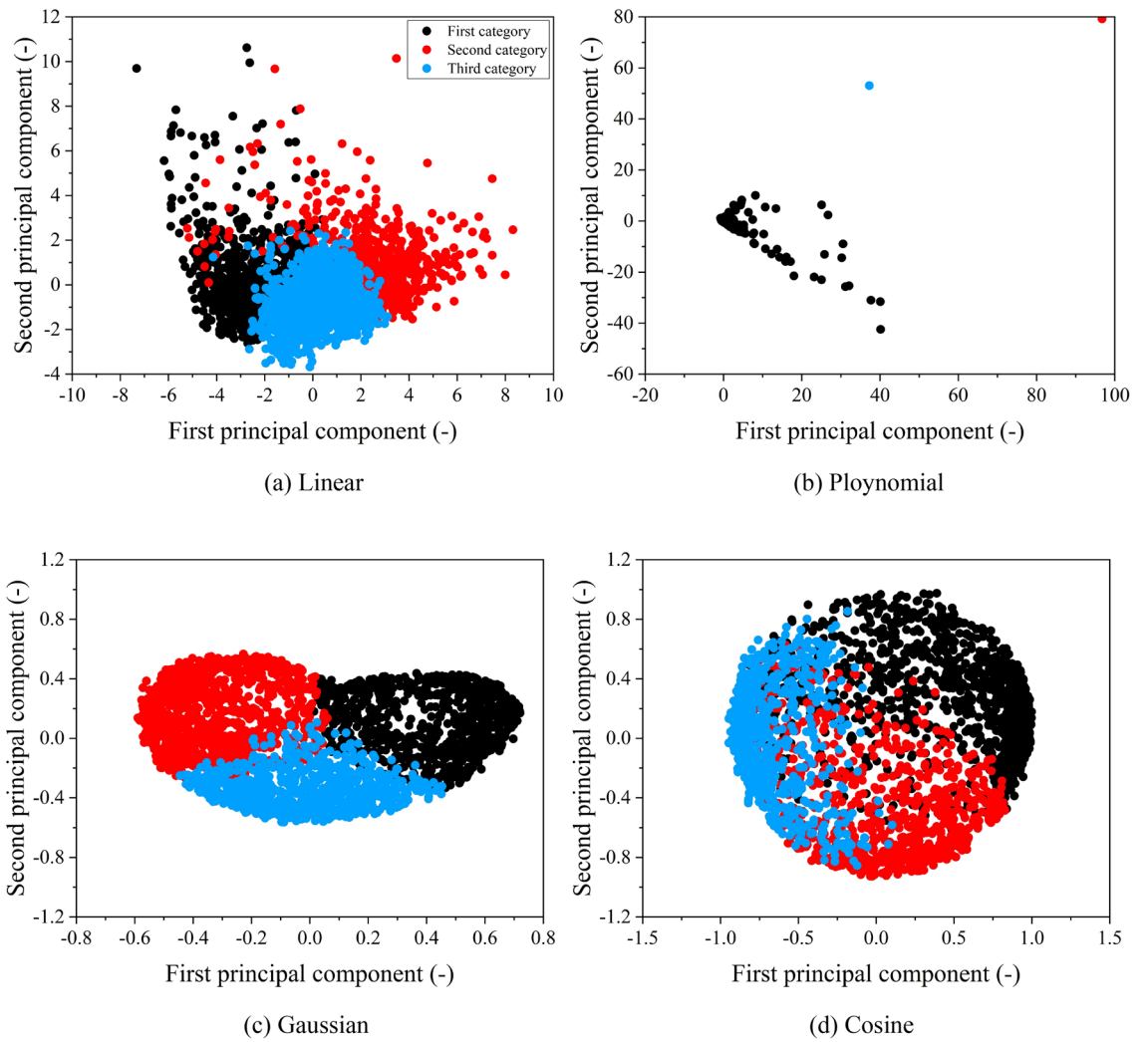


Figure 10. Scatter plots of the first and second principal components after clustering by Birch.

No	Parameter	Class 1	Class 2	Class 3
1	P_a	0.522	0.350	0.377
2	P_d	0.298	0.317	0.135
3	P_i	0.145	0.322	0.094
4	P_u	0.034	0.010	0.394
5	v_{max}	72.4	36.5	101.5
6	v_m	27.935	16.356	45.168
7	v_{md}	30.807	21.118	56.465
8	v_{sd}	17.449	14.831	34.170
9	a_{max}	2.9	3.2	3.3
10	a_{am}	0.379	0.524	0.264
11	a_{min}	-6.321	-7.786	-8.223
12	a_{dm}	-0.541	-0.663	-0.503
13	a_{asd}	0.409	0.475	0.398
14	a_{dsd}	0.595	0.864	0.541

Table 4. Characteristic parameters of three classes obtained by the Gaussian KPCA and Birch algorithms.

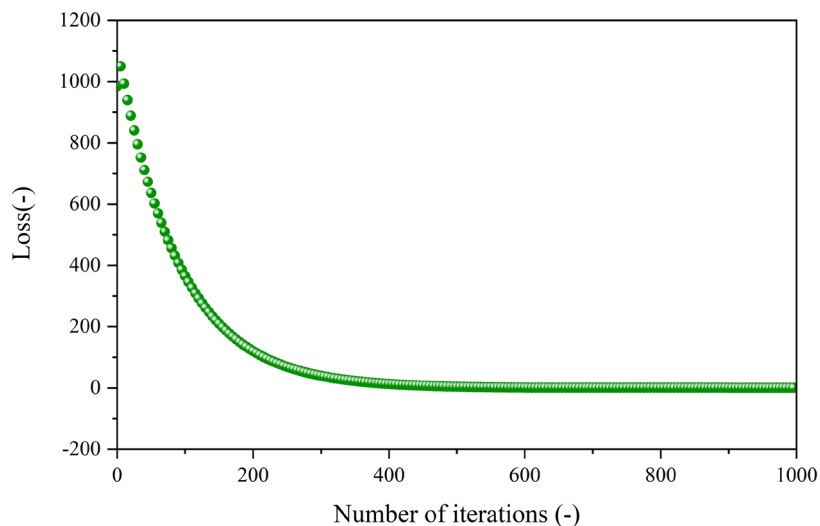


Figure 11. Loss function during the training process of the improved autoencoder model.

No	Parameter	Total data	Before optimization	After optimization (CS-DCC method)	SA-based method ¹⁹
1	P_a	0.417	0.466	0.438	0.367
2	P_d	0.250	0.315	0.274	0.243
3	P_i	0.187	0.181	0.181	0.225
4	P_u	0.146	0.038	0.107	0.164
5	v_{max}	101.500	93.200	94.890	87.536
6	v_m	29.820	30.419	30.407	26.107
7	v_{md}	36.130	37.147	37.144	34.146
8	v_{sd}	22.150	24.146	24.142	23.133
9	a_{max}	3.263	3.389	3.367	2.980
10	a_{am}	0.389	0.356	0.364	0.354
11	a_{min}	-8.223	-7.778	-7.978	-7.266
12	a_{dm}	-0.569	-0.525	-0.582	-0.493
13	a_{asd}	0.427	0.354	0.412	0.399
14	a_{dsd}	0.667	0.599	0.651	0.643
15	Average error	0	13.6%	6.1%	9.7%

Table 5. Comparison between the 14 characteristic parameters of the constructed driving cycles and the total data.

- the presence of extremely strong positive or negative correlations among the 14 extracted characteristic parameters.
- (2) Considering the distribution of the data in the high-dimensional space and the number of three classes after clustering, the Gaussian KPCA algorithm is chosen to reduce the dimensionality of the characteristic matrix. The number of principal components is determined as 5, and the cumulative contribution rate is 85.99%.
 - (3) The characteristic parameters of the three classes obtained by the Gaussian KPCA and Birch algorithms differ from each other. The average error of the characteristic parameters between the optimized drive cycle and total data is notably reduced from 13.6 to 6.1%, with a reduction ratio of 55.1%, showcasing the remarkable optimization performance of the improved autoencoder model.
 - (4) The proposed CS-DCC method demonstrates an effective method for constructing highly representative actual driving cycles, and the constructed driving cycle has obvious local characteristics in contrast to four standard driving cycles, which emphasizes the necessity of constructing an actual driving cycle that reflects localized driving patterns.

All of these not only provide an efficient method for the methodical construction of a representative driving cycle from original data, but also present a powerful application of artificial intelligence in advancing engineering technologies.

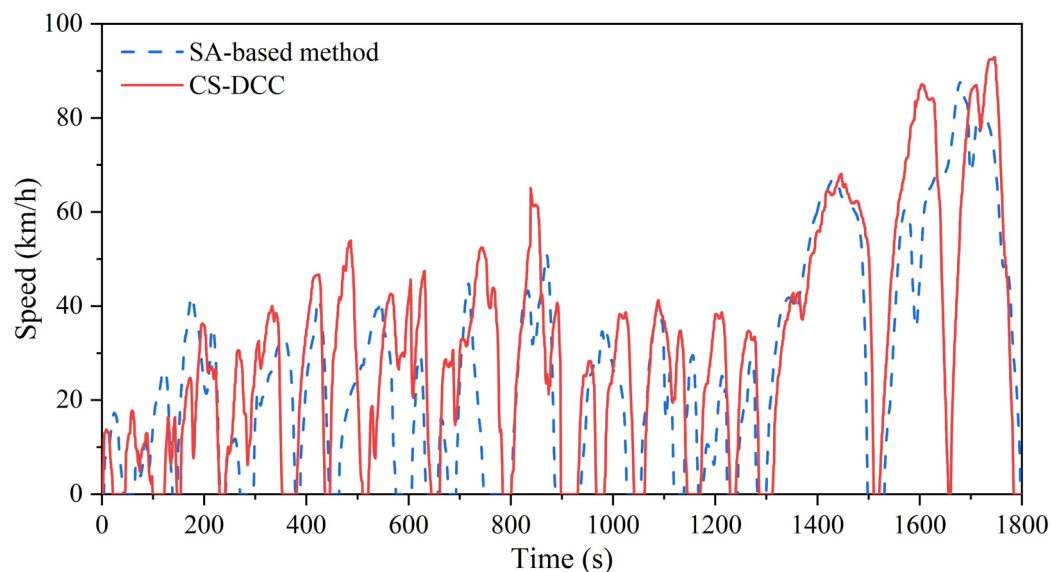


Figure 12. Driving cycles obtained based on the CS-DCC and SA-based methods.

Cycle	$T/(s)$	$P_d/(-)$	$P_f/(-)$	$P_w/(-)$	$V_m/(m \cdot s^{-1})$	$a_{am}/(m \cdot s^{-2})$
Constructed cycle	1800	0.4383	0.2744	0.1061	8.446	0.364
WLTC	1800	0.335	0.242	0.142	6.142	0.392
UDDS	1369	0.294	0.178	0.273	8.752	0.500
NEDC	1180	0.229	0.239	0.374	9.202	0.540
J10-15	660	0.261	0.326	0.195	6.300	0.540

Table 6. Results of the constructed driving cycle compared with four standard driving cycles.

Data availability

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Received: 31 May 2023; Accepted: 24 January 2024

Published online: 29 January 2024

References

- Adak, P., Sahu, R. & Elumalai, S. P. Development of emission factors for motorcycles and shared auto-rickshaws using real-world driving cycle for a typical Indian city. *Sci. Total Environ.* **544**, 299–308 (2016).
- Tong, H. Y. Development of a driving cycle for a supercapacitor electric bus route in Hong Kong. *Sustain. Cities Soc.* **48**, 101588 (2019).
- Wang, Y. *et al.* Fuel consumption and emission performance from light-duty conventional/hybrid-electric vehicles over different cycles and real driving tests. *Fuel* **278**, 118340 (2020).
- Bhatti, A. H. U. *et al.* Development and analysis of electric vehicle driving cycle for hilly urban areas. *Transp. Res. D Transp. Environ.* **99**, 103025 (2021).
- Lleti, R. *et al.* Selecting variables for k-means cluster analysis by using a genetic algorithm that optimises the silhouettes. *Anal. Chimica Acta* **515**, 87–100 (2004).
- Sun, X. *et al.* Hierarchical evolutionary construction of neural network models for an Atkinson cycle engine with double injection strategy based on the PSO-Nadam algorithm. *Fuel* **333**, 126531 (2023).
- Günther, R. *et al.* Big data driven dynamic driving cycle development for busses in urban public transportation. *Transp. Res. D Transp. Environ.* **51**, 276–289 (2017).
- Nesamani, K. S. & Subramanian, K. P. Development of a driving cycle for intra-city buses in Chennai, India. *Atmos. Environ.* **45**, 5469–5476 (2011).
- Anaman, R. *et al.* Identifying sources and transport routes of heavy metals in soil with different land uses around a smelting site by GIS based PCA and PMF. *Sci. Total Environ.* **823**, 153759 (2022).
- Zhang, J. *et al.* Driving cycles construction for electric vehicles considering road environment: A case study in Beijing. *Appl. Energy* **253**, 113514 (2019).
- Pouresmaeili, M. A., Aghayan, I. & Taghizadeh, S. A. Development of Mashhad driving cycle for passenger car to model vehicle exhaust emissions calibrated using on-board measurements. *Sustain. Cities Soc.* **36**, 12–20 (2018).
- Fu, J. *et al.* Multi-parameter optimization for the performance of the fuel cell air compressor based on computational fluid dynamics analysis at part load. *Thermal Sci. Eng. Progress* **44**, 102057 (2023).

13. Quirama, L. F. *et al.* Main characteristic parameters to describe driving patterns and construct driving cycles. *Transp. Res. D Transp. Environ.* **97**, 102959 (2021).
14. Tao, S. *et al.* Development of a representative driving cycle for evaluating exhaust emission and fuel consumption for Chinese switcher locomotives. *Appl. Energy* **322**, 119499 (2022).
15. Amirjamshidi, G. & Roorda, M. J. Development of simulated driving cycles for light, medium, and heavy duty trucks: Case of the Toronto Waterfront Area. *Transp. Res. D Transp. Environ.* **34**, 255–266 (2015).
16. Arun, N. H. *et al.* Development of driving cycles for passenger cars and motorcycles in Chennai, India. *Sustain. Cities Soc.* **32**, 508–512 (2017).
17. Berzi, L., Delogu, M. & Pierini, M. Development of driving cycles for electric vehicles in the context of the city of Florence. *Transp. Res. D Transp. Environ.* **47**, 299–322 (2016).
18. Chen, Z. *et al.* Optimization-based method to develop practical driving cycle for application in electric vehicle power management: A case study in Shenyang, China. *Energy* **186**, 115766 (2019).
19. Cui, Y. *et al.* Optimization based method to develop representative driving cycle for real-world fuel consumption estimation. *Energy* **235**, 121434 (2021).
20. Cui, Y. *et al.* A novel optimization-based method to develop representative driving cycle in various driving conditions. *Energy* **247**, 123455 (2022).
21. Gong, H. *et al.* Generation of a driving cycle for battery electric vehicles: A case study of Beijing. *Energy* **150**, 901–912 (2018).
22. Huzayyin, O. A., Salem, H. & Hassan, M. A. A representative urban driving cycle for passenger vehicles to estimate fuel consumption and emission rates under real-world driving conditions. *Urban Clim.* **36**, 100810 (2021).
23. Achour, H. & Olabi, A. G. Driving cycle developments and their impacts on energy consumption of transportation. *J. Clean. Prod.* **112**, 1778–1788 (2016).
24. Brady, J. & O'Mahony, M. Development of a driving cycle to evaluate the energy economy of electric vehicles in urban areas. *Appl. Energy* **177**, 165–178 (2016).
25. Ho, S. H., Wong, Y. D. & Chang, V. W. C. Developing Singapore Driving Cycle for passenger cars to estimate fuel consumption and vehicular emissions. *Atmos. Environ.* **97**, 353–362 (2014).
26. Hongwen, H. *et al.* Real-time global driving cycle construction and the application to economy driving pro system in plug-in hybrid electric vehicles. *Energy* **152**, 95–107 (2018).
27. Koosalapeerom, T. *et al.* Comparative study of real-world driving cycles, energy consumption, and CO₂ emissions of electric and gasoline motorcycles driving in a congested urban corridor. *Sustain. Cities Soc.* **45**, 619–627 (2019).
28. Liu, J., Wang, X. & Khattak, A. Customizing driving cycles to support vehicle purchase and use decisions: Fuel economy estimation for alternative fuel vehicle users. *Transp. Res. D Transp. Environ.* **67**, 280–298 (2016).
29. Ma, R. *et al.* Real-world driving cycles and energy consumption informed by large-sized vehicle trajectory data. *J. Clean. Prod.* **223**, 564–574 (2019).
30. Mafi, S. *et al.* Developing local driving cycle for accurate vehicular CO₂ monitoring: A case study of Tehran. *J. Clean. Prod.* **336**, 130176 (2022).
31. Sun, R. *et al.* Hybrid electric buses fuel consumption prediction based on real-world driving data. *Transp. Res. D Transp. Environ.* **91**, 102637 (2021).
32. Miotti, M. *et al.* Quantifying the impact of driving style changes on light-duty vehicle fuel consumption. *Transp. Res. D Transp. Environ.* **98**, 102918 (2021).
33. Sun, X. & Fu, J. Many-objective optimization of BEV design parameters based on gradient boosting decision tree models and the NSGA-III algorithm considering the ambient temperature. *Energy* **288**, 129840 (2024).
34. Sun, X. *et al.* An energy management strategy for plug-in hybrid electric vehicles based on deep learning and improved model predictive control. *Energy* **269**, 126772 (2023).
35. Mnih, V. *et al.* Human-level control through deep reinforcement learning. *Nature* **518**, 529–533 (2015).
36. Sun, X. *et al.* Multi-objective optimization for combustion, thermodynamic and emission characteristics of Atkinson cycle engine using tree-based machine learning and the NSGA II algorithm. *Fue* **342**, 127839 (2023).
37. Bao, H. *et al.* Performance prediction of the centrifugal air compressor for fuel cells considering degradation characteristics based on the hierarchical evolutionary model. *Thermal Sci. Eng. Progress* **46**, 102212 (2023).
38. Galvin, R. Energy consumption effects of speed and acceleration in electric vehicles: Laboratory case studies and implications for drivers and policymakers. *Transp. Res. D Transp. Environ.* **53**, 234–248 (2017).
39. Kudaruskas, N. Analysis of emergency braking of a vehicle. *Transport* **22**, 154–159 (2007).
40. Hoffmann, H. Kernel PCA for novelty detection. *Pattern Recogn.* **40**, 863–874 (2007).
41. Zhang, T., Ramakrishnan, R. & Livny, M. BIRCH: An efficient data clustering method for very large databases. *ACM Sigmod Record* **25**, 103–114 (1996).
42. Andrieu, C. *et al.* An introduction to MCMC for machine learning. *Mach. Learn.* **50**, 5–43 (2003).
43. Liou, C. Y. *et al.* Autoencoder for words. *Neurocomputing* **139**, 84–96 (2014).

Acknowledgements

This research work is jointly funded by the Technology Innovation and Application Development Project of Chongqing (CSTB2022TIAD-STX0005), the National Key Research and Development Program (No. 2020YFB1506003), the Outstanding Youth Fund of Hunan Province (2022JJ10009) and the Hunan Provincial Innovation Foundation for Postgraduate (No. CX20230421). The authors are grateful to the reviewers and editors for their careful review of the manuscript and for the many constructive comments and suggestions for improvement.

Author contributions

Z.Z.: Data analysis and Writing this manuscript; X.S.: Investigation, Supervision, Conceptualization, Methodology and Reviewing; X.W.: Supervision and Conceptualization; Y.W.: Methodology and Investigation; J.F.: Conceptualization and Reviewing; J.L.: Methodology and Reviewing.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to X.S. or J.F.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2024